



CLASSIFICAÇÃO NEURAL HIERÁRQUICA DE SINAIS DE SONAR PASSIVO
COM EXPLORAÇÃO DA FUSÃO DAS INFORMAÇÕES TEMPORAL E
ESPECTRAL

Marlon Jovenil de Souza

Dissertação de Mestrado 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 Mestre em Engenharia Elétrica.

Orientador: José Manoel de Seixas

Rio de Janeiro
Outubro de 2022

CLASSIFICAÇÃO NEURAL HIERÁRQUICA DE SINAIS DE SONAR PASSIVO
COM EXPLORAÇÃO DA FUSÃO DAS INFORMAÇÕES TEMPORAL E
ESPECTRAL

Marlon Jovenil de Souza

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

Orientador: José Manoel de Seixas

Aprovada por: Prof. José Manoel de Seixas

Prof. Natanael Nunes de Moura Jr.

Prof. José Antonio Apolinário Junior

Prof. Adriaio Duarte Doria Neto

Capitão-de-Corveta Gustavo Augusto Mascarenhas Goltz

RIO DE JANEIRO, RJ – BRASIL

OUTUBRO DE 2022

Jovenil de Souza, Marlon

CLASSIFICAÇÃO NEURAL HIERÁRQUICA DE SINAIS DE SONAR PASSIVO COM EXPLORAÇÃO DA FUSÃO DAS INFORMAÇÕES TEMPORAL E ESPECTRAL/Marlon Jovenil de Souza. – Rio de Janeiro: UFRJ/COPPE, 2022.

XXII, 269 p.: il.; 29, 7cm.

Orientador: José Manoel de Seixas

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2022.

Referências Bibliográficas: p. 103 – 110.

1. Sistemas de Sonar Passivo. 2. Inteligência Computacional. 3. Fusão de Informação Espectral e Temporal. 4. Aprendizado Profundo. I. Manoel de Seixas, José. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

*A minha mãe: Marli Jovenil de
Souza e meus irmãos: Adriana,
Rayza, Marcus Vinícius e
Luanna Abreu.*

Agradecimentos

Agradeço primeiramente a Deus pela minha vida e por ter me dado a oportunidade de chegar até aqui. A minha mãe, familiares e amigos agradeço pelo amor, carinho, exemplos, conselhos, amizade e ajuda. Agradeço ao professor Seixas, meu orientador, por ter confiado em mim e apostado em meu trabalho, pela sua paciência para ensinar-me e orientar-me durante essa jornada. Agradeço aos integrantes do LPS e LabSonar pelo companheirismo, em especial ao amigo Júlio de Castro Vargas Fernandes. Agradeço ao Prof. e amigo Natanael Nunes de Moura Júnior pelas discussões e ensinamentos ao longo de todo o desenvolvimento do trabalho, além da motivação nos momentos difíceis. A Marinha do Brasil, em especial, ao pessoal do IPQM e ao Capitão-Tenente Fábio Oliveira, pelo apoio e oportunidade de realizar esse trabalho. Por fim, à COOPE-UFRJ, às pessoas que a compõem, desde a diretoria e coordenação até o pessoal de serviços gerais, por proporcionarem as melhores condições para que os alunos desenvolvam suas pesquisas.

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

CLASSIFICAÇÃO NEURAL HIERÁRQUICA DE SINAIS DE SONAR PASSIVO
COM EXPLORAÇÃO DA FUSÃO DAS INFORMAÇÕES TEMPORAL E
ESPECTRAL

Marlon Jovenil de Souza

Outubro/2022

Orientador: José Manoel de Seixas

Programa: Engenharia Elétrica

O ruído acústico irradiado por navios propagando-se nos mares é detectado e classificado por submarinos militares através de técnicas que utilizam o sistema sonar passivo. Este sistema também é utilizado para monitoramento da costa brasileira e em aplicações no meio civil. A diversidade de navios existentes e a interferência do ruído ambiente tornam complexa a tarefa de identificação dos contatos, logo o desenvolvimento de sistemas que realizam identificação e classificação automática usando algoritmos de processamento de sinais e de inteligência computacional vêm ganhando espaço na área de pesquisas militares e no meio acadêmico. Muitos trabalhos na literatura propõem classificadores treinados com informações espectrais e poucos alvos envolvidos, o que está longe da realidade na qual deparamos com muitas classes e sinais no domínio do tempo. Essa quantidade de classes prejudica a eficiência do classificador. Sendo assim, este trabalho propõe a classificação hierárquica de sinais de sonar passivo, explorando a fusão das informações temporais e espectrais, por meio de duas análises (LOFAR e Temporal) e o uso de redes neurais profundas (mais especificamente as convolucionais e recorrentes) com o objetivo de melhorar a classificação de um sistema de sonar passivo. Nesta análise foram utilizados dois bancos de dados experimentais: o primeiro adquirido por submarinos em operação e o segundo de um sonar passivo monitorando o tráfego marítimo de uma região portuária do Rio de Janeiro. Os resultados mostram que o método proposto e as redes recorrentes apresentam bons resultados, $88,76\% \pm 2,81\%$ e $85,55\% \pm 3,28\%$ respectivamente, em comparação com as redes neurais tradicionais.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

HIERARCHICAL NEURAL CLASSIFICATION OF PASSIVE SONAR SIGNALS WITH THE EXPLOITATION OF THE FUSION OF TEMPORAL AND SPECTRAL INFORMATION

Marlon Jovenil de Souza

October/2022

Advisor: José Manoel de Seixas

Department: Electrical Engineering

The acoustic noise radiated by ships propagating in the seas is detected and classified by military submarines through passive sonar system techniques. This system is also used for monitoring the Brazilian coast and in applications in the civil environment. Due to the diversity of existing ships, interference from ambient noise and the complexity of identification of contacts, thus the development of systems that perform automatic identification and classification using processing algorithms of signals and computational intelligence have been gaining ground in the area of military and academic research. Many works in the literature address the development of ship classifiers trained with spectral information and with few targets involved, which is far from the reality in which we face a great number of classes and signals in the time domain. This amount of classes also becomes a challenge for a classifier. Thus this work proposes the hierarchical classification of passive sonar signals, exploring the fusion of temporal and spectral information through two analyzes (LOFAR and Temporal) and the use of deep neural networks (more specifically convolutional and recurrent) in order to make the classification of sonar system better. In this analysis two experimental datasets were used: the first acquired from submarine in operation and the second from a passive sonar monitoring the maritime traffic of a port region of Rio de Janeiro. The results show that the proposed method and the recurrent network present good results, $88,76\% \pm 2,81\%$ and $85,55\% \pm 3,28\%$ respectively in comparison with traditional neural networks.

Sumário

Lista de Figuras	xi
Lista de Abreviaturas	xxi
1 Introdução	1
1.1 Motivação	4
1.2 Objetivo	6
1.3 Organização e Conteúdo da Dissertação	7
2 Sistema Sonar	9
2.1 Equação Sonar para Sistemas de Sonar	10
2.1.1 Sistema de Sonar Ativo	11
2.1.2 Sistema de Sonar Passivo	12
2.2 Subsistemas de um Sistema de Sonar	13
2.2.1 Conformador de Feixes	14
2.2.2 Análise de Marcação em Frequência	14
2.2.3 Análise de Marcação no Tempo	18
2.2.4 Acompanhamento	18
2.2.5 Classificadores	18
3 Máquinas de Comitê	25
4 Método Proposto	32
4.1 Classificador Multi-Classes e Hierárquico	33
4.2 Processamento dos Sinais	35
4.3 Modelos de Redes Neurais	38
4.4 Treinamento da Rede Neural	39
4.4.1 Validação Cruzada	40
4.4.2 Figuras de Mérito	41
4.5 Fusão dos Modelos	42

5	Resultados e Discussões	46
5.1	Conjuntos de Dados Utilizados	46
5.2	Processamento de Sinais	47
5.3	Definição da Arquitetura e Treinamento dos Modelos	48
5.3.1	Águas Profundas (24 Classes)	49
5.3.2	Águas Rasas (31 Classes)	78
6	Conclusões e Trabalhos Futuros	100
6.1	Conclusões	100
6.2	Trabalhos Futuros	102
	Referências Bibliográficas	103
A	Fusão dos Modelos	111
A.1	Águas Profundas	111
A.1.1	Fusão MLP com MLP	111
A.1.2	Fusão CNN com CNN	115
A.1.3	Fusão MLP com LSTM	119
A.1.4	Fusão CNN com LSTM	122
A.2	Águas Rasas	126
A.2.1	Fusão MLP com MLP	126
A.2.2	Fusão CNN com CNN	129
A.2.3	Fusão MLP com LSTM	130
A.2.4	Fusão CNN com LSTM	136
B	Aprendizado de Máquina	140
B.1	Redes Neurais Artificiais	140
B.1.1	Rede Neural MultiLayer Perceptron	141
B.1.2	Rede Neural Convolutacional	142
B.1.3	Rede Neural Recorrente	144
C	Resultados Complementares	149
C.1	Águas Profundas	149
C.1.1	Rede Neural MLP	149
C.1.2	Rede Neural CNN	149
C.1.3	Rede Neural LSTM	154
C.2	Águas Rasas	155
C.2.1	Rede Neural MLP	155
C.2.2	Rede Neural CNN	156
C.2.3	Rede Neural LSTM	157
C.3	Curvas de Treinamento	160

C.3.1	Águas Profundas	160
C.3.2	Águas Rasas	169
C.4	Histogramas de saída da Rede	178
C.4.1	Águas Profundas	178
C.4.2	Águas Rasas	223
D	Publicações	268
D.1	Artigos em Congressos	268

Lista de Figuras

1.1	Composição da Amazônia Azul, sua extensão e limites.	3
1.2	Lançamento do primeiro Submarino construído pelo PROSUB.	4
2.1	Diferentes fontes de ruído no mar.	9
2.2	Bolhas sendo formadas pelo processo de cavitação na hélice de um navio.	11
2.3	Diagrama em blocos dos principais componentes de um sistema de sonar passivo.	13
2.4	Conformador de feixes do tipo atraso e soma.	15
2.5	Diagrama em blocos da Análise LOFAR.	15
2.6	Janela usada para remover o ruído de fundo com o algoritmo TPSW.	16
2.7	Gráfico das informações de frequência de um determinado alvo ao longo do tempo (lofargrama).	17
2.8	Diagrama em blocos da Análise DEMON.	17
2.9	Análise DEMON.	17
2.10	Diagrama em blocos da Análise Temporal.	18
2.11	Fluxograma de fusão de decisão de diferentes tipos de dados.	24
3.1	Diagrama esquemático da rede neural integrada.	31
3.2	Diagrama em blocos do método de fusão com Limiar Adaptativo.	31
4.1	Diagrama em blocos do método proposto.	32
4.2	Comitê hierárquico do conjunto de águas profundas oriundo de sinais de submarinos.	34
4.3	Comitê hierárquico do conjunto de águas rasas oriundo do VIPP.	34
4.4	Diagrama de blocos da Análise LOFAR.	35
4.5	Diagrama em blocos da Análise Temporal.	35
4.6	LOFARgrama para diferentes índices de decimação.	36
4.7	LOFARgrama para quantidades diferentes de pontos na janela de processamento.	37
4.8	Sinais de áudio com a representação das janelas de interesse.	37
4.9	Processo de validação cruzada com 10 subconjuntos.	41

4.10	Modelo de classificação com rede de fusão para um elemento da célula de comitê.	43
4.11	Processo de montagem do modelo proposto.	45
5.1	Vigilância e Informações Passivas em Portos (VIPP).	47
5.2	Gráfico do índice SP pela variação do número de neurônios (águas profundas).	50
5.3	Gráfico de Violino do desempenho dos modelos MLP no conjunto de validação (águas profundas).	52
5.4	Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40, <i>fold</i> 3 (águas profundas).	53
5.5	Matriz de Confusão dos modelos LOFAR MLP com 30 e 40 neurônios (águas profundas).	54
5.6	Histogramas da saída do modelo temporal MLP-30, <i>fold</i> 3 (águas profundas).	55
5.7	Matriz de Confusão do modelo temporal MLP com 30 neurônios (águas profundas).	56
5.8	Gráfico do índice SP pela variação do número de modelos CNN (águas profundas).	57
5.9	Gráfico de Violino do desempenho dos modelos LOFAR CNN no conjunto de validação (águas profundas).	59
5.10	Histogramas da saída dos modelos LOFAR CNN (8,8,10,3) e (8,8,20,3), <i>fold</i> 3 (águas profundas).	60
5.11	Matriz de Confusão dos modelos LOFAR CNN com 10 e 20 neurônios na camada <i>fully connected</i> (águas profundas).	61
5.12	Gráfico de Violino do desempenho dos modelos temporais CNN no conjunto de validação (águas profundas).	62
5.13	Histogramas da saída dos modelos temporais CNN (8,8,10,3) e (8,8,20,3), <i>fold</i> 3 (águas profundas).	63
5.14	Matriz de Confusão dos modelos temporais CNN com 10 e 20 neurônios na camada <i>fully connected</i> (águas profundas).	64
5.15	Gráfico do índice SP pela variação do número de células LSTM (águas profundas).	65
5.16	Gráfico de Violino do desempenho dos modelos temporais LSTM no conjunto de validação (águas profundas).	66
5.17	Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75, <i>fold</i> 3 (águas profundas).	67
5.18	Matriz de Confusão dos modelos temporais LSTM com 50 e 75 células (águas profundas).	68

5.19	Matriz de Confusão do conjunto de teste do modelo MLP, CNN e LSTM (águas profundas).	70
5.20	Matriz de Confusão do conjunto de validação da fusão dos modelos (águas profundas).	73
5.21	Histogramas da saída da fusão dos modelos MLP-MLP e MLP-LSTM do conjunto de validação (águas profundas).	74
5.22	Histogramas da saída da fusão dos modelos CNN-CNN e CNN-LSTM do conjunto de validação (águas profundas).	75
5.23	Matriz de Confusão do conjunto de teste da fusão dos modelos (águas profundas).	76
5.24	Gráfico do índice SP pela variação do número de neurônios (águas rasas).	78
5.25	Gráfico de Violino do desempenho dos modelos MLP no conjunto de validação (águas rasas).	80
5.26	Histogramas da saída do modelo LOFAR MLP com 10 neurônios (águas rasas).	80
5.27	Matriz de Confusão do modelo LOFAR MLP com 10 neurônios (águas rasas).	81
5.28	Histogramas da saída dos modelos temporais MLP-10 e MLP-20, <i>fold</i> 3 (águas rasas).	82
5.29	Matriz de Confusão dos modelos temporais MLP com 10 e 20 neurônios (águas rasas).	82
5.30	Gráfico do índice SP pela variação do número de modelos CNN (águas rasas).	83
5.31	Gráfico de Violino do desempenho dos modelos LOFAR CNN no conjunto de validação (águas rasas).	85
5.32	Histogramas da saída dos modelos LOFAR CNN (8,8,10,3) e (8,8,20,3), <i>fold</i> 3 (águas rasas).	86
5.33	Matriz de Confusão dos modelos LOFAR CNN com 10 e 20 neurônios na camada <i>fully connected</i> (águas rasas).	86
5.34	Gráfico de Violino do desempenho dos modelos temporais CNN no conjunto de validação (águas rasas).	87
5.35	Histogramas da saída dos modelos temporais CNN (8,8,10,3) e (8,8,20,3), <i>fold</i> 3 (águas rasas).	88
5.36	Matriz de Confusão dos modelos temporais CNN com 10 e 20 neurônios na camada <i>fully connected</i> (águas rasas).	89
5.37	Gráfico do índice SP pela variação do número de células LSTM (águas rasas).	90

5.38	Gráfico de Violino do desempenho dos modelos temporais LSTM no conjunto de validação (águas rasas).	91
5.39	Histogramas da saída do modelo temporal LSTM-50, <i>fold 3</i> (águas rasas).	92
5.40	Matriz de Confusão do modelo temporal LSTM com 50 células (águas rasas).	92
5.41	Matriz de Confusão do conjunto de teste do modelo MLP, CNN e LSTM (águas rasas).	94
5.42	Matriz de Confusão do conjunto de validação da fusão dos modelos (águas rasas).	97
5.43	Histogramas da saída da fusão dos modelos MLP-MLP e MLP-LSTM (águas rasas).	97
5.44	Histogramas da saída da fusão dos modelos CNN-CNN e CNN-LSTM (águas rasas).	98
5.45	Matriz de Confusão do conjunto de teste da fusão dos modelos (águas rasas).	98
A.1	Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos MLP (águas profundas).	112
A.2	Histogramas da saída da fusão dos modelos MLP, <i>fold 3</i> (águas profundas).	113
A.3	Matriz de Confusão da fusão dos modelos MLP com 10 e 30 neurônios na rede de fusão (águas profundas).	114
A.4	Curvas de treinamento e validação da fusão dos modelos MLP (águas profundas).	115
A.5	Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos CNN (águas profundas).	116
A.6	Histogramas da saída da fusão dos modelos CNN, <i>fold 3</i> (águas profundas).	116
A.7	Matriz de Confusão da fusão dos modelos CNN sem e com rede de fusão de 10 neurônios (águas profundas).	117
A.8	Curvas de treinamento e validação da fusão dos modelos CNN (águas profundas).	118
A.9	Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos MLP com LSTM (águas profundas).	119
A.10	Histogramas da saída da fusão dos modelos MLP com LSTM, <i>fold 3</i> (águas profundas).	120
A.11	Matriz de Confusão da fusão dos modelos MLP com LSTM sem e com a rede de fusão com 10 neurônios (águas profundas).	121

A.12	Curvas de treinamento e validação da fusão dos modelos MLP com LSTM (águas profundas).	122
A.13	Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos CNN com LSTM (águas profundas).	123
A.14	Histogramas da saída da fusão do modelo CNN com LSTM (<i>fold 3</i>).	123
A.15	Matriz de Confusão da fusão dos modelos CNN com LSTM sem e com a rede de fusão com 10 neurônios (águas profundas).	124
A.16	Curvas de treinamento e validação da fusão dos modelos CNN com LSTM (águas profundas).	125
A.17	Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos MLP (águas rasas).	127
A.18	Histogramas da saída da fusão dos modelos MLP, <i>fold 3</i> (águas rasas).	127
A.19	Matriz de Confusão da fusão dos modelos MLP (águas rasas).	128
A.20	Curvas de treinamento e validação da fusão dos modelos MLP (águas rasas).	128
A.21	Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos CNN (águas rasas).	129
A.22	Histogramas da saída da fusão dos modelos CNN, <i>fold 3</i> (águas rasas).	130
A.23	Matriz de Confusão da fusão dos modelos CNN sem e com rede de fusão de 10 neurônios (águas rasas).	131
A.24	Curvas de treinamento e validação da fusão dos modelos CNN com e sem a rede de fusão (águas rasas).	132
A.25	Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos MLP com LSTM (águas rasas).	133
A.26	Histogramas da saída da fusão dos modelos MLP com LSTM, <i>fold 3</i> (águas rasas).	133
A.27	Matriz de Confusão da fusão dos modelos MLP e LSTM (águas rasas).	134
A.28	Curvas de treinamento e validação da fusão do modelo MLP e LSTM com e sem a rede de fusão (águas rasas).	135
A.29	Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos CNN com LSTM (águas rasas).	136
A.30	Histogramas da saída da fusão dos modelos CNN com LSTM, <i>fold 3</i> (águas rasas).	137
A.31	Matriz de Confusão da fusão dos modelos CNN e LSTM (águas rasas).	138
A.32	Curvas de treinamento e validação da fusão dos modelos CNN e LSTM com e sem a rede de fusão (águas rasas).	139
B.1	Rede <i>Perceptron</i>	141
B.2	Redes <i>Multilayer Perceptron</i>	142

B.3	Operação de convolução entre a entrada e o filtro.	143
B.4	Processo de <i>Pagging</i> em uma imagem.	144
B.5	Topologia CNN.	145
B.6	Comparação da arquitetura de uma rede <i>feedforward</i> e uma recorrente.	145
B.7	Rede Neural Recorrente e o desdobramento de seu processo temporal.	146
B.8	Módulo de repetição em um LSTM.	147
B.9	Representação de um <i>gate</i> da célula LSTM e ao lado o gráfico de uma função sigmóide.	148
C.1	Curvas de treinamento e validação dos modelos LOFAR MLP, <i> folds</i> 3 e 4 (águas profundas).	150
C.2	Curvas de treinamento e validação do modelo temporal MLP com 30 neurônios, <i> folds</i> 3 e 4 (águas profundas).	150
C.3	Curvas de treinamento e validação dos modelos LOFAR (8,8,10,3) e (8,8,20,3), <i> folds</i> 3 e 4 (águas profundas).	151
C.4	Curvas de treinamento e validação dos modelos temporais (8,8,10,3) e (8,8,20,3), <i> folds</i> 3 e 4 (águas profundas).	154
C.5	Curvas de treinamento e validação dos modelos temporais LSTM, <i> folds</i> 3 e 4 (águas profundas).	155
C.6	Curvas de treinamento e validação do modelo LOFAR MLP com 10 neurônios, <i> folds</i> 3 e 4 (águas rasas).	155
C.7	Curvas de treinamento e validação dos modelos temporais MLP com 10 e 20 neurônios, <i> folds</i> 3 e 4 (águas rasas).	156
C.8	Curvas de treinamento e validação dos modelos LOFAR (8,8,10,3) e (8,8,20,3), <i> folds</i> 3 e 4 (águas rasas).	157
C.9	Curvas de treinamento e validação dos modelos temporais (8,8,10,3) e (8,8,20,3), <i> folds</i> 3 e 4 (águas rasas).	158
C.10	Curvas de treinamento e validação do modelo temporal LSTM, <i> folds</i> 3 e 4 (águas rasas).	158
C.11	Curvas de Treinamento do modelo LOFAR MLP (águas profundas).	160
C.12	Curvas de Treinamento do modelo temporal MLP (águas profundas).	161
C.13	Curvas de Treinamento do modelo LOFAR CNN (águas profundas).	162
C.14	Curvas de Treinamento do modelo temporal CNN (águas profundas).	163
C.15	Curvas de Treinamento do modelo temporal LSTM (águas profundas).	164
C.16	Curvas de Treinamento da fusão dos modelos MLP (águas profundas).	165
C.17	Curvas de Treinamento da fusão dos modelos CNN (águas profundas).	166
C.18	Curvas de Treinamento da fusão dos modelos MLP e LSTM (águas profundas).	167

C.19	Curvas de Treinamento da fusão dos modelos CNN e LSTM (águas profundas).	168
C.20	Curvas de Treinamento do modelo LOFAR MLP (águas rasas).	169
C.21	Curvas de Treinamento do modelo temporal MLP (águas rasas).	170
C.22	Curvas de Treinamento do modelo LOFAR CNN (águas rasas).	171
C.23	Curvas de Treinamento do modelo temporal CNN (águas rasas).	172
C.24	Curvas de Treinamento do modelo temporal LSTM (águas rasas).	173
C.25	Curvas de Treinamento da fusão dos modelos MLP (águas rasas).	174
C.26	Curvas de Treinamento da fusão dos modelos CNN (águas rasas).	175
C.27	Curvas de Treinamento da fusão dos modelos MLP e LSTM (águas rasas).	176
C.28	Curvas de Treinamento da fusão dos modelos CNN e LSTM (águas rasas).	177
C.29	Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (<i>folds</i> 1 e 2).	178
C.30	Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (<i>folds</i> 3 e 4).	179
C.31	Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (<i>folds</i> 5 e 6).	180
C.32	Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (<i>folds</i> 7 e 8).	181
C.33	Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (<i>folds</i> 9 e 10).	182
C.34	Histogramas da saída do modelo temporal MLP-30 (<i>folds</i> 1 e 2).	183
C.35	Histogramas da saída do modelo temporal MLP-30 (<i>folds</i> 3 e 4).	184
C.36	Histogramas da saída do modelo temporal MLP-30 (<i>folds</i> 5 e 6).	185
C.37	Histogramas da saída do modelo temporal MLP-30 (<i>folds</i> 7 e 8).	186
C.38	Histogramas da saída do modelo temporal MLP-30 (<i>folds</i> 9 e 10).	187
C.39	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 1 e 2).	188
C.40	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 3 e 4).	189
C.41	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 5 e 6).	190
C.42	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 7 e 8).	191
C.43	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 9 e 10).	192
C.44	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 1 e 2).	193
C.45	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 3 e 4).	194
C.46	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 5 e 6).	195
C.47	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 7 e 8).	196
C.48	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 9 e 10).	197

C.49 Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (<i>folds</i> 1 e 2).	198
C.50 Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (<i>folds</i> 3 e 4).	199
C.51 Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (<i>folds</i> 5 e 6).	200
C.52 Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (<i>folds</i> 7 e 8).	201
C.53 Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (<i>folds</i> 9 e 10).	202
C.54 Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 1 e 2).	203
C.55 Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 3 e 4).	204
C.56 Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 5 e 6).	205
C.57 Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 7 e 8).	206
C.58 Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 9 e 10).	207
C.59 Histogramas da saída do modelo CNN da fusão (<i>folds</i> 1 e 2).	208
C.60 Histogramas da saída do modelo CNN da fusão (<i>folds</i> 3 e 4).	209
C.61 Histogramas da saída do modelo CNN da fusão (<i>folds</i> 5 e 6).	210
C.62 Histogramas da saída do modelo CNN da fusão (<i>folds</i> 7 e 8).	211
C.63 Histogramas da saída do modelo CNN da fusão (<i>folds</i> 9 e 10).	212
C.64 Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 1 e 2).	213
C.65 Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 3 e 4).	214
C.66 Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 5 e 6).	215
C.67 Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 7 e 8).	216
C.68 Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 9 e 10).	217
C.69 Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 1 e 2).	218
C.70 Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 3 e 4).	219
C.71 Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 5 e 6).	220
C.72 Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 7 e 8).	221

C.73	Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 9 e 10).	222
C.74	Histogramas da saída do modelo LOFAR MLP-10 (<i>folds</i> 1 e 2).	223
C.75	Histogramas da saída do modelo LOFAR MLP-10 (<i>folds</i> 3 e 4).	224
C.76	Histogramas da saída do modelo LOFAR MLP-10 (<i>folds</i> 5 e 6).	225
C.77	Histogramas da saída do modelo LOFAR MLP-10 (<i>folds</i> 7 e 8).	226
C.78	Histogramas da saída do modelo LOFAR MLP-10 (<i>folds</i> 9 e 10).	227
C.79	Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (<i>folds</i> 1 e 2).	228
C.80	Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (<i>folds</i> 3 e 4).	229
C.81	Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (<i>folds</i> 5 e 6).	230
C.82	Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (<i>folds</i> 7 e 8).	231
C.83	Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (<i>folds</i> 9 e 10).	232
C.84	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 1 e 2).	233
C.85	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 3 e 4).	234
C.86	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 5 e 6).	235
C.87	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 7 e 8).	236
C.88	Histogramas da saída dos modelos LOFAR CNN (<i>folds</i> 9 e 10).	237
C.89	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 1 e 2).	238
C.90	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 3 e 4).	239
C.91	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 5 e 6).	240
C.92	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 7 e 8).	241
C.93	Histogramas da saída dos modelos temporais CNN (<i>folds</i> 9 e 10).	242
C.94	Histogramas da saída dos modelos temporais LSTM (<i>folds</i> 1 e 2).	243
C.95	Histogramas da saída dos modelos temporais LSTM (<i>folds</i> 3 e 4).	244
C.96	Histogramas da saída dos modelos temporais LSTM (<i>folds</i> 5 e 6).	245
C.97	Histogramas da saída dos modelos temporais LSTM (<i>folds</i> 7 e 8).	246
C.98	Histogramas da saída dos modelos temporais LSTM (<i>folds</i> 9 e 10).	247
C.99	Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 1 e 2).	248
C.100	Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 3 e 4).	249
C.101	Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 5 e 6).	250
C.102	Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 7 e 8).	251
C.103	Histogramas da saída da fusão dos modelos MLP (<i>folds</i> 9 e 10).	252
C.104	Histogramas da saída do modelo CNN da fusão (<i>folds</i> 1 e 2).	253
C.105	Histogramas da saída do modelo CNN da fusão (<i>folds</i> 3 e 4).	254

C.106	Histogramas da saída do modelo CNN da fusão (<i>folds</i> 5 e 6).	255
C.107	Histogramas da saída do modelo CNN da fusão (<i>folds</i> 7 e 8).	256
C.108	Histogramas da saída do modelo CNN da fusão (<i>folds</i> 9 e 10).	257
C.109	Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 1 e 2).	258
C.110	Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 3 e 4).	259
C.111	Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 5 e 6).	260
C.112	Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 7 e 8).	261
C.113	Histogramas da saída da fusão do modelo MLP com LSTM (<i>folds</i> 9 e 10).	262
C.114	Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 1 e 2).	263
C.115	Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 3 e 4).	264
C.116	Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 5 e 6).	265
C.117	Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 7 e 8).	266
C.118	Histogramas da saída da fusão do modelo CNN com LSTM (<i>folds</i> 9 e 10).	267

Lista de Abreviaturas

ART	Adaptive Resonance Theory, p. 19
CIRM	Comissão Interministerial para os Recursos do Mar, p. 1
CNN	Convolutional Neural Network, p. 5
CP	Curvas Principais, p. 21
DEMON	Detection of Envelope Modulation on Noise, p. 4
DOA	Direction of Arrival, p. 3
FFT	Fast Fourier Transform, p. 20
GAN	Generative Adversarial Neural Networks, p. 23
ICA	Independent Component Analysis, p. 19
IPqM	Instituto de Pesquisas da Marinha, p. 4
KNN	K-Nearest Neighbor, p. 21
LMS	Least Men Square, p. 31
LOFAR	LOw Frequency Analysis and Recording, p. 4
LSTM	Long Short-Term Memory, p. 25
MB	Marinha do Brasil, p. 1
MLP	Multilayer Perceptron, p. 6
MLP	Multilayé superior er Perceptron, p. 59
NLPCA	NonLinear Principal Component Analysis, p. 19
OS	Operador Sonar, p. 5
PCA	Principal Component Analysis, p. 19

PCD	Principal Component Discrimination, p. 19
PNN	Probabilistic Neural Network, p. 21
PROSUB	Programa de Desenvolvimento de Submarinos, p. 2
RBF	Radial Basis Function, p. 21
RF	Random Forest, p. 22
RJ	Rio de Janeiro, p. 22
RNA	Redes Neurais Artificiais, p. 5
RNNs	Redes Neurais Recorrentes, p. 5
SAE	Stacked AutoEncoders, p. 21
SDAC	Sistema de Detecção, Acompanhamento e Classificação de Contatos, p. 46
SNR	Relação Sinal Ruído, p. 10
SONAR	Sound Navigation and Ranging, p. 3
SP	Soma-Produto, p. 20
STFT	Short-time Fourier Transform, p. 15
SVM	Suport Vector Machine, p. 20
TPSW	Two-Pass Split Window, p. 15
UFEM	Unidade de Fabricação de Estruturas Metálicas, p. 3
VIPP	Vigilância e Informações Passivas em Portos, p. 5
ZEE	Zona Econômica Exclusiva, p. 1

Capítulo 1

Introdução

A Marinha do Brasil (MB) tem por missão preparar e empregar o poder naval a fim de contribuir para a Defesa da Pátria e propõe estar permanentemente pronta para atuar no mar e em águas interiores, de forma singular ou conjunta, de modo a atender os propósitos de sua missão [1]. O Brasil possui cerca de 7.400 *Km* de costa, e sob sua jurisdição 3,5 milhões de quilômetros quadrados de espaço marítimo [2]. Tal área somente o Brasil pode explorar economicamente, e por conta das riquezas naturais e minerais abundantes, é chamada de Amazônia Azul, numa alusão à importância da floresta amazônica para o país [2]. Essa extensa área marítima tem uma importância inquestionável para a economia brasileira por ser a principal via de transporte para o comércio exterior do País, por sua diversidade de recursos naturais, por suas reservas de petróleo e gás e outros recursos minerais [2]. A Figura 1.1 apresenta a composição da Amazônia Azul. Desde 2004, o Brasil reivindica, junto à Comissão de Limites da Plataforma Continental, a extensão dos direitos econômicos sobre a faixa marítima, chamada de Zona Econômica Exclusiva (ZEE¹) [3]. As reservas do pré-sal estão dentro da ZEE e delas são retiradas 85% do petróleo, 75% do gás natural e 45% do pescado produzido no país [4]. Por consequência disso, o Brasil tenta acelerar o processo de extensão da ZEE brasileira temendo que as áreas adjacentes passem a ser cobiçadas por outros países em virtude destes recursos [4]. A Comissão Interministerial para os Recursos do Mar (CIRM) também orienta o desenvolvimento das atividades que visem à efetiva utilização, exploração e aproveitamento sustentável dos recursos naturais da Amazônia Azul e áreas internacionais, de acordo com os interesses do Brasil e incentiva a formação de recursos humanos

¹A Zona econômica exclusiva (ZEE) começa a ser medida a partir das linhas de base adotadas pelo Estado costeiro e estende-se até a distância máxima de 200 milhas marítimas (370 km de extensão). O Estado costeiro tem direitos de soberania para fins de exploração e ao aproveitamento, conservação e gestão dos recursos naturais, das águas sobrejacentes ao leito do mar, do leito do mar e seu subsolo, e no que se refere a outras atividades com vista à exploração e aproveitamento da ZEE [2].

na área de Ciências do Mar², estimulando o desenvolvimento de pesquisa e inovação nas diversas áreas do conhecimento [3]. Para entender melhor a composição da Amazônia Azul é preciso elucidar três conceitos [2]:

1. Mar territorial: Estende-se das linhas de base adotadas pelo Estado costeiro até a extensão máxima de 12 milhas náuticas (22 km). No mar territorial, o Estado costeiro exerce soberania plena sobre a massa líquida e o espaço aéreo sobrejacente ao mar territorial, bem como ao leito e subsolo deste mar;
2. Zona contígua: Estado costeiro mantém o controle sob uma área de até 12 milhas náuticas, adicionalmente às 12 milhas do mar territorial, para o propósito de evitar ou reprimir infrações às suas leis e regulamentos aduaneiros, fiscais, de imigração e sanitários no seu território ou mar territorial;
3. Plataforma continental: A ser estabelecida conforme os critérios técnicos e condicionantes do Artigo 76 da Lei do Mar. Na plataforma continental, o Estado costeiro exerce direitos de soberania para efeitos de exploração e aproveitamento dos seus recursos naturais, que são os recursos minerais, dentre outros, além dos recursos do leito do mar. Inclui-se também os recursos do subsolo.

Diante do exposto, a MB intensificou sua atuação na defesa da soberania brasileira sobre o espaço, fazendo constantes operações de monitoramento com seus meios (navios de superfície, submarinos e outros), equipamentos de patrulhamento e investindo em projetos como a construção do primeiro submarino nuclear [2].

Esse projeto é chamado de Programa de Desenvolvimento de Submarinos (PROSUB) [6]. A Estratégia Nacional de Defesa [7], lançada em 2008, estabeleceu que o Brasil deve ter uma “força naval de envergadura”, incluindo submarinos com propulsão nuclear. Sendo assim, em um acordo de transferência de tecnologia firmado entre Brasil e França, deu-se início ao PROSUB, o qual viabilizará a produção de quatro submarinos convencionais com propulsão a diesel, que se somarão à frota de cinco submarinos já existentes e culminará na fabricação do primeiro submarino brasileiro de propulsão nuclear [6]. Um dos pontos importantes do PROSUB é o fomento da indústria brasileira na área de tecnologia nuclear e de vários setores da indústria nacional de importância estratégica para o desenvolvimento econômico do país [8]. Isso porque o PROSUB visa priorizar a aquisição de componentes fabricados no Brasil para os submarinos [7]. Vale ressaltar que o programa envolve a construção de um complexo de infraestrutura industrial e de apoio à operação dos submarinos, que engloba os Estaleiros, a Base Naval e a Unidade de Fabricação de

²Ciências do Mar é a área do saber que se dedica à produção e disseminação de conhecimentos sobre os componentes, processos e recursos do ambiente marinho e zonas de transição [5].

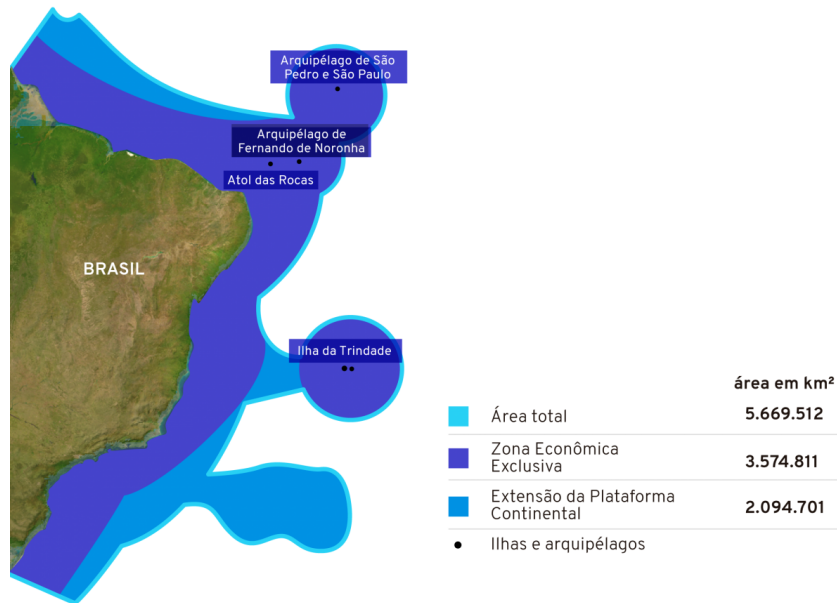


Figura 1.1: Composição da Amazônia Azul, sua extensão e limites.

Fonte: MARINHA DO BRASIL [2]

Estruturas Metálicas (UFEM) no Município de Itaguaí [6]. A Figura 1.2 apresenta a cerimônia de lançamento em dezembro de 2020 de um dos submarinos de propulsão a diesel do programa, o S.Humaitá.

O submarino militar utiliza sistemas de *Sound Navigation and Ranging* (SONAR), os quais são um conjunto de técnicas que se utilizam das propriedades do som se propagando na água do mar para navegação, comunicação, detecção de outros navios e classificação de ameaças, dentre outras tarefas [9]. Portanto, esta tecnologia é de fundamental importância para a MB. Os navios de superfícies podem ser detectados utilizando métodos baseados em ondas eletromagnéticas de alta-frequência, como as utilizadas em sistema radar [8]. Entretanto, esses métodos são ineficazes para identificação de um submarino devido à atenuação das ondas eletromagnéticas se propagando na água [10]. Dessa maneira, é mais apropriada a utilização de ondas mecânicas por serem mais eficientes em aplicações submarinas (como no caso de sistemas sonar), por terem baixa atenuação e se propagarem por longas distâncias [11].

Existem dois tipos principais de sistemas de Sonar: o passivo e o ativo [12]. O primeiro não emite sinal e utiliza a emissão acústica dos contatos de interesse para detectá-los [9] e classificá-los [8]. Já o segundo emite um sinal e analisa seu eco para detectar os contatos [9]. Os principais objetivos de um sistema de sonar passivo são a estimação da direção de aproximação ou DOA (*Direction of Arrival*), a detecção de contatos, sua identificação, bem como seu acompanhamento [12]. A estimação da DOA é feita utilizando algoritmos de conformação de feixes [11]. Dado



Figura 1.2: Lançamento do primeiro Submarino construído pelo PROSUB.

Fonte: MARINHA DO BRASIL [6]

que houve detecção de um sinal de interesse em uma determinada direção, o sinal é submetido a uma extração de características visando a sua classificação [8]. A MB utiliza, principalmente, duas análises para a extração de características: a análise LOFAR (Low Frequency Analysis and Recording) e a análise DEMON (*Detection of Envelope Modulation on Noise*) [13]. Essas duas análises serão aprofundadas no Capítulo 2. Em suma, as características extraídas por essas análises podem ser usadas para classificação dos contatos com o apoio do aprendizado de máquina. O aprendizado de máquina [14] é um subcampo da inteligência artificial dedicado ao desenvolvimento de algoritmos e técnicas que permitam o treinamento de modelos de classificação, estimação, predição ou agrupamento de dados [8].

Vale destacar que além da aplicação no meio militar, o SONAR tem diversas aplicações no meio civil. Das quais podemos destacar os sonares de pesca, ecobatímetros, sonares de monitoramento de atividade sísmica, equipamentos de ultrassom utilizados na área médica, sonares de varredura os quais são utilizados para exploração do leito dos mares e rios e exploração mineral, dentre outras aplicações [15].

1.1 Motivação

É de interesse estratégico da MB manter pessoal capacitado em estudos de sistema sonar, por se tratar de uma tecnologia de defesa muito importante, além de atender ao monitoramento das águas jurisdicionais do Brasil. Sendo assim, o Instituto de Pesquisas da Marinha (IPqM) vem intensificando pesquisas de identificação e

classificação de sinais em águas profundas, além de desenvolver o projeto VIPP (Vigilância e Informações Passivas em Portos), capaz de detectar e acompanhar navios através dos seus ruídos acústicos adquiridos por um sonar passivo [16].

O militar responsável pelas tarefas de identificação e classificação de contatos durante a operação de um submarino é o Operador Sonar (OS) . Ele é treinado para ouvir os sinais de áudio do domínio do tempo e classificá-los segundo as características observadas [17]. Dada a difícil e sensível tarefa do OS, faz-se necessário um sistema que realize a detecção e classificação de forma automática, por meios de algoritmos de processamento de sinais e inteligência computacional, para apoiar o OS em sua tomada de decisão [17].

Os sinais de sistema de sonar passivo são ruídos oriundos de diversas fontes sonoras presentes no ambiente subaquático (fenômenos sísmicos, tempestades, vida marinha, plataformas de petróleo, dentre outras), todavia, os ruídos de interesse para este trabalho são provenientes de embarcações [18]. Sendo assim, os outros ruídos do ambiente subaquático são considerados ruídos de fundo. Em um cenário tático, por causa da grande quantidade de ruídos de fundo presentes no ambiente marítimo, diversos contatos podem ser detectados ao mesmo tempo pelo sistema sonar, o que torna desafiadora a tarefa de identificação e classificação das embarcações [19]. Dada essa complexidade, diversas técnicas para classificação de sinais de sistema sonar vêm sendo desenvolvidas e testadas [18]. Atualmente, alguns sistemas militares processam os sinais por meio das análises LOFAR e DEMON [13], para a extração de características, conforme citado anteriormente. Após o processamento, o emprego de inteligência computacional (usada com sucesso em problemas de reconhecimento de padrões para classificação), em particular, as Redes Neurais Artificiais (RNA³) têm sido usadas em sonar passivo para a classificação de contatos demonstrando boa eficiência [18, 21]. Em razão disso, queremos avançar nesta direção. Os trabalhos que envolvem a utilização das RNA em sonar serão apresentados no Capítulo 2.

Dentre as RNA, as técnicas de redes neurais profundas (*deep learning*), nome dado a uma classe de algoritmos de aprendizado de máquinas que utilizam várias camadas de processamento, se destacam em modelar abstrações de alto nível de dados [22]. Elas vêm obtendo bons resultados na classificação de sinais em diversos trabalhos na literatura [18, 23, 24]. Em especial, podemos citar duas redes, convolucional e recorrente. A Rede Neural Convolucional (*Convolutional Neural Network* - CNN) é uma classe de rede neural artificial que vem sendo aplicada com sucesso no processamento e análise de imagens digitais e séries temporais [22]. A Rede Neural Recorrente (*Recurrente Neural Networks* - RNN) é uma classe de redes neurais artificiais bastante útil para o processamento de dados sequenciais, tais como séries

³As RNA são uma família de técnicas de aprendizado de máquina baseadas em um modelo computacional inspirados na estrutura de uma rede de neurônios natural [20].

temporais ou linguagem natural [22]. Não se pode negligenciar que redes não profundas do tipo *Perceptron* Multicamadas (*Multilayer Perceptron*, MLP) também são aplicadas em classificação de sinais de sistema de sonar passivo com ótimos desempenhos [25, 26]. Portanto, é também interessante investigar o uso dessas redes.

Alguns trabalhos na literatura foram desenvolvidos com propostas de classificação hierárquica [15, 27], valendo-se do conceito de comitê de máquinas (vide Capítulo 3): “produzir ganhos de desempenho frente as propostas de soluções isoladas, ou seja, que não formam comitês” [28]. Essa proposta é bastante interessante no meio naval, pois possibilita o agrupamento de classes de navios em determinados conjuntos de características comuns com o intuito de aumentar a eficiência de classificação. Tendo em vista que este trabalho pretende utilizar *datasets* com muitas classes, a aplicação de uma estrutura hierárquica pode ser benéfica para a classificação.

Uma das maneiras de analisar o sinal é observá-lo no domínio do tempo, o que faz a extração de características não depender de tantos parâmetros como nas análises em frequência. Ademais, o domínio do tempo é o domínio no qual o OS classifica os sinais obtidos. Alguns trabalhos na área de classificação de sinais de sistema sonar passivo utilizam informação temporal em conjunto com informações espectrais (no domínio da frequência), a fim de melhorar a eficiência de classificação [29, 30]. Apesar de trazer alguns benefícios, essa abordagem é pouco utilizada na literatura, o que permite explorar diversas alternativas a serem implementadas.

1.2 Objetivo

Diversos classificadores de ruído de navios têm sido propostos na literatura, entretanto com falta de enfoque operacional. Devido à dificuldade em adquirir dados reais de sistema de sonar para o desenvolvimento de soluções mais próximas das necessidades de um submarino, muitos dos trabalhos presentes na literatura empregam poucas classes no desenvolvimento do problema e/ou com o uso de dados sintéticos. Além de abordarem dados coletados em um ambiente controlado. Dada essa dificuldade, GOLTZ [15] propõe a criação de dois bancos de dados com sinais mais próximos de uma situação real: O primeiro banco é composto por gravações de sinais acústicos de navios em regime operativo via sistema sonar de atuais submarinos da MB navegando submersos em águas profundas, contendo 24 classes. E o outro são gravações de sinais acústicos coletados de um sonar passivo, denominado Vigilância e Informações Passivas em Portos (VIPP) na baía da Guanabara, monitorando o tráfego marítimo em águas rasas numa região portuária (contendo 31 classes) no Rio de Janeiro. Esses dados são dispostos hierarquicamente com base em conhecimento especialista em navios e classificados com classificadores especialistas baseados em modelos não profundos. Este trabalho abordará apenas o nível mais alto da dispo-

sição hierárquica (descrita com detalhes na Subseção 4.1) de cada conjunto. Sendo assim, o conjunto de águas profundas terá 4 subclasses e o conjunto de águas rasas terá 2 subclasses que serão classificadas.

Este trabalho tem por objetivo:

- Dar continuidade ao estudo referente à classificação hierárquica de navios com o uso de sinais de sistema de sonar passivo, abordado por GOLTZ [15], utilizando os dois bancos de dados propostos e desenvolvendo modelos de classificação não especialistas baseados em técnicas de redes neurais profundas (mais especificamente, convolucionais e recorrentes) com os dados processados no domínio do tempo e da frequência.
- Desenvolver um modelo de classificação capaz de fundir as informações temporais e espectrais dos sinais do sistema de sonar passivo e comparar seus resultados com os classificadores treinados por apenas uma informação (temporal ou espectral).

A fusão será feita treinando um classificador com base nos mapeamentos obtidos dos modelos treinados com os sinais de sistema de sonar passivo processados de duas maneiras distintas: A primeira é através de uma análise de extração de características do domínio da frequência conhecida como análise LOFAR [21], e a outra no domínio do tempo chamada de Análise Temporal [19]. Esses processamentos combinados com diferentes modelos de redes neurais geram classificações com diferentes visões acerca do sinal de entrada. Essas visões são fundidas a fim de melhorar as estimativas de classificação de um sistema de sonar passivo se comparadas a classificação oriunda de apenas uma visão. Uma vez que os dados de sonar passivo, normalmente, envolvem muitas classes, a avaliação utilizando apenas os índices tradicionais, como acurácia de eficiência não podem ser o critério único para avaliação. Dada a dificuldade de se trabalhar com muitas classes, outra figura de mérito foi utilizada e será descrita com mais detalhes no Capítulo 4.

1.3 Organização e Conteúdo da Dissertação

A organização do texto é a seguinte: O Capítulo 2 apresenta uma breve introdução de um sistema sonar e suas características. Além de uma revisão da literatura sobre os métodos de aprendizado de máquina explorados até o momento para classificação de sistemas de sonar. No Capítulo 3 explica-se o funcionamento das Máquinas Comitês e das propostas de fusão das Redes Neurais Artificiais utilizadas no desenvolvimento deste trabalho. Dentro do Capítulo 4 o método aplicado para obter os resultados experimentais será apresentado em detalhe. No Capítulo 5 são apresentados os resultados experimentais obtidos durante o desenvolvimento deste trabalho

e, por fim, no Capítulo 6 são apresentadas as conclusões e os trabalhos futuros. As referências bibliográficas são apresentadas após o capítulo final.

Capítulo 2

Sistema Sonar

Este capítulo tem por objetivo descrever o funcionamento de um sistema sonar além de cobrir os requisitos de processamento de sinais necessários para a classificação dos sinais irradiados pelos navios.

Conforme mencionado no Capítulo 1, o sistema de Sound Navigation and Ranging (SONAR) é um conjunto de técnicas que utilizam as propriedades do som se propagando na água do mar. Várias fontes de sinal acústico contribuem para o ruído recebido pelo sistema sonar, como vida marinha, navios, plataformas marítimas de petróleo, fenômenos sísmicos e diversas outras dificultando o processo de identificação e classificação do sinal desejado [15]. A Figura 2.1 apresenta as diferentes fontes de sinal acústico presentes nos oceanos, com suas respectivas faixas de frequência e níveis de ruído.

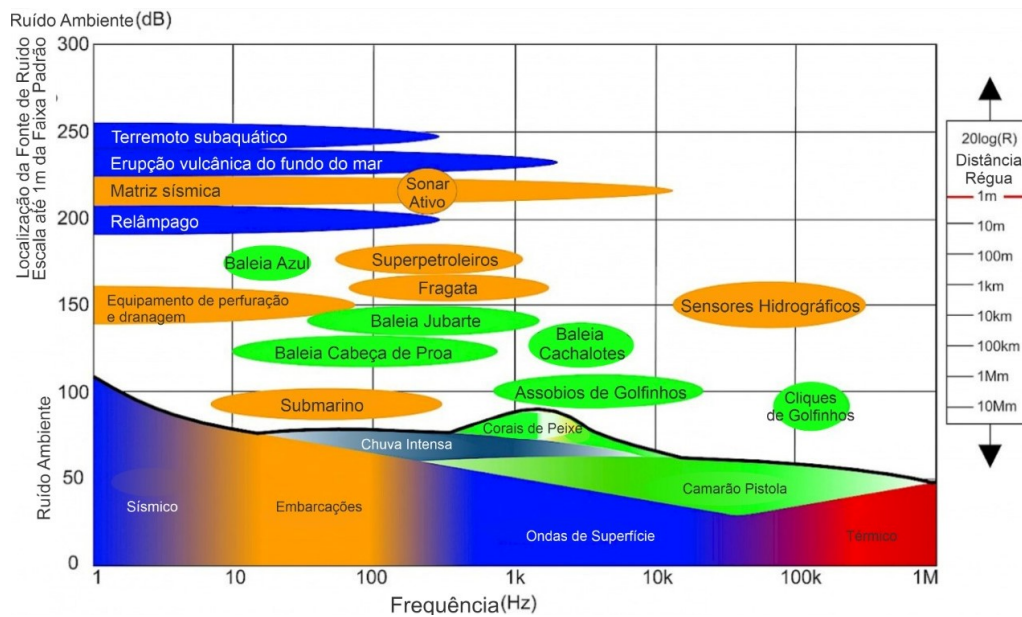


Figura 2.1: Diferentes fontes de ruído no mar.

Fonte: <https://www.ospar.org/work-areas/eiha/noise>

Dentre os diversos ruídos detectados pelo sistema sonar, os ruídos de interesse oriundos dos navios são provenientes de três principais fontes e são chamados de: ruído de máquinas, ruído da hélice e ruído hidrodinâmico [21, 31, 32]. O ruído da hélice é uma combinação dos ruídos de cavitação e pela ressonância da vibração do casco do navio [9]. O processo de cavitação ocorre quando a hélice gira na água criando regiões de alta e baixa pressão (na parte anterior e posterior da pá da hélice). Se essas regiões sofrerem uma quantidade de pressão suficiente para romper a resistência natural da água, ocorre a formação de bolhas de ar. Essas bolhas emitem um pulso sonoro na água, gerando ruído quando colapsam por serem atingidas pelo próprio hélice ou pelo fluxo de água. O espectro do ruído gerado é dominado por altas frequências em média. O fenômeno de cavitação pode ser visto na Figura 2.2. Vale enfatizar que o ruído de cavitação também é modulado em amplitude pela rotação de hélice que está diretamente ligada à velocidade do navio e seu número de pás. Este processo de modulação contribui na classificação dos sinais produzidos pelos navios de interesse adquiridos pelo sistema sonar [8].

O ruído de máquinas é predominantemente composto pelo ruído gerado pelo sistema de propulsão do navio e por ruídos gerados pelas máquinas auxiliares, como geradores, bombas, equipamentos de ar condicionados, entre outros. Os ruídos são propagados pela vibração de seu casco enquanto o navio está se movendo na água.

O ruído hidrodinâmico é gerado pela passagem da água pelo casco da embarcação enquanto ela se encontra em movimento. Este ruído varia de forma irregular, pois o contato da água com o casco do navio é irregular de acordo com o movimento das ondas (o casco também pode ser irregular) e o quanto o casco está submerso também influencia neste fenômeno. O ruído hidrodinâmico ainda pode entrar em processo de ressonância com partes do casco, gerando fenômenos que podem auxiliar no processo de classificação [21, 33]. 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 [8].

2.1 Equação Sonar para Sistemas de Sonar

A equação que modela a propagação dos ruídos no ambiente marinho é chamada de equação sonar [9]. Ela é uma maneira de estimar as amplitudes de sinais de interferências que podem ser observadas nos transdutores do sistema sonar, obtendo assim a SNR do sistema [8]. O sistema de sonar tem como objetivo detectar sinais dos quais possuem ruídos indesejados e medir o desempenho desta detecção [17]. Esse desempenho pode ser medido através da equação sonar, a qual leva em consideração o nível da fonte emissora de sinal, o espalhamento do som, a absorção sonora, as perdas de reflexão, o nível de ruído ambiente e as características do receptor. A



Figura 2.2: Bolhas sendo formadas pelo processo de cavitação na hélice de um navio.

Fonte: <https://www.google.com>

equação pode ter seus parâmetros modificados de acordo com dois cenários em que pode ser aplicada, o primeiro chamamos de sistema de sonar ativo e o outro de sistema de sonar passivo [8].

2.1.1 Sistema de Sonar Ativo

No sistema de sonar ativo há a transmissão de um pulso sonoro para a análise do eco recebido e pode ser utilizado para diversas tarefas, tais como ecosondas, localizadores de peixes, sonar militar, dentre outras. A equação sonar de sistema de sonar ativo é:

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

onde as variáveis da Equação (2.1) são descritas como:

- SL - Potência da onda emitida pelo emissor;
- TL - Atenuação da emissão causada pelo meio de propagação;
- TS - Potência de reflexão do pulso emitido após o navio de interesse;
- NL - Ruído de fundo; e
- AG - As características dos sensores de recepção.

Ressaltando que o ruído de fundo mencionado (NL) é a soma de todas as fontes sonoras que não são de interesse e todos os termos são medidos em decibéis. Um sinal com uma potência de fonte (SL) é transmitido pelo sonar ativo. Após a transmissão ocorre uma perda de transmissão (TL), pois o som é atenuado pelo meio de propagação na medida em que o sinal se propaga em direção ao alvo. Para o alvo, a intensidade do som se torna então $SL - TL$. A sistema de sonar ativo recebe apenas a parte do som ($SL - TL$) que foi refletida pelo alvo dado que a intensidade do eco a um metro do alvo é dada por TS . A intensidade do eco retornado é dada por $(SL - TL) + TS - TL$. Como o nível de ruído do receptor é NL , a relação sinal-ruído no receptor pode ser calculada como $SL - 2TL + TS - (NL - AG)$ [8, 17].

2.1.2 Sistema de Sonar Passivo

O sistema sonar passivo adquire sons gerados por diversas fontes sonoras e não emite sons. Sendo assim, ele analisa os ecos refletidos de um determinado contato. A equação sonar de sistema de sonar passivo é:

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

onde as variáveis da Equação (2.2) são descritas como:

- SL - Potência da onda emitida pelo alvo;
- TL - Atenuação da emissão causada pelo meio de propagação;
- NL - Ruído de fundo; e
- AG - As características dos sensores de recepção.

Diferentemente da Equação (2.1), a Equação (2.2) desconsidera a intensidade do eco (TS) uma vez que não temos a transmissão de um pulso, como no sonar passivo [17]. Qualquer fonte sonora irradiadora de ruído emite um sinal com um nível de fonte SL . Dado que o som viaja a partir da fonte para o receptor, a perda de transmissão TL reduz a intensidade do som. Sendo assim, a intensidade do sinal no receptor é dada por $SL - TL$. No caso de um receptor composto por um único transdutor, a relação sinal ruído é $SL - TL - NL$. Assim como no sistema de sonar ativo, dado um sistema composto por vários transdutores ou sensores, temos que levar em consideração as características espaciais do receptor (AG) para determinar a relação sinal ruído do receptor [8].

2.2 Subsistemas de um Sistema de Sonar

A Figura 2.3 mostra a operação básica de um sistema sonar passivo. Os N sinais, representados por s_0, s_1, \dots, s_n , oriundos de navios de todas as direções possíveis são adquiridos por um conjunto de N transdutores chamados de hidrofones (que compõem o sistema de sonar passivo), tal que $n = 0, 1, \dots, N - 1$. Depois que os sinais são adquiridos, eles são processados por um conformador de feixes [15], ou seja, dada uma direção específica de chegada (neste estágio todas as direções possíveis de chegada são verificadas, ou seja, 360° de cobertura). Todos os sinais vindos dessa direção são enfatizados enquanto sinais e ruídos que chegam de outras direções são atenuados. Neste sentido a conformação de feixes pode ser entendida como uma filtragem espacial para determinar as importantes direções de chegada [17]. Após os sinais serem conformados, estes podem ser observados tanto no domínio do tempo quanto no domínio da frequência. Os dois domínios podem ser usados para detectar os sinais tornando possível realizar o seu acompanhamento e posteriormente o processo de classificação é iniciado.

Analisando as Equações (2.1) e (2.2) podemos observar que devido à ausência da emissão do pulso de referência por parte do sistema de sonar passivo, a SNR tende a ser menor que o do sonar ativo. Sendo assim, o processo de detecção de um contato se torna mais desafiador. Em virtude disso, os diversos subsistemas citados no parágrafo anterior são acoplados ao sistema sonar com o intuito de tornar o processo de detecção mais eficiente [8].

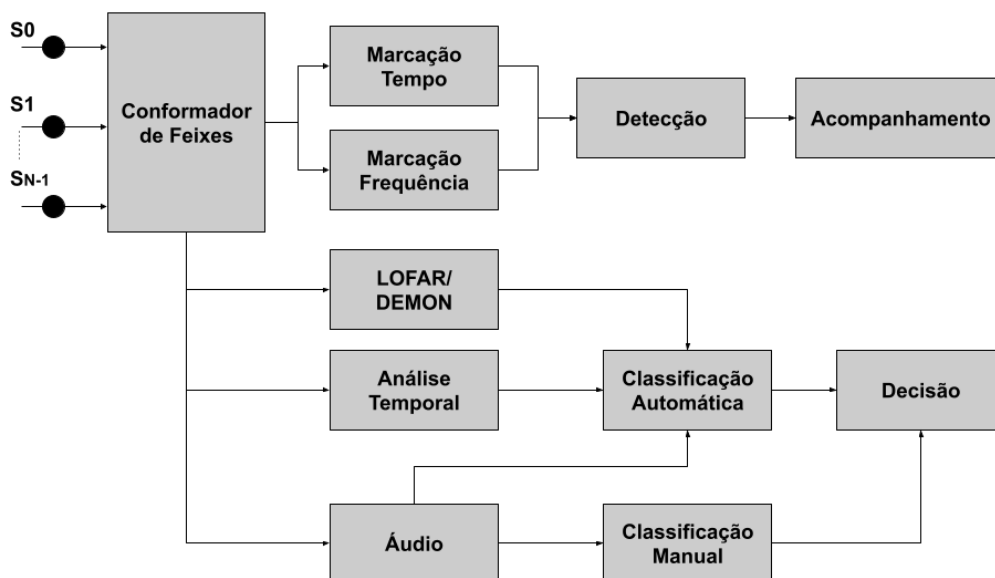


Figura 2.3: Diagrama em blocos dos principais componentes de um sistema de sonar passivo.

2.2.1 Conformador de Feixes

Nos sonares modernos são usados vários hidrofones em uma configuração matricial para a captura dos sinais de interesse o que também contribui para a supressão dos ruídos e interferências do ambiente [17]. O processo de conformação de feixe, também chamado de *beamforming*, nada mais é do que uma filtragem espacial dos sinais provenientes dos transdutores para obtenção das DOA [8], de modo a observar a energia de um determinado contato em uma determinada direção. Embora haja diversas técnicas para a realização da conformação de feixes [34], a mais comum consiste em fazer a composição dos sinais com o intuito de cancelar ou adicionar atrasos de tempo decorrentes das diferenças de tempo da chegada da frente de onda em cada um dos sensores. Cada configuração específica de arranjo (linear, cilíndrico, esférico, planar...) precisa empregar uma compensação diferente para lidar com a geometria específica que está sendo usada [8]. A Figura 2.4 mostra o diagrama de um conformador de feixes do tipo atraso e soma. Nela podemos observar um arranjo com N hidrofones, sendo $s_n(t)$ o sinal recebido no n ésimo hidrofone, tal que $n = 0, 1, \dots, N - 1$. Os sinais provenientes do arranjo de sensores são as entradas $s_n(t)$, τ_n são os deslocamentos temporais inseridos para direcionar os sinais em uma determinada direção e ω_n são ponderações usadas para formar a resposta do conformador. Quando os atrasos temporais são iguais aos atrasos atuais de um sinal que está se propagando através do arranjo, este sinal é somado de forma coerente pelo conformador e reforçado na sua direção. São atenuados os sinais que chegam de outras direções, pois são somados de forma incoerente pelo conformador [17].

A etapa de conformação de feixes é fundamental para a direção de chegada dos contatos, após esta etapa os sinais passam por uma etapa de processamento para extração de algumas características que contribuam para a tarefa de detecção, acompanhamento e classificação do sistema de sonar passivo. As principais análises utilizadas neste trabalho e também para apoio a tomada de decisão em aplicações militares são: a análise LOFAR e análise DEMON. Além disso, esta dissertação propõe uma nova análise para extração de características do sinal denominada de análise temporal.

2.2.2 Análise de Marcação em Frequência

Análise LOFAR

O sinal conformado por ser analisado no domínio da frequência. Este domínio tem por objetivo encontrar tons de frequências que auxiliem as tarefas de detecção e classificação do contato. A análise LOFAR é uma análise espectral de banda larga que caracteriza o ruído de frequência das máquinas do contato [35]. A análise

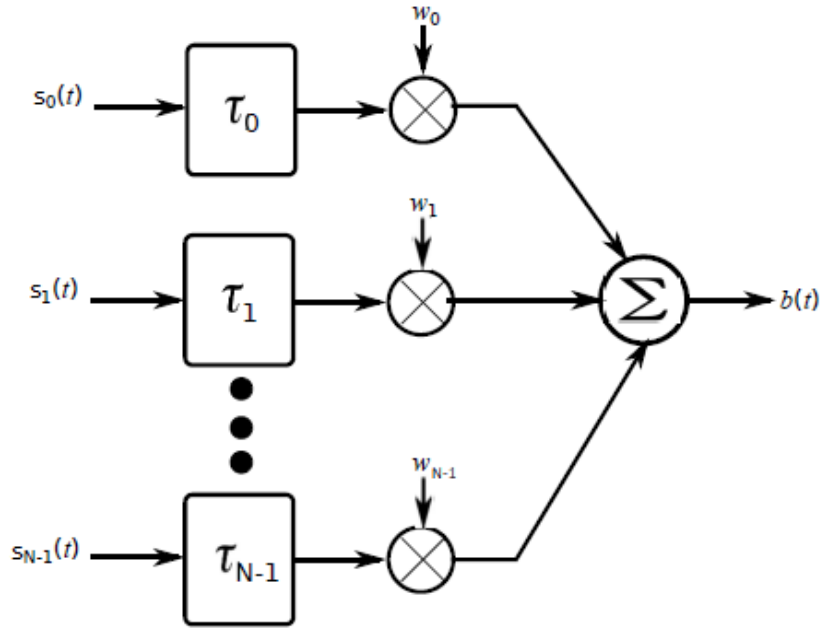


Figura 2.4: Conformador de feixes do tipo atraso e soma.

Fonte: DE MOURA [17]

utilizada no desenvolvimento deste trabalho foi proposta por SOARES FILHO [21] e foi aplicada em diversos trabalhos científicos em colaboração com a MB [8, 15, 18]. O método depende de especificações como: tamanho da janela de processamento, índice de decimação e parâmetros da estimação do ruído de fundo [8].

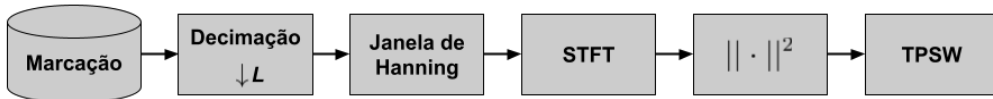


Figura 2.5: Diagrama em blocos da Análise LOFAR.

O diagrama de blocos exemplificando a análise LOFAR está descrito na Figura 2.5. Nela podemos constatar que os sinais já no domínio do tempo discreto passam por uma etapa de decimação com o intuito de fazer a seleção da banda de interesse do sistema de sonar passivo. A análise é seguida por uma janela de *Hanning* [36] para redução do efeito *Aliasing* quando for aplicado ao sinal à STFT [37]. Após a STFT o módulo do espectro é normalizado e por fim, a informação passa por uma função *Two-Pass Split Window* (TPSW) com o propósito de estimar o ruído de fundo do ambiente de medição. A Figura 2.6 mostra como é realizada a filtragem usando o TPSW. O algoritmo estima o ruído aplicando uma janela específica a cada um dos *bins* do espectro estimado. A janela se desloca ao longo do sinal realizando uma média local removendo assim o ruído e normalizando o sinal [21].

A Figura 2.7 mostra uma representação típica da análise LOFAR. O lofargrama,

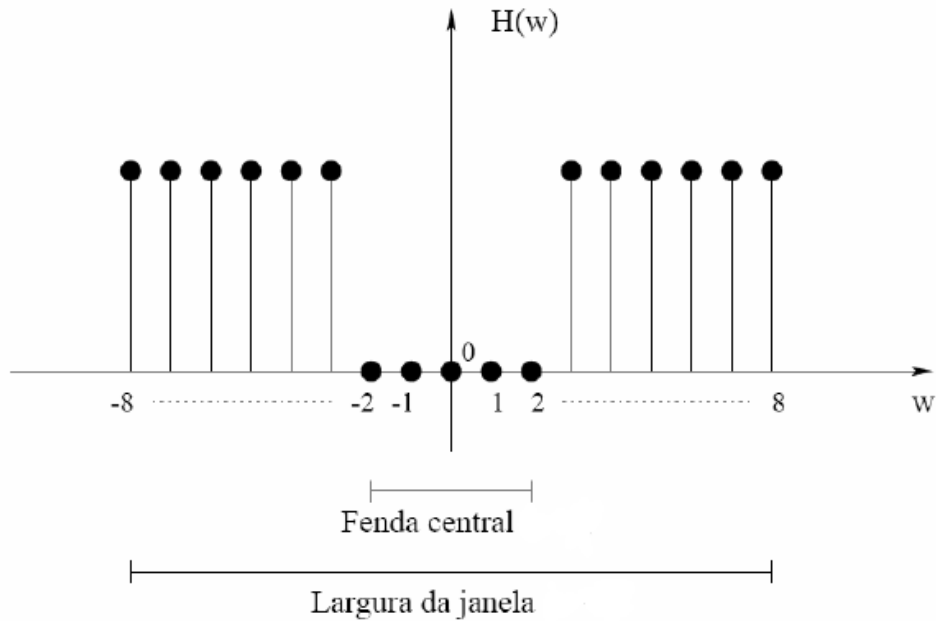


Figura 2.6: Janela usada para remover o ruído de fundo com o algoritmo TPSW.

Fonte: DE MOURA [17]

como é denominada essa representação, é uma representação do sinal através de um gráfico cartesiano onde as abcissas são as frequências, as ordenadas são o tempo e as amplitudes são representadas numa escala de cor. Com isso podemos acompanhar no tempo o comportamento das amplitudes das frequências [18].

Análise DEMON

Análise DEMON é uma análise de banda estreita que fornece informações sobre o ruído de cavitação da propulsão do contato com o propósito de identificar o número de eixos, sua frequência de rotação e o número de pás [12, 17]. A Figura 2.8 mostra o diagrama de blocos de como funciona a análise DEMON. O sinal é filtrado por um filtro passa-faixa que ajusta a banda onde o sinal de cavitação é mais acentuado [38]. Após a filtragem, o módulo do sinal é reamostrado para que o sinal seja transposto para a banda de propulsão, já que a frequência de amostragem do sinal é mais elevada. Ao sinal é aplicada uma transformada de Fourier de tempo curto (STFT - *Short Time Fourier Transform*) para obtenção do espectro do sinal. Por fim, assim como mencionado anteriormente, o sinal passa por uma função TPSW com o propósito de estimar o ruído de fundo do ambiente de medição [21].

A Figura 2.9 apresenta um exemplo de uma análise DEMON. O gráfico é preenchido ao longo do tempo, formando um gráfico em cascata ou *waterfall display*. As estrias observadas são pontos de maior concentração de energia nas direções em questão ao longo do tempo.

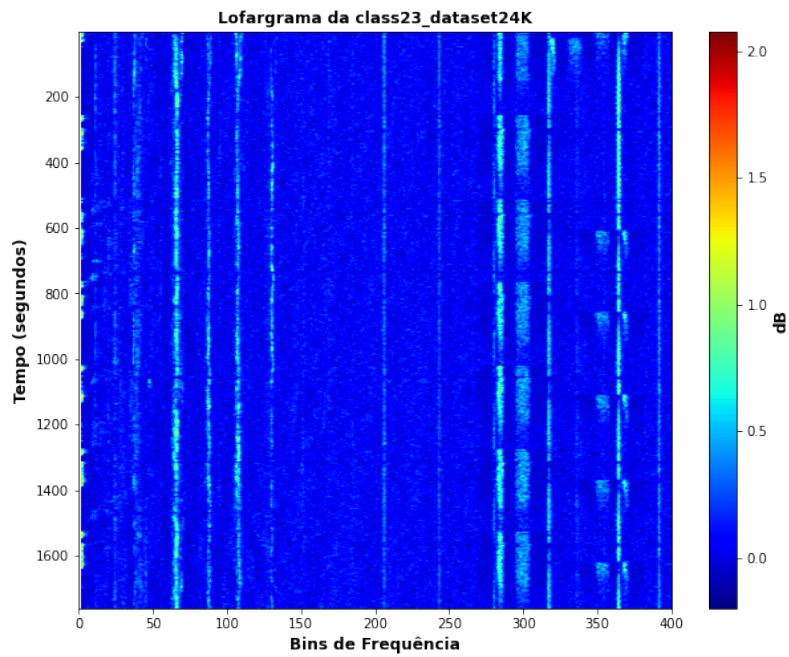


Figura 2.7: Gráfico das informações de frequência de um determinado alvo ao longo do tempo (lofargrama).

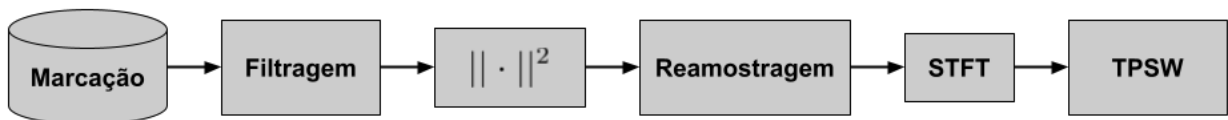


Figura 2.8: Diagrama em blocos da Análise DEMON.

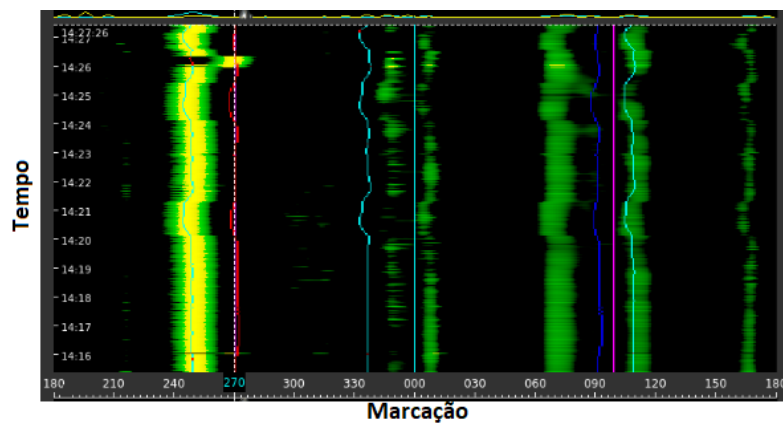


Figura 2.9: Análise DEMON.

2.2.3 Análise de Marcação no Tempo

Análise Temporal

A análise temporal distingue-se das outras análises anteriormente citadas, análise LOFAR e análise DEMON, uma vez que ela utiliza e processa as informações dos sinais apenas no domínio do tempo. A Figura 2.10 mostra o diagrama de blocos de uma proposta de análise temporal. Este diagrama foi baseado nas etapas da análise LOFAR. Os sinais de áudios oriundos das marcações passam primeiro por uma decimação, após esta etapa o sinal passa por um filtro passa-baixa para fazer a seleção da banda de interesse. A passagem pelo filtro é de grande importância, pois os sinais provenientes de navios geralmente têm relevância para baixas frequências [17]. Por fim, os sinais passam por uma normalização.

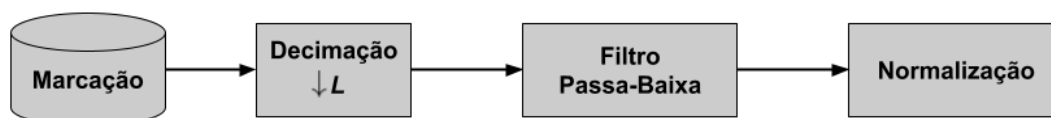


Figura 2.10: Diagrama em blocos da Análise Temporal.

2.2.4 Acompanhamento

Com as informações do alvo, tanto no domínio da frequência quanto no domínio do tempo, o processo de detecção do contato pode ser realizado. Após a conclusão do processo de detecção, o processo de acompanhamento do sistema de sonar passivo é iniciado. O processo de acompanhamento, ou *tracking*, recebe as informações dos subsistemas anteriores e faz uma classificação simples para identificar em qual direção o alvo em questão está se dirigindo. Com essa classificação, o sistema de acompanhamento pode rastrear cada um dos contatos classificados ao longo do tempo [8]. Várias técnicas de monitoramento em sistema de sonar passivo são discutidas na literatura [39, 40].

2.2.5 Classificadores

Após o sinal ser detectado e monitorado, dá-se início ao processo de classificação dos alvos para dar apoio a tomada de decisão na identificação de um contato por parte do OS [41]. Com acesso às informações disponíveis do contato, o processo de classificação pode ser realizado de diversas formas, e muitas delas estão presentes na literatura.

Em RAMOS [42], as características não estacionárias dos sinais provenientes do sistema de sonar passivo foram analisadas, acarretando a necessidade da utilização

de janelas para a análise. Ele propõe o desenvolvimento de um sistema neural de monitoração de sinais com base em uma rede não supervisionada ART (*Adaptive Resonance Theory*) modificada, na qual acompanha a estatística do sinal identificando possíveis alterações. A rede ART modificada foi utilizada para fazer periodicamente a clusterização dos espectros de frequência do sinal. A análise DEMON foi utilizada para a obtenção de características dos espectros do sinal. Foi utilizada a Análise de Componentes Principais (PCA) no pré-processamento do sinal para redução de dimensionalidade, devido à elevada dimensão dos dados de entrada. De forma auxiliar, o impacto do uso da Análise de Componentes Independentes (ICA) na eficiência de um classificador MLP também foi avaliada. Analisando os resultados constatou-se benefícios em termos da eficiência da classificação neural de contatos utilizando ICA no pré-processamento do sinal de sonar passivo. Para validação do sistema de monitoração proposto, foram realizados alguns testes fazendo uso da informação mútua e dos cumulantes cruzados de quarta ordem como figuras de mérito para análise da não estacionariedade estatística. Nos melhores casos, foi atingida uma eficiência de detecção de até 89%, para os cumulantes cruzados, e de até 75%, para a informação mútua. Os resultados demonstram a aplicabilidade do sistema de monitoração proposto como ferramenta de auxílio ao operador sonar na tomada de decisão.

SOARES FILHO [21] desenvolveu classificadores neurais para um conjunto de sinais de quatro classes de navios, coletados em uma raia acústica da MB, em Araraial do Cabo. É utilizada uma abordagem híbrida, onde se empregam técnicas não-neurais de pré-processamento, baseadas no conhecimento especialista de características específicas do sinal analisado. Dado esse conhecimento, na fase de pré-processamento foram avaliados algumas características de propagação do ruído no meio oceânico, como o impacto da resolução, da faixa de frequência a ser utilizada, dos tons mais significativos, dos números de *bins* do espectro e da técnica de eliminação do ruído de fundo, TPSW [43]. Após o pré-processamento, com o objetivo de avaliar a inclusão de uma nova classe a um sistema já treinado para as redes MLP e ART [44], os espectros foram compactados explorando 3 técnicas de compactação, PCA, Análise de Componentes Principais Não-Lineares (*NonLinear Principal Component Analysis* - NLPCA) [45] e Análise de Componentes Principais de Discriminação (PCD) [46]. A análise PCD apresentou melhores resultados em relação aos demais compactadores. A eficiência média alcançada para a rede MLP foi de 94%, e a ART atingiu 63%.

Em operações de guerra naval o sistema de sonar opera em condições muito ruidosas e, eventualmente, precisam ser identificadas novas classes de navios sem perda de eficiência nas classes para as quais um determinado sistema de suporte à decisão e o OS já foram treinados. DE MOURA JUNIOR e DE SEIXAS [35] abordaram o problema de detecção de uma nova classe para um classificador de sonar

passivo. Para o treinamento das classes conhecidas foi proposta uma Máquina de Vetores de Suporte (SVM: Suport Vector Machine) [47] como detector e após o treinamento as classes desconhecidas foram apresentadas ao detector treinado. Assim como em SOARES FILHO [21], o conjunto de dados utilizado possui quatro classes e o pré-processamento para compactação dos sinais é semelhante. Foi proposta quatro situações onde uma das quatro classes era a classe desconhecida. Os resultados obtidos por DE MOURA JUNIOR e DE SEIXAS [35] alcançaram uma eficiência de classificação para classes conhecidas variando entre 33,33% e 96,88%, e para a eficiência de detecção da classe desconhecida, os resultados variaram entre 72,84% e 87,09%.

Assim como descrito brevemente na Seção 1.2 deste trabalho, GOLTZ [15] propôs o desenvolvimento de um classificador que esteja mais próximo da realidade de um submarino. Para isso foram utilizados dois bancos de dados: um com gravações de sinais acústicos coletados de sonares passivos de submarinos em operação (contendo 24 classes) e o outro de gravações de sinais acústicos coletados de um sonar passivo monitorando o tráfego marítimo numa região portuária (contendo 31 classes), denominado Vigilância e Informações Passivas em Portos (VIPP). Quatro classificadores foram implementados: uma rede neural do tipo MLP, um comitê de classes especialistas composto por redes neurais, um comitê especialista composto por Máquinas de Vetor de Suporte e um comitê de redes neurais dispostas hierarquicamente, construído com base em conhecimento especialista em navios. Para treinamento dos classificadores foi estudado também o impacto da análise PCD. Para os conjuntos de dados de 24 e 31 classes, a topologia de MLP escolhida apresentou um valor médio com desvio-padrão do índice Soma-Produto (SP) de $0,80 \pm 0,00$ e $0,70 \pm 0,07$ respectivamente. Essa figura de mérito visa expressar, em um único número, a eficiência de classificação com sensibilidade para possíveis desbalanceamentos (descrito na Seção 4.4.2) [17, 25, 35]. Com relação aos resultados, de um modo geral, os classificadores tiveram mais dificuldade em classificar o conjunto de dados de 31 classes (oriundas de gravações do VIPP), do que do conjunto de 24 classes (oriundas de gravações de submarinos). Para o comitê hierárquico constatou-se através dos resultados que, uma vez operando juntamente com o OS, ele pode retornar para o usuário não somente sua saída do último nível hierárquico, mas sim, apresentar ao OS a saída de todos os níveis, facilitando assim a tomada de decisão.

ZUREK *et al.* [48] apresenta uma implementação de Redes Neurais Artificiais (RNA) para a classificação de três classes de navios através de sua assinatura acústica. Antes de entrar no processo de classificação da RNA, é aplicado ao sinal um pré-processamento (Filtro de Hamming [36], seguido da Transformada Rápida de Fourier - FFT) e uma compactação utilizando PCA, com o intuito de extrair suas características espectrais. Avaliou-se o uso entre 64 e 256 componentes em quatro

classificadores: KNN (K-Nearest Neighbor) [47], MLP, RBF (Radial Basis Function) [49] e PNN (Probabilistic Neural Network) [50]. Para cada classificador, os melhores índices de acerto foram 96,1% para RBF, 96,0% para MLP, 86,7% para KNN e 84,9% para PNN.

CERQUEIRA e DE SEIXAS [26] propuseram a utilização de uma das técnicas mais aplicadas de *Deep Learning* chamada de Stacked AutoEncoders (SAE) para inicializar os pesos de uma rede profunda de maneira não-supervisionada com o intuito de classificar sinais e detectar novidades em sonar passivo. A detecção de novidade consiste na detecção de classes que não estavam presentes durante o processo de desenvolvimento do sistema de classificação [51]. Devido ao grande número de contatos em um ambiente de operação, é comum que alguns contatos não tenham participado da etapa de desenvolvimento do classificador. O banco de dados utilizado é o mesmo usado em SOARES FILHO [21]. As corridas foram feitas em um ambiente controlado de 1500 metros e o hidrofone de aquisição foi colocado a 45 metros de profundidade. Os sinais são pré-processados pela análise LOFAR e classificados utilizando redes do tipo MLP e Convolucionais (CNN). Para as redes neurais artificiais MLP foram obtidos $84,27\% \pm 4,52\%$ na figura de mérito índice Soma-Produto, chamado de índice SP e descrito na Seção 4.4.2. Já para redes neurais profundas, os resultados foram obtidos com o aumento gradativo da complexidade das redes. O melhor resultado do índice SP foi de $88,92\% \pm 3,03\%$ e o mais baixo foi de $82,04\% \pm 3,93\%$ para o treinamento utilizando os pesos da segunda e terceira camadas do SAE.

FERNANDEZ [52] propõe um classificador de navios baseado em Curvas Principais (CP) com o auxílio de uma técnica de processamento estatístico empregada na representação de dados em um espaço de dimensão elevada. Os dados experimentais empregados para avaliação do classificador foram obtidos por um sistema de sonar passivo e imagens infravermelhas simuladas. Os sinais provenientes do sistema de sonar passivo irradiados por 25 navios, pertencentes a 8 classes distintas, durante 263 corridas de prova realizadas na raia acústica da MB situada em Arraial do Cabo no Rio de Janeiro, foram pré-processados sob a forma de espectrogramas e empregados na extração das curvas principais. Os classificadores baseados em CP obtiveram uma eficiência média de classificação de 96,8% considerando como critério de decisão a menor distância euclidiana do espectrograma recebido da curva extraída para cada classe. O segundo sistema de classificação a partir das imagens infravermelhas segmentadas visa classificar 5 classes diferentes através de 40.600 vistas de navios e obteve uma eficiência média de classificação igual a 97,3%.

DA SILVA MUNIZ *et al.* [27] revisita o problema abordado em FERNANDEZ [52], porém considerando o uso de classificadores arbitrários em ambos os níveis da hierarquia. Sendo assim, primeiramente, emprega-se um classificador para a iden-

tificação da classe de contato e, em seguida, é utilizado um segundo classificador especializado para identificar qual o navio mais provável. Na avaliação foram considerados os algoritmos de KNN e Random Forest (RF) [53]. O RF é uma técnica de comitê baseada no algoritmo de Árvores de Decisão [53], a qual aplica nos dados um conjunto de regras capazes de determinar seus rótulos. O algoritmo criará uma estrutura similar a um fluxograma, com “nós” onde é verificada uma condição, e se atendida, o fluxo segue por um ramo, caso contrário, por outro, seguindo para um próximo nó até a finalização da árvore. Assumindo que os dados descritos possuem p atributos, são criadas m árvores, cada uma considerando um subconjunto aleatório de s atributos, onde $s \leq m$, e uma amostra bootstrap dos dados. O KNN é uma técnica de aprendizado baseada em instâncias e explora a proximidade de um dado com seus vizinhos mais próximos. Essa proximidade pode ser inferida por meio da distância Euclidiana, Manhattan, Minkowski, Ponderada, dentre outras. Para um dado sob avaliação, são identificados os seus k vizinhos mais próximos. O rótulo mais frequente dentre os vizinhos é considerado para definir a saída do classificador. Ademais, KNN e RF [53] consistem em duas técnicas clássicas frequentemente empregadas em trabalhos na área do Aprendizado de Máquina [54]. A base de dados utilizada é oriunda de sinais gravados em uma raia acústica da MB, localizada em Arraial do Cabo-RJ e é constituída por 28 navios pertencentes a 8 classes distintas. Avaliando os resultados, conclui-se que o desempenho do KNN é superior ao RF no caso hierárquico e direto. O algoritmo KNN obteve uma acurácia média de acertos de navios de 95,6%.

VAHIDPOUR *et al.* [55] propõe um novo método para o reconhecimento e classificação de navios utilizando o som produzido e irradiado sob a água. Para isso, é proposto um procedimento em três etapas. Em primeiro lugar, as operações de pré-processamento são utilizadas para reduzir os efeitos do ruído e fornecem sinal para a extração de características. Em segundo lugar, uma imagem binária feita do espectro de frequência da segmentação do sinal, é formado para extrair características eficazes. Em terceiro lugar, um classificador neural (MLP) é projetado para classificar os sinais. O método proposto e o método baseado em fractais são comparados e testados com base em dados reais. Os resultados comparativos indicaram melhor capacidade de reconhecimento e mais desempenho robusto do método proposto do que no método baseado em fractais. Portanto, o método proposto poderia melhorar o precisão de reconhecimento de alvos acústicos submarinos.

Existe uma dificuldade de se obter grandes quantidades de dados (sinais acústicos de navios) para o desenvolvimento de classificadores supervisionados devido a diversos fatores, tais como: o custo do deslocamento de um navio para a coleta das informações e a repetição dos experimentos em diferentes condições operacionais. FERNANDES *et al.* [56] propõe o uso de Redes Adversárias Generativas (Gene-

native Adversarial Neural Networks - GAN) para a geração de amostras sintéticas que podem ser usadas no treinamento de modelos de aprendizado profundo para a classificação de sinais de sonar passivo. O conjunto de dados utilizado é o mesmo abordado por CERQUEIRA e DE SEIXAS [26]. Sobre cada gravação dos navios é calculado um lofargrama produzido pela análise de extração de características denominada análise LOFAR. Os lofargramas gerados passam por uma normalização e então, a partir deles, diversos sub-lofargramas são originados (esses sub-lofargramas também são imagens), porém, de tamanho $l \times 400$. Os sub-lofargramas são utilizados para o treinamento da GAN, que constituem uma classe de modelos que tentam gerar dados sintéticos com a mesma distribuição de probabilidade dos dados de treinamento [57]. O estudo foi abordado usando diferentes estratégias, máxima probabilidade e classe especialista. Uma alta precisão foi alcançada utilizando os dados experimentais, $99,0 \pm 0,4\%$. Esse resultado é superior a projetos anteriores de aprendizagem de máquina neste campo de estudo.

YUE *et al.* [58] classifica 16 tipos diferentes de alvos acústicos submarinos usando redes CNN e *Deep Brief Network* (DBN). Os modelos alcançaram precisões de até 94,75% e 96,96%, respectivamente. Os resultados foram comparados com métodos tradicionais de aprendizagem de máquina, SVM e Wndchrm [59]. Este último é originalmente uma ferramenta aplicada para a análise de imagem biológica. O software aplica transformadas em informações extraídas da análise LOFAR. Os resultados mostram que métodos de aprendizagem profunda podem alcançar maior precisão de reconhecimento ao classificar alvos subaquáticos a partir de seus ruídos de radiação.

CAO *et al.* [60] introduzem uma estrutura de classificação de alvos submarinos utilizando uma rede neural profunda para aprender características profundas a partir de uma grande entrada conjunta de múltiplos domínios. O trabalho propõe incorporar informações espectrais e de domínio wavelet com resoluções diferentes para compreender a estrutura “global” e a variação transitória “local” dos sinais brutos irradiados. As características extraídas de diferentes domínios são organizadas em um vetor de entrada. O vetor é composto pelos componentes periódicos de baixa frequência causados pelo giro da hélice, por uma característica logarítmica do espectro da frequência (*Log-Power Spectrum* - LPS) [61] e a energias dos componentes do pacote wavelet (*Wavelet Packet Component Energy* - WPCE) em várias sub-bandas [34]. Para aprender características mais discriminatórias para classificação é adotado um modelo de Autoencoder esparso empilhado (SSAE), que é composto de múltiplas camadas ocultas [62]. O modelo SSAE proposto é avaliado no conjunto de dados que consiste no sinal acústico subaquático recebido em diferentes profundidades oceânicas. Os resultados dos autores mostram que o modelo SSAE proposto com características de entrada conjunta alcançou uma melhoria de 5% na precisão de classificação em comparação com a abordagem DBN de última geração.

WANG *et al.* [30] propõe um método de fusão de decisão Dempster-Shafer (ML-DS) de aprendizagem de máquinas. O algoritmo combina os resultados de reconhecimento da aprendizagem de máquinas e *deep learning*. Ele usa a teoria da tomada de decisão baseada em evidências para realizar a fusão de características sob diferentes classificadores de redes neurais e melhorar a precisão. Primeiro, é utilizada uma rede CNN para extrair os recursos da imagem LOFAR bidimensional e classificá-los para obter os resultados de reconhecimento. A rede LSTM (*Long Short-Term Memory*) [22] é usada para extrair a característica de amplitude contínua do sinal e classificá-la para obter o resultado de reconhecimento. O algoritmo de aprendizagem de máquina SVM é usado para classificar as características de cromaticidade do ruído irradiado. De acordo com os resultados de classificação dos diferentes classificadores, um modelo básico de atribuição de probabilidade (Basic Probability Assignment - BPA) é projetado. A fusão de diferentes classificadores em nível de decisão nos resultados de predição é realizada através da teoria de evidência D-S. A Figura 2.11 mostra a fusão proposta pelo artigo. Vale ressaltar que o método propõe fusionar primeiramente os resultados dos modelos profundos e por fim, a taxa de precisão de reconhecimento após a fusão de aprendizado profundo é fundida com a probabilidade de reconhecimento média determinada pelo classificador do algoritmo de aprendizado de máquina para fazer o segundo método de fusão de decisão. Os resultados experimentais mostram que as duas fusões de aprendizagem profunda combinadas com uma fusão de aprendizagem de máquinas podem melhorar significativamente os resultados de classificação do conjuntos de dados de baixa SNR. O menor resultado de classificação da fusão pode chegar a 76,01% e a taxa média de classificação da fusão pode chegar a 94,92%. Em comparação com algoritmos tradicionais de classificação de característica única, a precisão de classificação melhora significativamente. O processo de classificação para este trabalho será descrito no Capítulo 4.

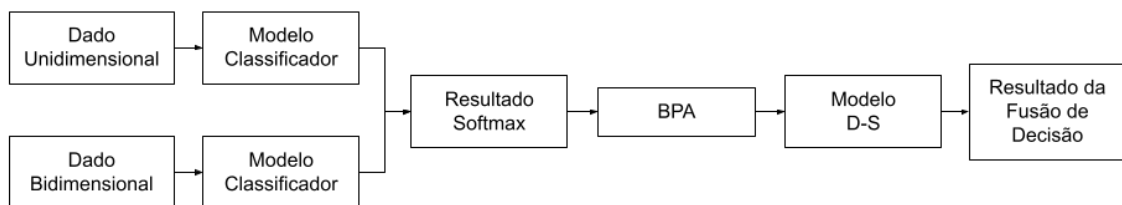


Figura 2.11: Fluxograma de fusão de decisão de diferentes tipos de dados.

Fonte: WANG *et al.* [30]

Capítulo 3

Máquinas de Comitê

Este capítulo tem por objetivo apresentar o conceito de Máquinas de Comitê e algumas formas de integrá-las. Possibilitando o entendimento sobre as formas de fusão dos modelos de redes neurais MLP, CNN e LSTM (*Long Short-Term Memory*) implementados neste trabalho.

Um comitê de máquinas é uma forma de aprendizado de máquina que pode ser aplicada em problemas de classificação e regressão. As abordagens de comitê de máquinas visam produzir ganhos de desempenho frente às propostas de soluções isoladas, ou seja, que não formam comitês. Elas são guiadas pelo conceito de *dividir para conquistar* [28]. Dado que as amostras de treinamento para uma RNA apresentam um número finito, em muitas das vezes, não é possível obter um modelo com máxima generalização.

Segundo BISHOP *et al.* [63], a implementação de comitê de máquinas está relacionada a alguns aspectos das RNA, tais como: Convergência do algoritmo de treinamento para um mínimo local, Risco de sobre-ajuste ou sub-ajuste e Complexidade do problema a ser solucionado. Caso aconteça sobre-ajuste ou sub-ajuste do modelo durante o processo de treinamento, o modelo apresenta um *erro de estimação*. A complexidade do problema a ser solucionado impacta diretamente na flexibilidade do modelo. Ocorre um *erro de aproximação* caso a flexibilidade do modelo não seja compatível com a complexidade do problema. Os dois erros citados acima impactam na generalização das RNA. O desenvolvimento de comitê de máquinas pode trazer os seguintes benefícios: redução na variância do modelo final, melhora na tolerância a ruído nos dados e melhora na capacidade de generalização [28]. As propostas para comitês de máquinas utilizadas neste trabalho são *ensemble* e mistura de especialistas [63]. No decorrer deste trabalho os modelos classificadores que compõem um *ensemble* serão chamados de componentes, e os modelos classificadores que compõem uma mistura de especialistas serão chamados de especialistas. As diferenças entre os dois serão descritas ao longo deste capítulo. Uma vantagem prática é que cada componente do *ensemble* e cada especialista da mistura não pre-

cisam apresentar uma flexibilidade compatível com a complexidade do problema. O conhecimento adquirido pelos modelos que o compõem chegam a uma solução global com melhor desempenho frente à obtida por qualquer um dos componentes atuando isoladamente.

Para entender porque esse conceito é tão importante para nós, vamos mencionar brevemente as características dos dados utilizados neste trabalho. Os sinais acústicos de navios adquiridos por meio de um sistema de sonar passivo submarino possuem características importantes que impactam no desenvolvimento de um classificador. Estamos tentando classificar um ruído de interesse de alta dimensionalidade em um ambiente altamente ruidoso. Além disso, vale ressaltar que as fontes de ruído comportam-se de maneira distinta para cada aquisição de amostras [15]. Considerando os problemas citados e a existência de uma grande quantidade de classes abordadas nesse trabalho, a construção de um único modelo para a classificação pode trazer problemas com relação a generalização.

Na abordagem *ensemble* cada componente representa isoladamente uma solução para o problema, podendo cada solução ser obtida por meios distintos e independentes entre si. Na mistura especialista a tarefa é decomposta em várias subtarefas e a solução final requer a contribuição de todos ou de um subconjunto dos especialistas. Sendo assim, um especialista isolado não representa uma solução para todo o problema e sim a composição de todos os especialistas [28, 47]. A arquitetura de uma Máquina de Comitê resume-se em duas partes: o conjunto dos membros e um integrador de suas saídas [64].

BISHOP *et al.* [63] e VILLANUEVA [28] demonstraram que o desempenho do *ensemble* pode ser superior ao de um único componente atuando de forma isolada. Considere um *ensemble* em que $y_i(x)$ seja a saída do i -ésimo componente, tal que $i = 1, 2, 3, \dots, M$ e $x \in X$, onde M é o número total de elementos treinados e X é o espaço de entrada. Seja $f(x)$ o valor correto a ser esperado para a entrada x e $\epsilon_i(x)$ é o erro associado. A função de mapeamento que cada componente realizará é:

$$y_i(x) = f(x) + \epsilon_i(x), x \in X \quad (3.1)$$

logo, o erro quadrático médio (EQM_i) para o componente i pode ser escrito como:

$$EQM_i = E[\{y_i(x) - f(x)\}^2] = E[\epsilon_i^2(x)], \quad (3.2)$$

onde a esperança matemática ($E[\cdot]$) corresponde a uma integral sobre x ponderada pela função densidade de probabilidade ($p(x)$) tal que:

$$E[\epsilon_i^2] = \int \epsilon_i^2(x)p(x) dx. \quad (3.3)$$

O erro quadrático médio cometido pelos M componentes (EQM_{me}) atuando individualmente pode ser estimado a partir da Equação (3.2)

$$EQM_{me} = \frac{1}{M} \sum_{i=1}^M EQM_i = \frac{1}{M} \sum_{i=1}^M E[\epsilon_i^2] \quad (3.4)$$

Sendo a integração da saída dos M membros o valor médio entre eles, a saída $y_{ensemble}(x)$ do *ensemble* é dada por:

$$y_{ensemble}(x) = \frac{1}{M} \sum_{i=1}^M y_i(x) \quad (3.5)$$

Sendo assim, o erro do *ensemble* ($EQM_{ensemble}$) é:

$$EQM_{ensemble} = E \left[\left(\frac{1}{M} \sum_{i=1}^M y_i(x) - f(x) \right)^2 \right] = E \left[\left(\frac{1}{M} \sum_{i=1}^M \epsilon_i \right)^2 \right] \quad (3.6)$$

Supondo que os erros $\epsilon_i(x)$ possuem média zero e são não correlacionados, resulta-se:

$$E[\epsilon_i] = 0, E[\epsilon_i \epsilon_j] = 0, \text{ se } j \neq i. \quad (3.7)$$

A partir das Equações (3.4), (3.6) e (3.7) é possível obter [28]:

$$EQM_{ensemble} = \frac{1}{M^2} \sum_{i=1}^M E[\epsilon_i^2] = \frac{1}{M} EQM_{me}, \quad (3.8)$$

ou seja, analisando a Equação (3.8) observamos que o erro do *ensemble* é menor que o erro médio dos membros do *ensemble* ($EQM_{ensemble} \leq EQM_{me}$), reduzido por um fator M . Entretanto, na prática, os membros estão correlacionados por serem treinados para o mesmo problema. Quando erros ocorrem no *ensemble*, a ocorrência deve se restringir ao mínimo número de membros. O aumento da independência estatística, chamada de diversidade, entre os membros do *ensemble* diminui o erro médio do *ensemble* até um certo ponto, pois o aumento demasiado também faz com que os membros do *ensemble* percam em exatidão [15]. Existem várias formas de tentar alcançar a diversidade, dentre elas temos [28]:

1. Inicialização aleatória dos pesos das conexões de cada RNA: espera-se que o processo de treinamento das redes convirja para mínimos locais distintos;
2. Variação da arquitetura da RNA: do número de camadas, neurônios das ca-

madras ocultas e função ativação;

3. Variação do algoritmo de treinamento: nesta etapa podem ser empregados algoritmos de otimização que buscam múltiplas propostas de otimização simultaneamente; e
4. Re-amostragem dos dados: Técnicas de re-amostragem como *Bagging* [65], *Boosting* [66] introduzem, também, diversidade nos componentes.

Outro algoritmo que envolve a manipulação dos dados de treinamento, o qual difere dos algoritmos de re-amostragem citados acima, é o *Random Forest* desenvolvido por BREIMAN [65] e trata-se de uma variação do *bagging*. Esse algoritmo é um *ensemble* de árvores de decisão, cujos parâmetros de treinamento variam aleatoriamente. Os parâmetros podem ser réplicas dos dados de treinamento, mas também podem ser um subconjunto de características diferentes [28]. O algoritmo *bagging* vem de *bootstrap AGGREGatING* ou agregação *bootstrap* proposto por BREIMAN [67]. É uma técnica de geração de conjuntos e treinamento a partir de um único conjunto, através de uma re-amostragem com reposição. Estes sub-conjuntos de treinamento serão usados para treinar cada membro que compõe o *ensemble*. Espera-se que o *bagging* gere membros mais independentes estatisticamente. O algoritmo desenvolvido por SCHAPIRE [68], o *boosting*, também cria vários conjuntos de treinamento por re-amostragem dos dados. Mas no *boosting*, a re-amostragem é gerada para fornecer dados de treinamento mais informativos para os componentes que são gerados de forma consecutiva. Em suma, as amostras classificadas como erradas em um membro recebem um peso maior no treinamento do próximo membro.

Após treinados os membros de um comitê, existem várias formas de combiná-los a fim de produzir uma única resposta a partir das respostas individuais. Essa tarefa depende da forma que o comitê foi treinado e da tarefa a qual ele deve resolver (regressão, classificação ao agrupamento de dados) [28]:

1. Em classificação de dados: técnicas baseadas em votação a partir de saída simbólica dos componentes. Voto simples, onde o resultado é indicado pela classe mais votada dos componentes. Integração do tipo o vencedor leva tudo (*the winner takes all*), onde a resposta do *ensemble* é a saída do membro com maior resposta;
2. Em regressão de dados: média simples ou a média ponderada das saídas numéricas dos componentes, onde os pesos podem ser estimados pelo erro de treinamento de cada membro; e
3. Em agrupamento de dados: envolvem técnicas mais elaboradas que nos casos anteriores, dado que cada componente apresenta como resposta um conjunto de grupos, pode ser usada uma matriz de co-associação.

No âmbito da classificação, além das maneiras descritas acima, existem outras formas de combinar modelos a fim de produzir uma resposta única a partir das respostas individuais. Podemos citá-las através de trabalhos publicados anteriormente.

YÜZKAT *et al.* [69] implementa um sistema que indica duas possibilidades de votação entre os resultados de cada modelo, chamadas de votação forçada (*hard-voting*) e votação suave (*soft-voting*). A votação forçada usa a previsão de cada rede como valor para a eleição da melhor escolha, por exemplo: dados quatro modelos (A, B, C e D), onde os três modelos indiquem uma classificação X e um modelo indique Y, a classificação final será X, já que possui três votos contra um. A segunda técnica é a votação suave, neste caso são somadas todas as probabilidades encontradas pelos modelos para cada uma das classes e a classificação final será dada pela classe que possuir o maior valor. O trabalho implementa e funde, através das técnicas *hard-voting* e *soft-voting*, seis modelos diferentes de redes convolucionais para automatizar a classificação morfológica de imagens de espermatozoides. O desempenho da proposta é avaliado com base em três conjuntos de dados de morfologia espermática publicamente disponíveis, SMIDS, HuSHeM e SCIAN-Morpho. As acurácias de classificação obtidas foram, respectivamente, 90,73%, 85,18% e 71,91%, usando a fusão baseada em votação suave. Os resultados sugeriram que a abordagem proposta poderia classificar automaticamente, bem como alcançar alto sucesso em três conjuntos de dados diferentes.

MELO *et al.* [70] apresenta um novo método de detecção de nuvens visíveis de vapores de hidrocarbonetos que é constituído de redes neurais convolucionais, as quais são aplicadas para classificar imagens (*frames* do circuito fechado de televisão) como pertencentes às classes com ou sem vazamento de gás natural em cabeças de poços *onshore*. Para isso, o estudo propõe e apresenta o resultado de três arquiteturas de redes neurais além de utilizar redes pré-treinadas (DenseNet-201, GoogLeNet, MobileNetV2 e ResNet18) e aplicar a técnica de *Feature Fusion*¹ aos modelos pré-treinados. Os resultados experimentais mostraram que o modelo de melhor desempenho apresentou acurácia de 100,00% e taxa de falsos negativos de 0,00%, superando o desempenho de métodos anteriores de detecção de vazamento de gás natural por imagens de circuito fechado de televisão.

SARAIVA *et al.* [72] investiga o uso de fusão de dados clínicos de pacientes com informações extraídas de imagens de raio-x no contexto de classificação de Covid-19. Durante o desenvolvimento do trabalho foram investigadas duas abordagens de fusão, em nível de características e em nível de decisão. A fusão em nível de características busca fundir características extraídas de imagens de raio-x utilizando

¹A técnica de *Feature Fusion* é usada para fundir os recursos extraídos de imagens e os recursos de dados numéricos, recursos além da imagem, do alvo para aumentar a representatividade dos dados do alvo [71].

uma rede convolucional. O modelo utilizado foi a VGG-16, pré-treinada com o conjunto de dados ImageNet. Em seguida houve uma transferência de aprendizado para uma outra rede, a qual foi retreinada com os dados de Covid-19. O vetor de características da imagem é concatenado com os dados clínicos em um único vetor. Esse vetor é utilizado no treinamento de uma rede neural densa para classificação das instâncias. A rede densa possui uma camada oculta com 100 neurônios e uma camada de saída com 1 neurônio (A classificação é Covid-19 vs não-Covid-19). A segunda abordagem divide o processo em três etapas: extração de características tanto das imagens de raio-x quanto das notas clínicas; treinamento de um modelo de classificação para cada grupo de características e uso de uma função de fusão para obter uma única decisão a partir das decisões fornecidas pelos dois modelos de classificação obtidos. Os experimentos mostram que a fusão de dados melhora o desempenho do sistema em relação ao uso de classificadores individuais. Além disso, mostra-se uma superioridade da fusão em nível de decisão sobre a abordagem de fusão em nível de características.

ZHANG *et al.* [24] propôs uma rede neural integrada para o reconhecimento de alvos acústicos subaquáticos através do aprendizado de características da fusão. Esse método é proposto para tentar contornar a dificuldade em extrair características dos dados devido à alta intensidade do ruído de fundo em ambientes marítimos e a inevitável perda de dados causada pela extração de características de algumas tecnologias (DEMON, LOFAR e Mel frequency cepstrum coefficient - MFCC). A rede neural integrada (três redes CNN) foi projetada e treinada com três características diferentes dos sinais acústicos submarinos, a amplitude do espectro da transformada de Fourier (Short-time Fourier transform, STFT), a fase do espectro e a característica de *bispectrum*. A saída de cada rede CNN é dada por uma camada *Softmax* e a fusão é realizada em nível de decisão, após a camada de saída. Por fim, o algoritmo *shuffled frog leaping algorithm* (SFLA) é utilizado para treinar os coeficientes de peso de diferentes redes. O SFLA é um método de cálculo evolutivo que combina métodos determinísticos e métodos aleatórios. A Figura 3.1 mostra o fluxograma de montagem do processo de treinamento e teste da rede integrada. Os resultados experimentais mostram que o método de rede neural integrada baseado na fusão de características tem maior precisão de reconhecimento em relação a métodos tradicionais.

Em KOMARI ALAIE e FARSI [29] é proposto um novo método para a detecção de alvos em sonares passivos usando limiar adaptativo, o qual pode ser observado na Figura 3.2. Neste método, cada som é dividido em som ambiente (que é chamado de ruído ambiente) e som do alvo. Estes sons são comparados pelo ruído ambiente registrado (sinal de treino) e as suas semelhanças são determinadas numericamente por pontos de detecção. Para reduzir o ruído de entrada, é aplicado um filtro medi-

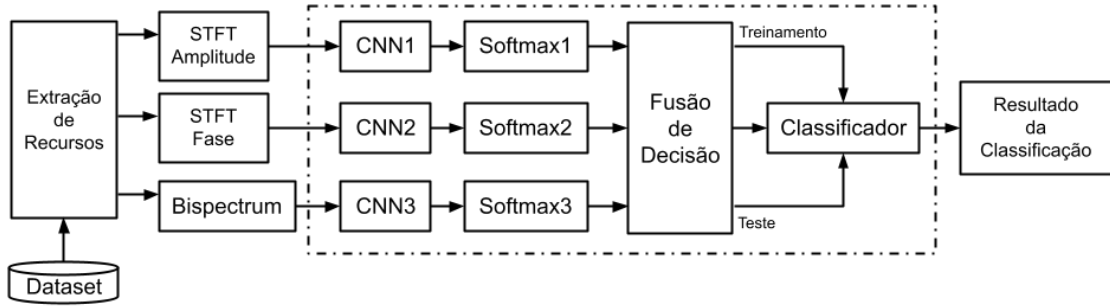


Figura 3.1: Diagrama esquemático da rede neural integrada.

Fonte: ZHANG *et al.* [24]

ano. A classificação utilizada para classificar o sinal de ruído em ambos os domínios, tempo e frequência, é Bayesiana e a distribuição posterior é estimada pelo algoritmo de Máxima verossimilhança. Ressalta-se que os sinais antes de serem classificados, passam por um filtro passa baixa e, após a classificação, os pontos de detecção são calculados através da média. Por fim, o alvo é detectado pela combinação dos pontos de detecção em ambos os domínios usando o filtro adaptativo LMS (*Least Men Square*). Os resultados mostraram que o método proposto melhorou a taxa de detecção real em cerca de 24% quando comparado a outro método de detecção.

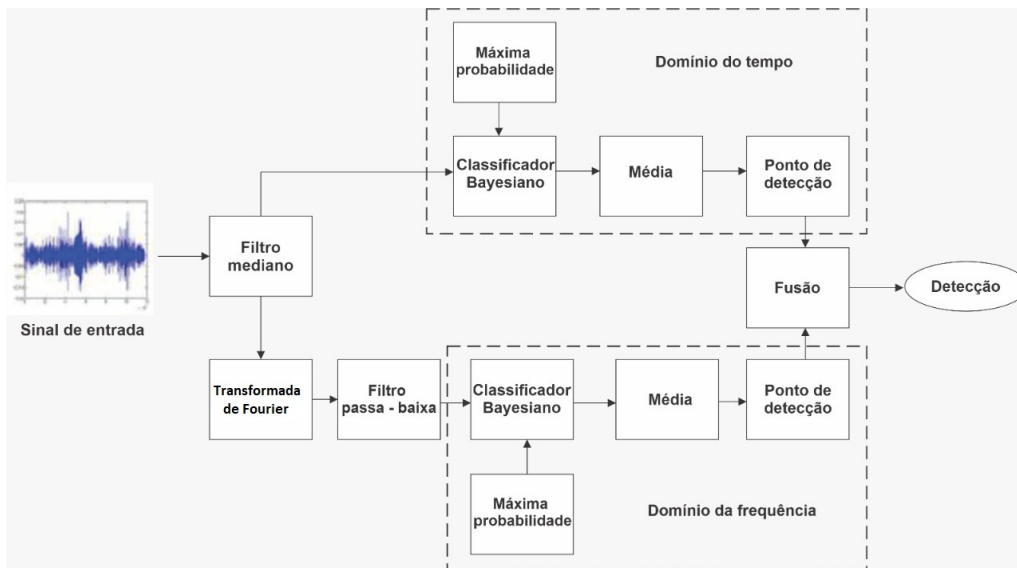


Figura 3.2: Diagrama em blocos do método de fusão com Limiar Adaptativo.

Fonte: KOMARI ALAIE e FARSI [29]

Capítulo 4

Método Proposto

Este capítulo descreve o método que será utilizado durante este trabalho. O método é baseado nos trabalhos de SOARES FILHO [21] e SOUZA FILHO e DE SEIXAS [25], os quais exploraram uma eficiente forma de processamento e compactação dos dados. Este processamento serviu de base para GOLTZ [15] implementar uma classificação hierárquica baseada em conhecimento especialista, considerando que estratégias usando comitês de redes neurais têm sido empregadas com sucesso em aplicações com dados de alta dimensionalidade e grande presença de ruído, além de classes desbalanceadas. A Figura 4.1 apresenta de forma macro o método que será seguido. Em suma, o método tem por objetivo:

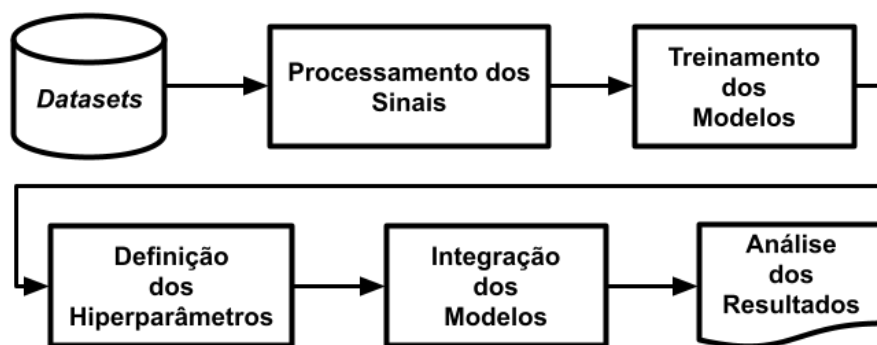


Figura 4.1: Diagrama em blocos do método proposto.

- Realizar o processamento dos sinais acústicos do sistema de sonar passivo, utilizando as análises LOFAR [21] e Temporal [19];
- Implementar e treinar três modelos de redes neurais, MLP, CNN e LSTM com os dados dispostos de forma hierárquica [15];
- Definir os melhores hiperparâmetros para cada modelo, de acordo com os resultados obtidos durante o treinamento;

- Fusionar os melhores modelos de classificação alimentados pelas análises Temporal e LOFAR; e
- Avaliar os possíveis ganhos dos modelos compostos pelas análises Temporal e LOFAR.

4.1 Classificador Multi-Classes e Hierárquico

Este trabalho utiliza-se do comitê de classificadores disposto de forma hierárquica desenvolvido por GOLTZ [15], o qual é disposto da seguinte forma: No primeiro nível (mais alto), denominado de “Super”, foi implementado um classificador capaz de distinguir todas as classes do problema, porém reagrupadas em novas classes. As novas classes são compostas por elementos das classes do problema original, os elementos foram agrupados usando como critério navios que possuem características operativas em comum. O segundo nível da hierarquia é composto por novos classificadores criados a partir da saída do classificador do nível hierárquico acima. Cada elemento (classificador) da hierarquia é treinado para distinguir as classes que o compõem. Esse processo de treinamento é idêntico para o segundo nível, porém, caso não haja características em comum relevantes entre as classes, o elemento da hierarquia é treinado para distinguir entre as classes originais do problema. Essa lógica proposta se propaga para todos os níveis hierárquicos inferiores até que não haja mais características em comum entre as classes originais, ou seja, o último nível de classificação identifica as classes originais do problema.

Vale ressaltar que as características em comum entre as classes são advindas de conhecimento especialista do IPqM e não podem ser divulgadas por questões de confidencialidade, porém não causam impacto na pesquisa. Esse conhecimento possibilitou a criação das novas classes do comitê além de sua disposição hierárquica [15]. Cada nível representa um grau de detalhamento dos navios. Por exemplo, um classificador poderia ser especializado em uma Fragata (navio de guerra de emprego geral ou apenas antissubmarino), e o nível abaixo poderia classificar o tipo e propulsão, diesel ou gás. A Figura 4.2 apresenta a disposição dos níveis hierárquicos para o conjunto de dados de águas profundas. Os níveis representados são: “Super”, A, B, C, D, AA, AB, AC, DA e DB.

A Figura 4.3 apresenta a disposição dos níveis hierárquicos “Super”, A, B e B1 a B8 para o conjunto de dados provenientes do VIPP (águas rasas).

A disposição proposta por GOLTZ [15] dispõe das seguintes regras de formação: Cada nível do comitê hierárquico é treinado com as classes que o compõem e é chamado de membro do comitê. A ativação do comitê dá-se pelo membro “Super” após o treinamento de todos os membros. A saída de um membro identifica qual o

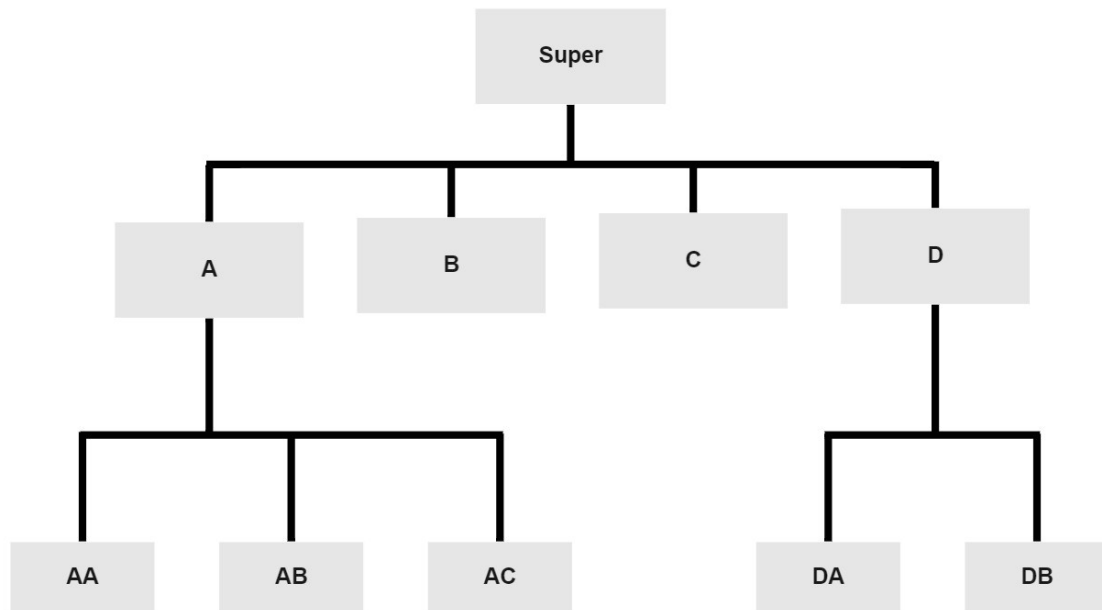


Figura 4.2: Comitê hierárquico do conjunto de águas profundas oriundo de sinais de submarinos.

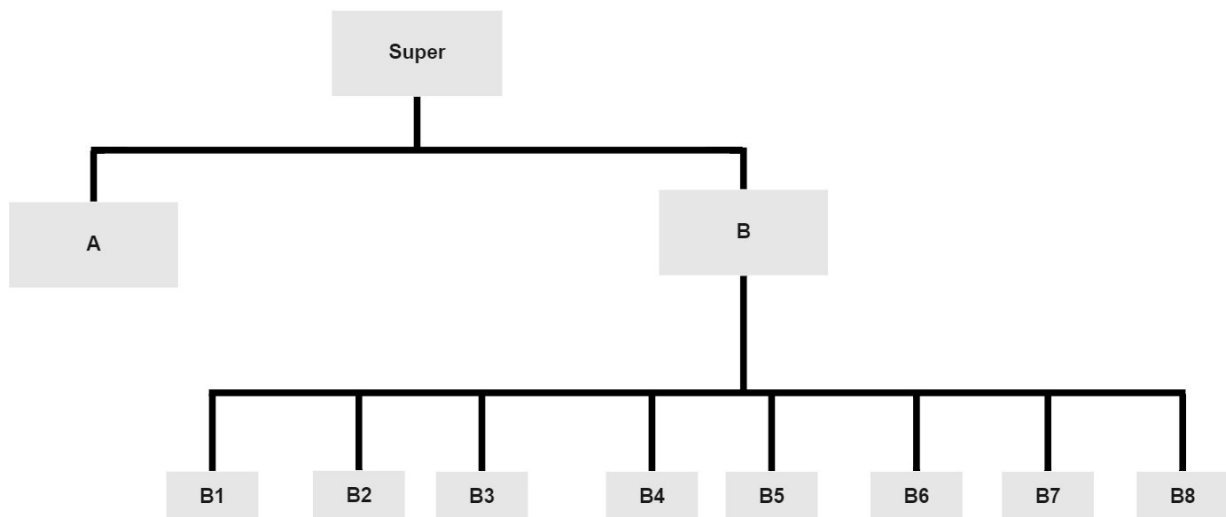


Figura 4.3: Comitê hierárquico do conjunto de águas rasas oriundo do VIPP.

próximo membro a ser ativado com a mesma entrada do nível acima. A ativação se propaga para níveis inferiores até a resposta das classes originais de um membro.

4.2 Processamento dos Sinais

As gravações, ou seja, os arquivos de áudio das corridas de cada classe são processadas pelas análises LOFAR e Temporal, conforme descrito no Capítulo 2. Os diagramas de blocos das duas análises estão descritos mais uma vez nas Figuras 4.4 e 4.5.

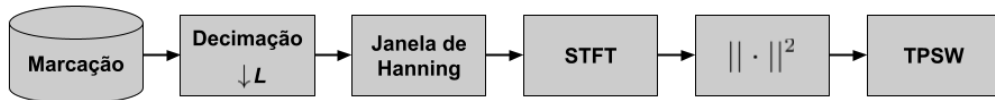


Figura 4.4: Diagrama de blocos da Análise LOFAR.

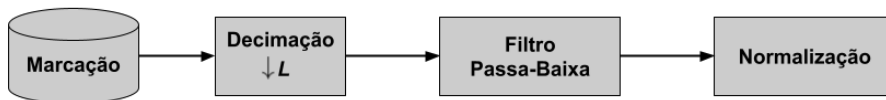
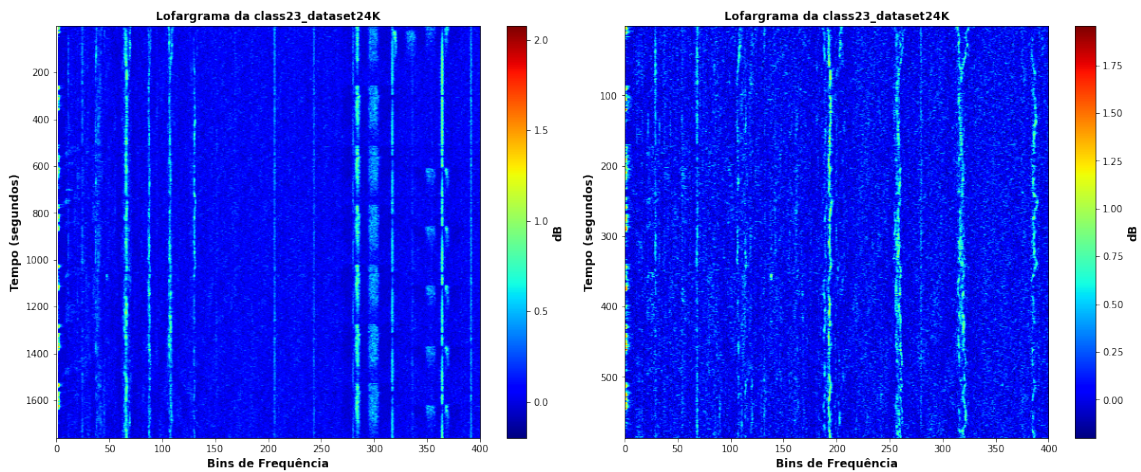


Figura 4.5: Diagrama em blocos da Análise Temporal.

Vale ressaltar que os sinais do conjunto de águas profundas estão em taxas de amostragem distintas (os valores variam de 22.050 até 48.000 Hz), o que dificulta a comparação dos resultados e união das corridas. Para contornar esse problema, inicialmente os sinais foram reamostrados de acordo com a menor taxa de amostragem apresentada no *dataset* (22.050 Hz). O processo de reamostragem consiste em uma sub-amostragem e sobre-amostragem por fatores que advém da divisão entre as taxas de amostragem de dois sinais distintos. O resultado da divisão entre as taxas é convertido para um número racional, enquanto o numerador é o fator responsável pela sobre-amostragem, o denominador é o fator responsável pela sub-amostragem. Sendo assim, o sinal que se deseja modificar é sobre-amostrado por fator inteiro, um filtro passa-baixa é aplicado e, em seguida, o sinal é sub-amostrado pelo outro fator. Convertendo então a sua taxa de amostragem à taxa desejada. Esse procedimento não se fez necessário para o conjunto de águas rasas, pois todas as corridas apresentam a mesma taxa de amostragem.

Os parâmetros da análise LOFAR foram os mesmos utilizados por vários trabalhos como em DE MOURA JUNIOR [8]. Entretanto, devido a especificidade dos conjuntos de dados utilizados, ainda foi necessária uma busca paramétrica para alguns parâmetros. Esta análise primeiro decimou o sinal dos dois conjuntos de dados (águas profundas e águas rasas) por um fator L , o que reduz a faixa de frequência

do sinal. Como mencionado anteriormente na Seção 2.2.2 do Capítulo 2, essa etapa é feita para remover informações de frequência indesejáveis e restringir a faixa de frequência de interesse, pois geralmente a informação de interesse nos sinais de sonar passivo está concentrada em baixas frequências [73]. A Figura 4.6 mostra, através do lofargrama, os efeitos do uso da decimação da análise LOFAR. Comparando as Figuras 4.6a e 4.6b, pode-se observar que o não uso da decimação apresenta estrias mais definidas e pouco ruído, entretanto, o emprego da decimação foi necessária para reduzir a quantidade de dados, acarretando em um tempo de processamento menor ao utilizarmos as redes neurais. Em seguida, é aplicado ao sinal um filtro digital passa-baixa de ordem R . Projetado com frequência de corte $f(Hz)$. O próximo passo é aplicar ao sinal uma STFT. Com base no estudo feito em [21], foi realizada uma variação do tamanho da janela da STFT para extrair o espectro do sinal, os valores foram 1024 e 2048. Vale evidenciar que esses valores foram utilizados com sucesso em trabalhos anteriores [8, 15, 21], onde o processamento empregado no sinal é o mesmo usado por SOARES FILHO [21]. Por fim, o módulo do espectro passa por um filtro TPSW para a redução da amplitude do ruído de fundo, no qual foi ajustada a largura da fenda central e largura da fenda lateral [74].



(a) Sem o uso de decimação.

(b) Índice de decimação igual à 3.

Figura 4.6: LOFARgrama para diferentes índices de decimação.

Importante destacar que o tamanho da janela de processamento tem impacto direto na estacionaridade do sinal, quanto maior a janela, menor a probabilidade de o sinal ser estacionário dentro do intervalo. Sendo assim, umas das estratégias para atacar o problema da não-estacionariedade é reduzir o tamanho da janela de processamento [8]. A Figura 4.7 mostra através do lofargrama os efeitos do tamanho da janela na análise LOFAR. Observa-se que a diferença entre os espectros obtidos é relativamente pequena, o que justifica a utilização de qualquer um dos valores para classificação.

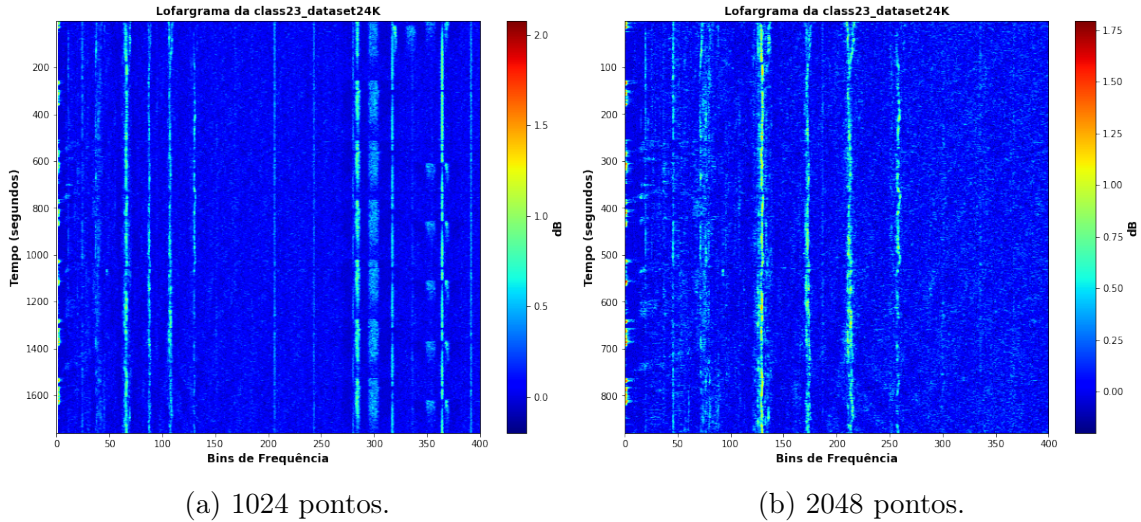
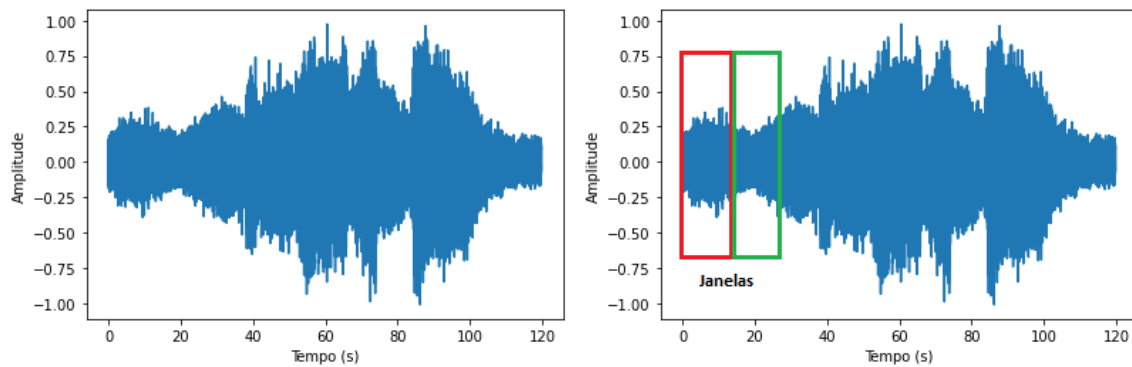


Figura 4.7: LOFARgrama para quantidades diferentes de pontos na janela de processamento.

É a primeira vez que um estudo se propõe utilizar a análise temporal nos conjuntos de águas profundas e rasas, mas o parâmetro L de decimação utilizado na análise LOFAR foi replicado nesta análise. Após a etapa de decimação, é aplicado um filtro passa-baixa com as mesmas especificações mencionadas na análise LOFAR e por fim, é realizada uma normalização na amplitude do sinal pelo seu mínimo e máximo.

Assim como mencionado anteriormente, é preciso segmentar o sinal para tentar obter um processo o mais localmente estacionário possível. Portanto, também foi feita uma variação com os valores de 1024 e 2048 pontos para o tamanho da janela e o índice SP (descrito na Seção 4.4.2). Esse foi o critério de avaliação para a definição do melhor tamanho. A Figura 4.8 ilustra como é realizado esse janelamento.



(a) Sinal de áudio de uma corrida. (b) Sinal de áudio com as janelas de aquisição.

Figura 4.8: Sinais de áudio com a representação das janelas de interesse.

4.3 Modelos de Redes Neurais

Os modelos de redes neurais abordados neste trabalho podem ser vistos com mais detalhes no Apêndice B. Para as amostras provenientes da análise LOFAR, a rede MLP tem como entrada um vetor com uma dimensão que corresponde ao número de *bins* do espectro do sinal. Já o vetor de entrada da rede MLP dos dados provenientes da análise temporal possui o número de atributos associados a quantidade de amostras por janelas do sinal no domínio do tempo. Dois valores distintos foram testados, 1024 e 2048. Ressaltando que não houve sobreposição (*overlap*) das amostras e o desempenho da rede foi analisado usando como base o índice SP descrito na Seção 4.4.2 para definir qual valor obteve os melhores resultados. As redes convolucionais são comumente utilizadas em dados com mais de uma dimensão (vide o Apêndice B.1.2), entretanto, neste trabalho as redes serão alimentadas com amostras unidimensionais provenientes das análises LOFAR e Temporal. Por esse motivo, pode-se substituir a convolução bidimensional realizada na camada convolucional por uma convolução unidimensional, operando com vetores ao invés de matrizes.

O número de neurônios da camada oculta da rede MLP, de filtros da camada CNN e de células da camada LSTM são definidos de acordo com a quantidade que apresente o melhor índice SP no processo de treinamento (índice calculado com o uso do conjunto de validação). O índice SP está detalhado na Subseção 4.4.2. As camadas de saída de todos os modelos são maximamente esparsas [20], em outras palavras, significa que a quantidade de neurônios corresponde à quantidade de classes do problema. Cada neurônio da camada de saída é associado a uma classe. Nos neurônios é utilizada uma função *Softmax* [22], ela transforma as saídas de cada neurônio em valores entre 0 e 1 e também divide pela soma das saídas. Isso essencialmente dá a probabilidade de a entrada estar em uma determinada classe. A maior probabilidade define a classe do problema.

A arquitetura proposta difere para cada modelo:

MLP: Composta por uma camada de entrada, uma de saída e uma camada intermediária (oculta). Os números de neurônios na camada oculta (n) foram variados até encontrar os melhores parâmetros para generalização da rede, sendo $n = 10, 20, \dots, 100$.

CNN: Esta arquitetura é composta por uma camada de entrada, uma de saída e entre elas uma estrutura típica das redes CNN, a qual possui diversos parâmetros que devem ser observados. Estes parâmetros podem ser vistos com detalhes no Apêndice B.1.2. Em suma, o modelo pode ser segmentado em duas partes: a primeira destinada à extração de características do sinal e a segunda é uma rede neural clássica MLP. Dentro da extração de características,

os principais elementos da rede são os filtros de cada camada convolucional e o *Kernel* utilizado para a convolução. A informação resultante desta convolução é chamada de *feature maps*. Essa informação é o vetor de entrada da rede totalmente conectada (MLP). Na etapa de extração de características houve uma variação de uma a duas camadas convolucionais testando 8 e 16 filtros com o *Kernel* de 3 e 6. Já na camada totalmente conectada, o número de neurônios testados foram 10, 20 e 30. Sendo assim, essas combinações resultaram em um total de 30 modelos.

LSTM: O número de células na camada LSTM variam de 25 a 100 com o passo de 25 células. Deste modo, a arquitetura é composta por uma camada de entrada, uma camada LSTM e uma camada de saída.

4.4 Treinamento da Rede Neural

O algoritmo desenvolvido neste trabalho foi implementado na linguagem de programação Python com o auxílio da biblioteca Keras [75]. O processo de treinamento supervisionado dos modelos visava à maximização da eficiência de classificação das classes através da minimização do erro de classificação da saída do modelo. Durante o processo foi usada a função de perda “Entropia cruzada”. As entradas foram normalizadas para evitar problemas durante a modelagem, em virtude da grande diferença na escala dos dados. O otimizador usado foi o Adam [22] com taxa de aprendizado η e as taxas de decaimento para as estimativas do primeiro e segundo momento são β_1 e β_2 . Estes são os valores padrões para o Adam no Keras [75]. Ao longo do desenvolvimento não houve a necessidade de modificá-los.

Como mencionado no Capítulo 3, o conjunto de dados utilizado é desbalanceado. Para solucionar esse problema empregou-se no treinamento das redes neurais um fator de ponderação no gradiente. Esta solução é baseada na proposta estabelecida por SOUZA FILHO e DE SEIXAS [25] para classificadores classe especialista. A ponderação de classe ajusta a função de custo do modelo para que a classificação incorreta de uma observação da classe minoritária seja mais penalizada do que a classificação incorreta de uma observação da classe majoritária. Essa abordagem pode ajudar a melhorar a precisão do modelo reequilibrando a distribuição de classes. No entanto, é importante observar que a ponderação de classe não cria novos dados e não pode compensar a falta de dados. Os pesos para cada classe podem ser calculados utilizando sua frequência, dessa forma, as classes minoritárias receberam mais pesos durante o treinamento. A Equação (4.1) demonstra como foram calculados os pesos de cada classe.

$$w_{classe} = \frac{Total\ de\ Amostras}{Quantidade\ de\ Classes \times Total\ de\ amostras_{classe}}, \quad (4.1)$$

onde, w_{classe} é o peso para cada classe e é alcançado dividindo o total de amostras do conjunto de dados pelo produto da quantidade de classes com o total de amostras da classe desejada.

Para obter maior generalização no conjunto de dados e evitar *overfitting*, alguns modelos foram treinados utilizando uma técnica chamada de *dropout*, a qual consiste em eliminar aleatoriamente (e temporariamente) alguns neurônios ocultos na rede, deixando os neurônios de entrada e saída intocados [22]. Além disso, foi utilizado também a parada prematura, ou *early stopping*, para reduzir os tempos de treinamento, ou seja, durante o processo de treinamento o erro do conjunto de validação é monitorado e caso ele não apresente melhoras dentro de um número específico de épocas, o processo de treinamento é interrompido e os melhores parâmetros do treinamento são restaurados, obtendo assim o melhor modelo.

De forma a validar o aprendizado das arquiteturas propostas foi feito o treinamento usando a validação cruzada com *k-folds* [47].

4.4.1 Validação Cruzada

Dado um conjunto de dados limitado, o procedimento de validação cruzada nos permite usar todos os exemplos disponíveis no conjunto de dados na estimativa do erro médio de validação e teste, ao preço de um aumento do custo computacional [22]. Foi utilizado um método usual de validação cruzada conhecido como *k-fold* baseado na técnica de *jackknife* [76]. Nele os dados são subdivididos em n subconjuntos (também chamados de *folds*) disjuntos e de tamanhos iguais que serão utilizados para o treinamento e teste de um modelo. O conjunto de treinamento é composto por $n-1$ desses subconjuntos e o conjunto restante é usado apenas para a avaliação do modelo. Um dos subconjuntos que fazem parte do conjunto de treinamento é destinado para compor o subconjunto de validação. O subconjunto de validação é responsável por avaliar o treinamento e auxiliar na estimativa dos melhores parâmetros para o modelo. Todo esse processo é repetido n vezes, de forma que o conjunto de validação e de teste flutue entre todas as partições do conjunto de dados original, mas lembrando que o subconjunto de teste não participa da etapa de treinamento e o subconjunto de validação não auxilia no ajuste dos parâmetros. Ao término deste processo, a estatística associada ao processo pode ser avaliada através das figuras de mérito citadas na Subseção 4.4.2. Na Figura 4.9 temos uma representação gráfica do método de validação cruzada.

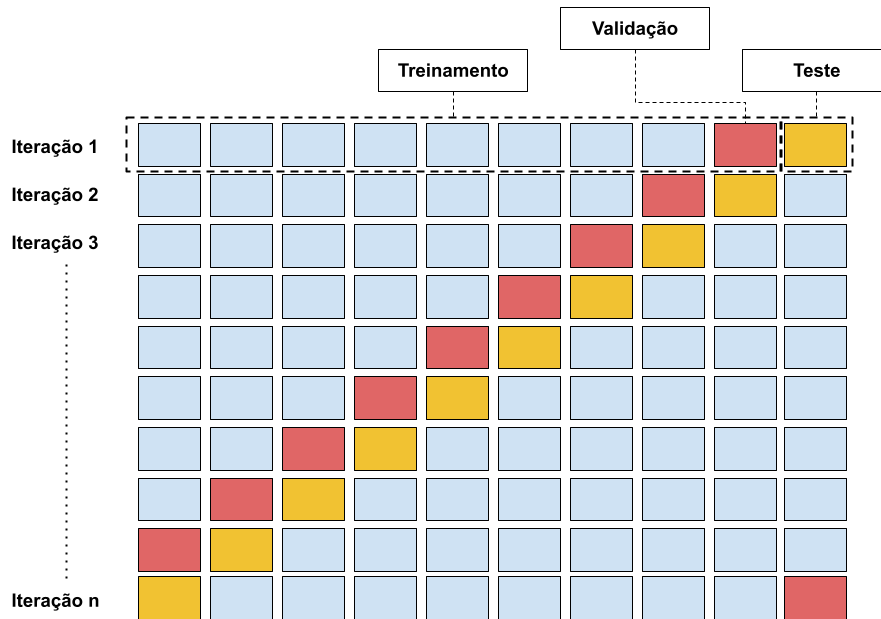


Figura 4.9: Processo de validação cruzada com 10 subconjuntos.

4.4.2 Figuras de Mérito

São utilizadas diversas medidas de desempenho para avaliar a eficiência dos métodos de classificação dos classificadores. A principal figura de mérito utilizada foi o índice Soma-Produto (SP), que será descrito abaixo. Além dele, outras figuras clássicas na literatura no âmbito da inteligência artificial foram usadas, como acurácia (A_{cc}), precisão (P_{re}) e *recall* (R_{ec}). A acurácia indica uma performance geral do modelo. Ela informa quantas classificações o modelo classificou corretamente, dentre todas as classificações. A precisão tem por objetivo informar dentre todas as classificações de classe Positivo que o modelo fez, quantas estão corretas. Ela pode ser usada em uma situação em que os Falsos Positivos são considerados mais prejudiciais que os Falsos Negativos. O *Recall* estima dentre todas as situações de classe Positivo como valor esperado, quantas estão corretas. É usado em uma situação em que os Falsos Negativos são considerados mais prejudiciais que os Falsos Positivos.

O índice SP foi desenvolvido para avaliar de forma equilibrada a eficiência dos classificadores. Para problemas que envolvem duas ou mais classes, ele se torna bastante útil. Sua eficiência advém do fato de que o índice tende a colapsar quando pelo menos uma das classes apresenta uma baixa eficiência. O índice será próximo de 1 caso todas as classes obtiverem altos valores de eficiência. Dado que o índice SP é calculado em função da eficiência de classificação, precisamos antes entender como funciona essa eficiência para depois apresentarmos a equação do índice SP e seus parâmetros. Suponha uma problema de N classes, a Eficiência de Classificação Eff_i para a i -ésima classe é calculada de acordo com a Equação (4.2).

$$Eff_i = \frac{\text{eventos}_i \text{ classificados corretamente da classe}}{\text{total de eventos}_i \text{ da classe}} \quad (4.2)$$

A eficiência de classificação é a quantidade de eventos da classe que foram classificadas corretamente. Seu resultado varia entre zero e um conforme o classificador erra ou acerta a classificação dos exemplos da i -ésima classe. A Equação (4.3) apresenta como é calculado o índice SP [17, 25, 35]. Durante o processo de treinamento dos modelos esse índice foi monitorado e serviu como critério para a seleção do melhor modelo de cada arquitetura apresentada na Seção 4.3.

$$SP = \sqrt{\left(\frac{\sum_{i=1}^N Eff_i}{N}\right) \cdot \left(\sqrt[N]{\prod_{i=1}^N Eff_i}\right)} \quad (4.3)$$

Nas Equações (4.4), (4.5) e (4.6) podemos observar as outras figuras de mérito clássicas citadas anteriormente. Onde TP (*true positive*) é a classe detectada corretamente, FP (*false positive*) é quando o modelo previu o pertencimento do sinal a classe erroneamente, TN (*true negative*) é a classe falsa atribuída corretamente, e FN (*false negative*) é quando o modelo previu o não pertencimento do sinal a classe erroneamente [77]. Como este trabalho aborda duas ou mais classes, o resultado apresentado pelas figuras de mérito (acurácia, precisão e *recall*) do modelo é a média não ponderada do valor encontrado para cada classe. Isto posto, o desequilíbrio entre as classes não é levado em consideração.

$$A_{cc} = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.4)$$

$$P_{re} = \frac{TP}{TP + FP} \quad (4.5)$$

$$R_{ec} = \frac{TP}{TP + FN} \quad (4.6)$$

4.5 Fusão dos Modelos

Este trabalho utiliza dois processamento distintos para os sinais do sistema de sonar passivo. O primeiro é realizado através da análise LOFAR, e o segundo é aplicando a chamada Análise Temporal. Ambas as análises são descritas com detalhes no Capítulo 2, nas Subseções 2.2.2 e 2.2.3. Esses processamentos combinados com di-

ferentes modelos e redes neurais geram classificadores com diferentes visões acerca do sinal de entrada. Essas visões são fusionadas a fim de melhorar as estimativas de classificação se comparadas as classificações provenientes dos classificadores de forma individual. No primeiro momento, os modelos individuais são treinados de maneira isolada pelos dados processados, como os modelos são diferentes eles desenvolvem recursos de classificação diferentes. Ao longo deste trabalho chamaremos os modelos treinados com os dados da análise Temporal de modelo temporal e os modelos treinados com os dados da análise LOFAR de modelo LOFAR. Na etapa final, a classificação individual de cada modelo é fusionada. No Capítulo 3 foram abordadas diferentes formas de integração e fusão, entretanto algumas delas não são aconselháveis para as especificidades deste trabalho. Por exemplo, dado que apenas dois modelos serão utilizados, o voto simples não é aplicável, pois não há um terceiro modelo para desempatar a classificação. O voto ponderado também pode trazer problemas dado que em uma aplicação real não se tem certeza qual modelo deve ter maior peso para a classificação. Sendo assim, a fusão acaba se tornando a melhor abordagem para esse problema.

A Figura 4.10 apresenta o esquema proposto para a fusão dos modelos. Nela é possível observar que os sinais são processados pelas análises LOFAR e Temporal. O resultado de cada análise alimenta um modelo de rede neural, o qual pode ser qualquer modelo entre, MLP, CNN e LSTM (descritos com detalhes no Apêndice B.1). Essa composição de processamento mais modelo pode ser chamada de “visão” a cerca de um domínio. Cada célula de comitê hierárquico dos dados da Seção 4.1 é composto pelas visões fusionadas.

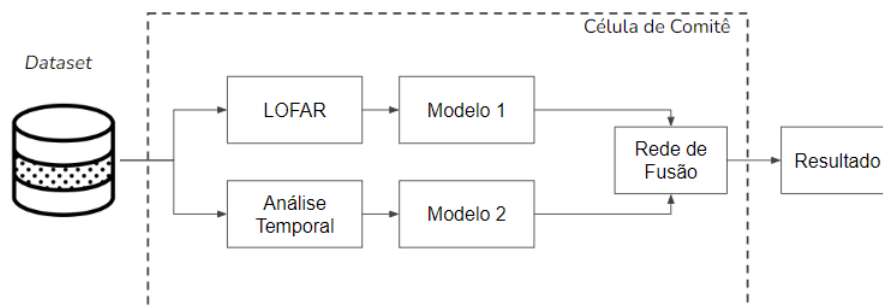


Figura 4.10: Modelo de classificação com rede de fusão para um elemento da célula de comitê.

O modelo proposto é composto por uma rede MLP, a qual chamamos de rede de fusão. A entrada da rede de fusão é composta pelos modelos LOFAR e temporal (sem a camada de saída), ou seja, as entradas da rede de fusão são as representações da camada oculta imediatamente anterior à camada de saída dos modelos LOFAR e temporal. A exceção é quando utilizamos a rede CNN, nela são retiradas a camada

fully connected e a saída. Desse modo, as entradas da rede de fusão são as representações da última camada convolucional. A rede de fusão possui uma camada de saída com o número de neurônios (com funções de ativação *Softmax*) correspondente à quantidade de classes do problema e uma única camada oculta. Algumas etapas são necessárias para entender a montagem do modelo proposto (vide Figura 4.11), antes de iniciar o processo de treinamento da rede de fusão:

- É retirada a camada de saída dos dois modelos (temporal e LOFAR), cujos parâmetros e hiper-parâmetros são estabelecidos durante o treinamento individual dos modelos (MLP, CNN e LSTM).
- As representações desses dois modelos são concatenadas e servem de entrada para a rede de fusão.
- Os parâmetros (pesos e *biases*) dos modelos LOFAR e temporal são congelados, ou seja, esses valores não são alterados durante o treinamento da rede de fusão.

As combinações dos modelos LOFAR e temporal que serão implementadas são: MLP com MLP, CNN com CNN, MLP com LSTM e CNN com LSTM. Com o intuito de avaliar o contexto da fusão sem a rede de fusão, foi implementado também um modelo que concatena as representações dos modelos LOFAR e temporal e as conecta diretamente a uma camada de saída.

O processo de treinamento da rede de fusão, semelhante ao descrito na Seção 4.4, é realizado no novo modelo proposto, o qual é alimentado pelos dados oriundos das análises LOFAR e Temporal. A variação da quantidade de neurônios na camada oculta da rede de fusão é 10, 30, 50 e 60 quando os modelos LOFAR e temporal não utilizam modelos profundos, e quando os modelos profundos são utilizados a variação do número de neurônios da rede de fusão é 10, 30, 50 e 100. Vale ressaltar que durante o treinamento da rede de fusão, as outras camadas antes das redes não são retreinadas.

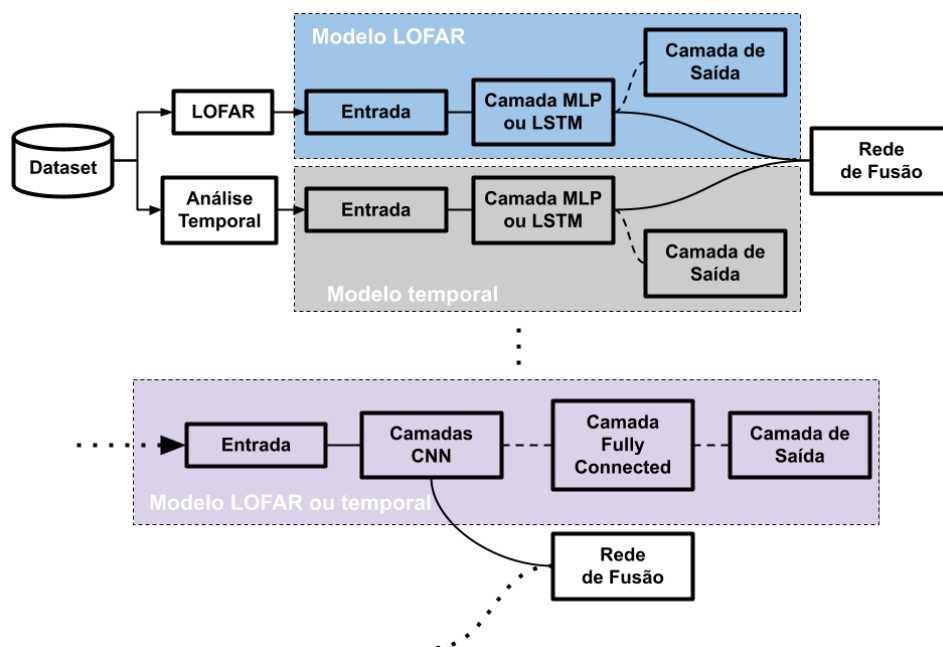


Figura 4.11: Processo de montagem do modelo proposto.

Capítulo 5

Resultados e Discussões

Neste capítulo serão descritos detalhadamente os conjuntos de dados e os processamentos abordados para a elaboração deste trabalho, bem como a avaliação dos diferentes classificadores desenvolvidos, seus resultados e discussões pertinentes.

5.1 Conjuntos de Dados Utilizados

Neste trabalho foram utilizados dois conjuntos de dados. O primeiro com gravações de sinais acústicos coletados de sonares passivos de submarinos em operação contendo 24 classes. Eles foram gravados em arquivos com extensão .wav pela MB. A resolução de amplitude é de 16 bits e a frequência de amostragem varia para cada classe (os valores variam de 22.050 até 48.000 Hz). Vale ressaltar que esses dados foram adquiridos através de um sistema de sonar passivo de submarino em um ambiente real de operação. Logo, possuem todas as influências que um sinal acústico produzido por um navio pode sofrer até chegar ao canal de áudio do OS. Essas influências podem ser: perdas na propagação do sinal no ambiente marinho, interferência de outras fontes sonoras, ruído eletrônico do próprio sistema de sonar, dentre outros fatores.

O segundo conjunto é composto por gravações de sinais acústicos coletados de um sonar passivo monitorando o tráfego marítimo numa região portuária com profundidade variando em torno de 10 metros (contendo 31 classes), denominado Vigilância e Informações Passivas em Portos (VIPP). Os sinais foram gravados em arquivos com a extensão .wav, 16 bits de resolução e frequência de amostragem de 31.250Hz. O sistema VIPP, desenvolvido pela Instituto de Pesquisas da Marinha (IPqM), consiste em um arranjo de hidrofones dispostos em uma estrutura semi elíptica, operando de modo passivo integrado ao Sistema de Detecção, Acompanhamento e Classificação de Contatos (SDAC). Além das funcionalidades descritas em seu próprio nome, o SDAC grava os sinais detectados na área de vigilância do VIPP e pode ser empregado em outros sistemas de sonar [15]. O arranjo de sensores acústicos do VIPP

estão representados na Figura 5.1. Cada sensor acústico é chamado de *stave*: um conjunto de hidrofones encapsulados (neste caso específico são três hidrofones). No caso do *stave* utilizado, a tensão do sinal de saída é a soma das tensões de cada hidrofone. A Figura 5.1 apresenta o sistema VIPP.

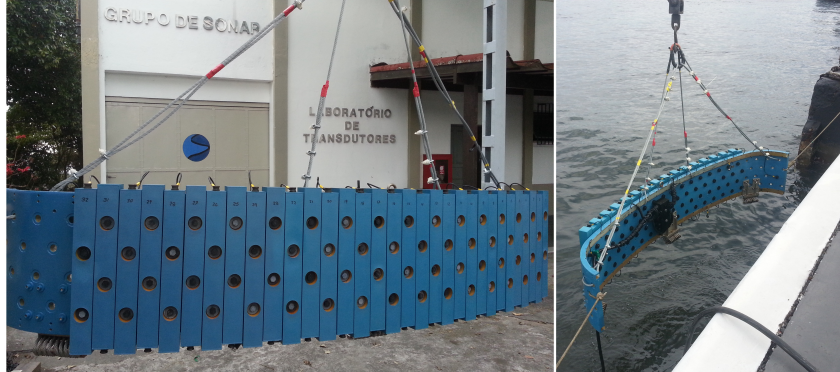


Figura 5.1: Vigilância e Informações Passivas em Portos (VIPP).

Fonte: GOLTZ [15]

5.2 Processamento de Sinais

O processamento dos sinais está de acordo com os métodos descritos na Seção 2.2.2 do Capítulo 2. A análise LOFAR foi aplicada com parâmetros que foram utilizados com sucesso em trabalhos anteriores [8, 15, 21], onde o processamento empregado no sinal é o mesmo usado por SOARES FILHO [21]. Os parâmetros são decimação igual a 3, seguido de um filtro passa-baixa digital de ordem 8, desenvolvido com frequência de corte em 2.870 HZ [21]. Após o filtro, o sinal passa por uma janela de Hanning e uma STFT [37]. Nestes dois processamentos, dois valores foram utilizados para o tamanho da janela de processamento, 1024 e 2048. O melhor valor para esse parâmetro será descrito mais a frente neste capítulo, pois sua escolha depende dos resultados do índice SP (descrito na Seção 4.4.2) do conjunto de validação para cada modelo treinado. Como o sinal foi filtrado, alguns *bins* de frequência não contém informação útil e, portanto são descartados nesta fase. Deste modo, 400 *bins* são aproveitados para a janela de 1024 e 800 *bins* para a janela de 2048 pontos [35]. No processamento da Análise Temporal também foram utilizados os mesmos parâmetros do bloco de decimação e de filtragem da análise LOFAR. Após a filtragem, os dados foram normalizados com média zero e variância unitária. Os valores utilizados na janela de processamento são 1024 e 2048. As Tabelas 5.1 e 5.2 apresentam as quantidades de amostras do conjunto de dados após o processamento das análises (LOFAR e Temporal) respectivamente para os dados oriundos do *dataset* de águas profundas e rasas. O número de ponto das janelas apresentadas nas Tabelas (5.1 e

5.2) é 1024. Sendo assim, para 2048 pontos os valores de cada classe é reduzido pela metade. Por fim, a largura da fenda central e a largura da fenda lateral do filtro TPSW são 5 e 17 *bins* [74].

Tabela 5.1: Quantidade de janelas para cada classe de navios do conjunto de dados proveniente do submarino

Classe	Nº Janelas	Classe	Nº Janelas	Classe	Nº Janelas
1	490	9	3750	17	1270
2	460	10	5420	1	850
3	490	11	1070	19	1290
4	490	12	530	20	710
5	1700	13	2640	21	1240
6	520	14	1290	22	3800
7	2270	15	1290	23	10770
8	1370	16	4500	24	3190

Tabela 5.2: Quantidade de janelas para cada classe de navios do conjunto de dados proveniente do VIPP

Classe	Nº Janelas	Classe	Nº Janelas	Classe	Nº Janelas
1	2050	12	1050	23	2220
2	600	13	1370	24	2950
3	3870	14	710	25	1750
4	1320	15	1170	26	1300
5	820	16	410	27	1430
6	1760	17	320	28	2720
7	1020	18	1350	29	1920
8	630	19	2490	30	1580
9	1200	20	1410	31	1840
10	890	21	1570	–	–
11	3740	22	600	–	–

5.3 Definição da Arquitetura e Treinamento dos Modelos

Após o processamento dos sinais, os dados foram reorganizados de forma hierárquica de acordo com o conhecimento especialista [15], conforme descrito na Seção 4.1. Os níveis representados para o conjunto de águas profundas são “Super”, A, B, C, D, AA, AB, AC, DA e DB. No conjunto de dados proveniente do VIPP, os níveis são “Super”, A, B e B1 a B8. Este trabalho abordará apenas o nível “Super” de cada conjunto e a Tabela 5.3 apresenta a quantidade de amostra em cada nível.

Mesmo após a organização hierárquica, ambos os conjuntos apresentam um grande desbalanceamento entre as classes. No *dataset* de águas profundas (24 classes) as classes A e C são respectivamente as classes com o maior e menor número de amostras. No de águas rasas (31 classes) a Classe B é a com o maior número de amostras. Esse fator pode levar os classificadores a apresentarem dependências para as classes com maior quantidade de janelas, fator prejudicial ao treinamento. Como

mencionado na Seção 4.4, para contornar esse problema empregou-se no treinamento das redes neurais um fator de ponderação no gradiente. Os pesos para cada classe podem ser calculados utilizando sua frequência, dessa forma as classes minoritárias receberam mais pesos durante o treinamento.

Tabela 5.3: Quantidade de amostras por classe da saída do elemento “Super”

Dataset	Níveis	Classes	Amostras	Total de Amostras
24 Classes	A	9, 10, 13, 14, 16, 23, 1, 2, 22, 21	33.120	50.160
	B	4, 6, 8, 12, 17, 19	5.470	
	C	11, 24	4.260	
	D	5, 7, 15, 3, 18, 20	7.310	
31 Classes	A	1, 2, 3, 4	7.840	48.060
	B	5, 6, 7, \dots , 29, 30, 31	40.220	

Como mencionado na Seção 4.4 alguns parâmetros foram escolhidos para o treinamento. O otimizador é o Adam com a taxa de aprendizado $\eta = .001$ e as taxas de decaimento são $\beta_1 = .9$ e $\beta_2 = .99$. O critério de parada prematura são 30 épocas de treinamento e o conjunto de dados é subdividido em 10 *folds*, sendo que um *fold* foi reservado para o conjunto de teste e dos 9 *folds* restante, 8 foram destinados para o treinamento e um para validação. Vale ressaltar que o conjunto de validação dentro da mesma iteração não faz parte do conjunto de treinamento e o conjunto de teste variará em cada iteração (conforme o critério de estabelecido na Seção 4.4.1). Esses parâmetros são replicados para todas as arquiteturas. Os nós de entrada da arquitetura MLP possuem 400 e 800 atributos para as janelas de 1024 e 2048 pontos da análise LOFAR respectivamente. Na análise Temporal os atributos são 1024 e 2048.

5.3.1 Águas Profundas (24 Classes)

Resultados para a Rede Neural MLP

A Figura 5.2 apresenta a curva do valor médio, com margem de erro, do índice SP aplicado sobre o conjunto de validação pela variação de neurônios da camada oculta da rede MLP. Analisando o comportamento das curvas, o índice SP tem uma ligeira melhora com o aumento do número de neurônios.

A Tabela 5.4 apresenta o valor médio e o erro do índice SP variando o número de neurônios. É possível observar que o modelo LOFAR obteve resultados melhores que o temporal. O uso da janela de 2048 pontos apresenta um valor médio ligeiramente melhor que os valores obtidos com a janela de 1024 pontos, mas considerando a variância estatística os valores se sobrepõem. Esta análise pode ser replicada para o temporal, sendo que neste modelo a janela com 1024 pontos obteve os melhores valores.

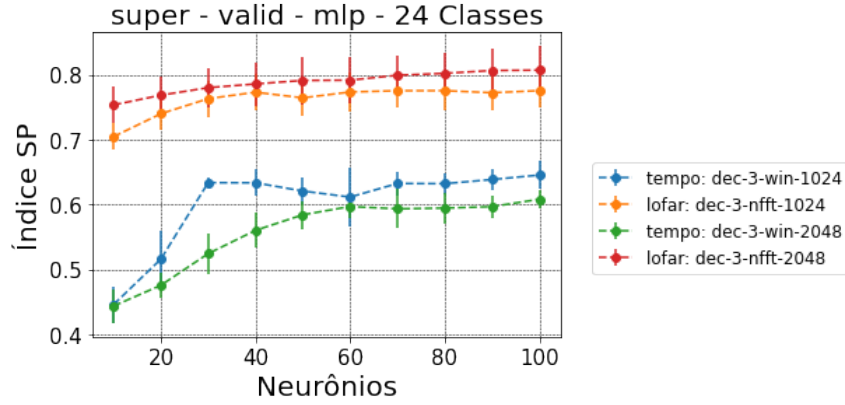


Figura 5.2: Gráfico do índice SP pela variação do número de neurônios (águas profundas).

Tabela 5.4: Resultados experimentais do índice SP para o modelo MLP do conjunto de validação (águas profundas)

Neurônios	Índice SP (%)			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
10	70,44 ± 2,05	75,33 ± 2,82	44,56 ± 2,85	44,33 ± 2,56
20	73,98 ± 2,44	76,84 ± 2,91	51,56 ± 4,33	47,53 ± 1,92
30	76,30 ± 2,77	77,99 ± 3,00	63,35 ± 0,79	52,41 ± 3,19
40	77,29 ± 2,86	78,57 ± 3,32	63,33 ± 2,04	56,00 ± 2,71
50	76,42 ± 2,68	79,11 ± 3,68	62,06 ± 2,04	58,39 ± 2,17
60	77,33 ± 2,99	79,15 ± 3,48	61,13 ± 4,52	59,67 ± 1,84
70	77,53 ± 2,52	79,90 ± 3,11	63,22 ± 1,88	59,31 ± 3,05
80	77,53 ± 2,94	80,20 ± 3,09	63,20 ± 1,64	59,43 ± 2,31
90	77,21 ± 2,75	80,64 ± 3,31	63,83 ± 1,71	59,65 ± 1,77
100	77,54 ± 2,51	80,70 ± 3,69	64,52 ± 2,21	60,79 ± 1,42

Apenas os valores do índice SP não são suficientes para definir o número de neurônios da camada oculta da rede. Outros fatores como, o número de parâmetros, a distribuição dos resultados da camada de saída e o comportamento das curvas de treinamento (apresentadas com detalhes no Apêndice C.3) são importantes para essa tomada de decisão. O número total de amostras do *dataset* para as janelas de 1024 pontos é 50.160. Como o conjunto de dados é subdividido em 10 *folds*, em cada iteração da validação cruzada 80% dos dados são para treinamento, 10% para validação e 10% para teste. Portanto, 40.128 amostras são destinadas ao treinamento. A classe C com 3.408 amostras para a treinamento é o classe minoritária. A classe A é a classe majoritária com 33.120 amostras para o treinamento. Com as janelas de 2048 pontos esses valores apresentados reduzem pela metade. A Tabela 5.5 apresenta a quantidade de parâmetros variando pelo número de neurônios da camada oculta da rede MLP. Modelos com muitos parâmetros possuem viés relativamente baixo, mas variância alta, já que é necessário estimar todos eles. Já modelos com poucos parâmetros possuem variância baixa, mas viés muito alto, já que são demasiado simplistas para descrever o modelo classificador dos dados [78]. Assim, com a finalidade de obter um bom poder de classificação, deve-se escolher um modelo com o número de parâmetros nem tão alto, nem tão baixo e o número de amostras de treinamento é um dos balizadores para a escolha dos modelos.

Tabela 5.5: Quantidade de parâmetros para cada modelo MLP (águas profundas)

Neurônios	Parâmetros			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
10	4.274	8.274	10.514	20.754
20	8.524	16.524	21.004	41.484
30	12.774	24.774	31.494	62.214
40	17.024	33.024	41.984	82.944
50	21.274	41.274	52.474	103.674
60	25.524	49.524	62.964	124.404
70	29.774	57.774	73.454	145.134
80	34.024	66.024	83.944	165.864
90	38.274	74.274	94.434	186.594
100	42.524	82.524	104.924	207.324

Os gráficos de violino¹ ajudam a avaliar a diferença estatística dos resultados, dentre os modelos que obtiveram resultados semelhantes. Na Figura 5.3 são representados os gráficos de violino do índice SP dos dados do conjunto de validação pela variação de neurônios da camada oculta do modelo. No modelo LOFAR com 1.024 amostras, o índice SP não mostra evolução de desempenho para valores acima de 40 neurônios. Por outro lado, para os modelos com amostras de 2.048, apesar de um valor médio crescente com a complexidade, a barra de erro também mostra que não

¹O gráfico de violino é a combinação do *boxplot* e a linha de densidade no mesmo diagrama, permitindo visualizar detalhes que as duas técnicas sozinhas não seriam capazes de detectar [79].

há um ganho estatístico. Esse resultado somado ao fato que 2.048 torna o modelo mais complexo e o processamento mais lento, critério relevante para a um cenário de guerra, consideraremos apenas o número de pontos das janelas de 1024. Os modelos LOFAR com 30 e 40 neurônios na camada oculta serão analisados e seus respectivos índice SP são: $(76,30 \pm 2,77)\%$ e $(77,29 \pm 2,86)\%$.

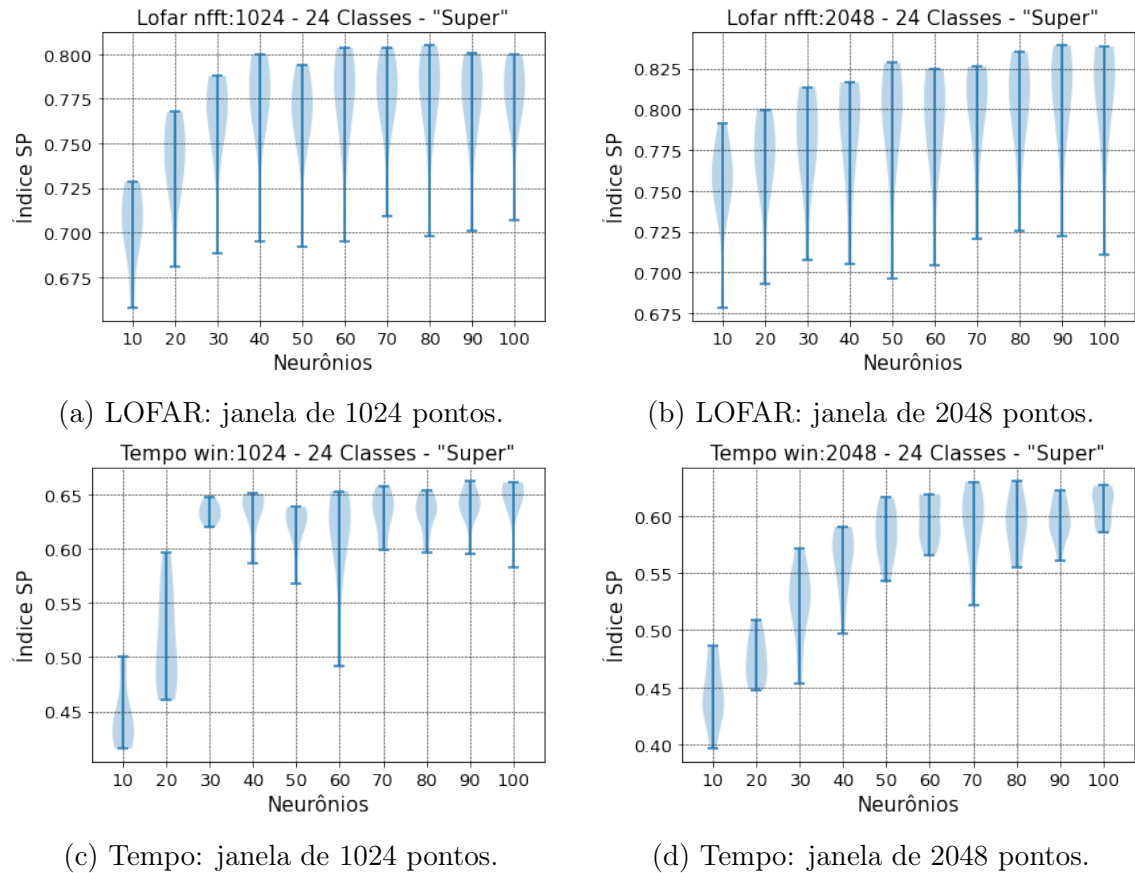


Figura 5.3: Gráfico de Violino do desempenho dos modelos MLP no conjunto de validação (águas profundas).

Outra maneira de comparar os modelos é analisando a distribuição de probabilidade de saída. A Figura 5.21 mostra uma matriz de histogramas da saída dos modelos MLP com 30 e 40 neurônios na camada oculta. O processo de geração ocorreu da seguinte maneira: os modelos treinados foram alimentados por amostras do conjunto de validação. As linhas “neurônios” representam cada neurônio de saída da rede. As colunas com indicação das classes significam que todos os histogramas daquela coluna são resultados dos modelos alimentados por dados daquela classe específica. Então é esperado que, na coluna “Classe A”, a distribuição da linha “Neurônio 01” esteja mais próxima do valor 1 e nas outras linhas mais próxima de 0. As duas distribuições estão bem parecidas e próximas dos extremos na maioria dos casos. Entretanto, os eventos dos histogramas das classes A e B dos neurônios 01 e 02 estão mais distribuídos com uma grande quantidade de eventos no centro. Con-

tudo, a MLP com 40 neurônios apresentou um pouco mais de eventos concentrados nos extremos.

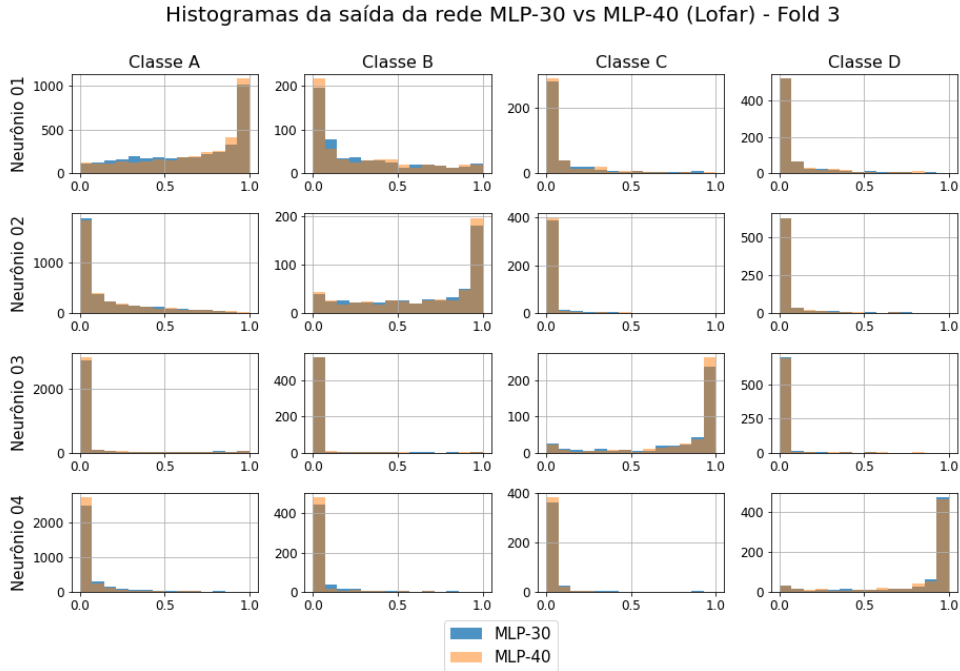


Figura 5.4: Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40, *fold* 3 (águas profundas).

Para uma melhor compreensão do processo de treinamento, a Figura 5.5 mostra a matriz de confusão dos conjuntos de treinamento e validação dos modelos LOFAR com 30 e 40 neurônios. Em ambos os casos a rede não conseguiu diferenciar bem as classes A e B. Esse problema está associado ao alto número de Falsos Negativos da classe B. Este problema também ocorreu entre as classes A e D, porém, a interferência foi um pouco mais suave. Em ambos os modelos, a maior acurácia durante o treinamento foi na classe C com $(99,35 \pm 0,33)\%$ e $(99,41 \pm 0,46)\%$ para 30 e 40 neurônios na camada oculta respectivamente. Entretanto, durante a validação, a classe D obteve um desempenho semelhante ao da classe C com as acurácias de $(81,55 \pm 3,67)\%$ e $(82,79 \pm 2,25)\%$. A menor acurácia é na classe B com resultados em torno de 70% na validação. As curvas de treinamento dos modelos encontram-se no Apêndice C.1.1.

Após analisar o treinamento para os modelos com 30 e 40 neurônios na camada oculta, não foi constatado uma melhora significativa que justifique o aumento de neurônios para 40. Isso também pode ser observado na Tabela 5.6, que mostra o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos. Sendo assim, o modelo LOFAR com 30 neurônios será utilizado como entrada da rede de fusão.

Observando o modelo temporal na Tabela 5.5, nota-se um aumento do valor do índice SP com o modelo com 30 neurônios, entretanto, os modelos mais complexos

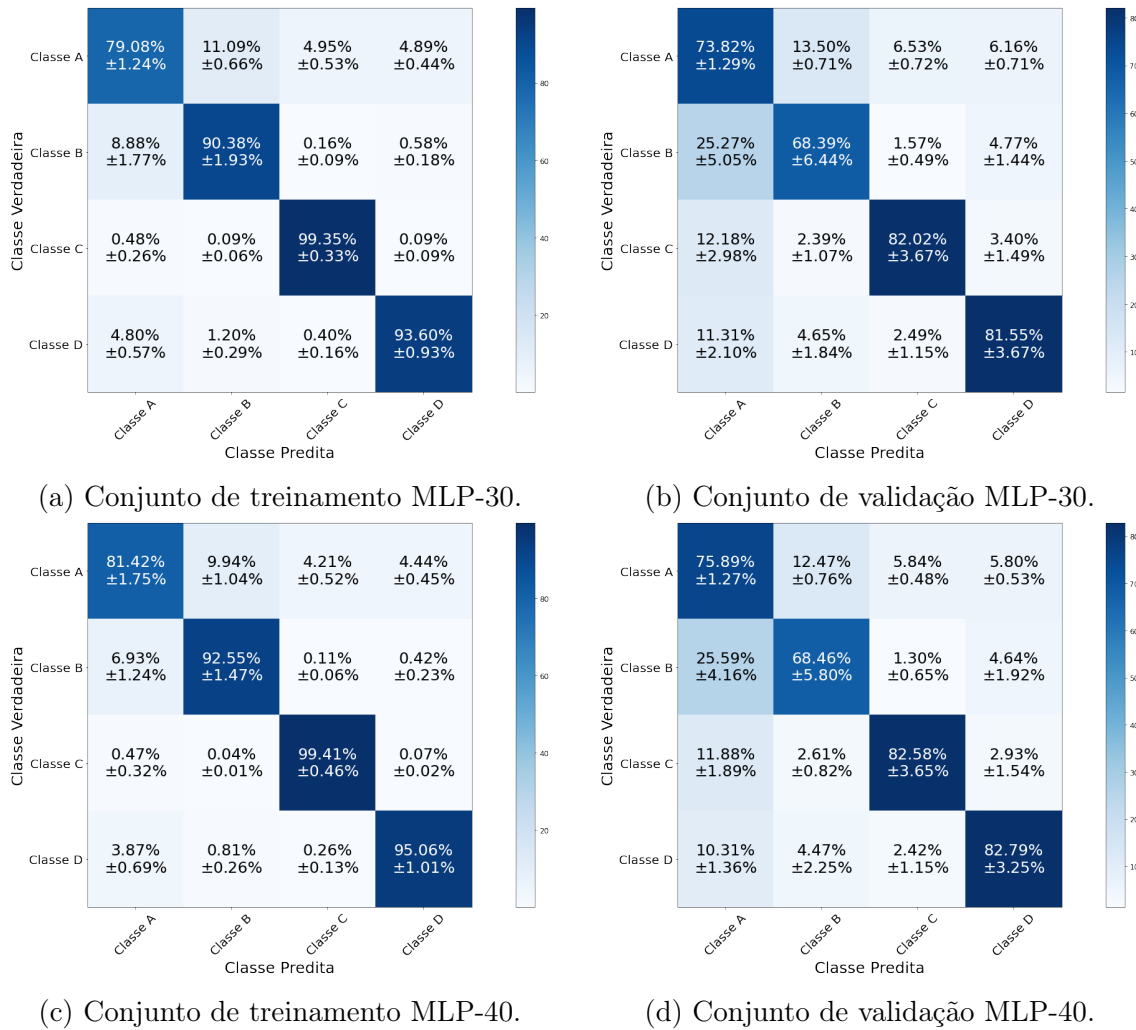


Figura 5.5: Matriz de Confusão dos modelos LOFAR MLP com 30 e 40 neurônios (águas profundas).

Tabela 5.6: Resultados experimentais da análise LOFAR (1024) dos modelos MLP para o conjunto de validação (águas profundas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
30	76,30 ± 2,77	75,02 ± 1,29	65,30 ± 1,71	76,44 ± 2,71
40	77,29 ± 2,86	76,63 ± 1,63	66,96 ± 2,16	77,43 ± 2,81

não melhoram significativamente o valor médio e ocorre um aumento da incerteza. O SP do modelo com 30 neurônios é $(63,35 \pm 0,79)\%$. O gráfico de violino da Figura 5.3c mostra que os resultados dos *folds* estão mais concentrados na parte superior violino. A Figura 5.6 mostra a matriz de histogramas da saída do modelo temporal MLP com 30 neurônios na camada oculta. As distribuições estão mais dispersas, o que era de se esperar, tendo em vista que os resultados do SP estão em torno de 60%. Observando as colunas A e B para os neurônios 01 e 02, é possível notar uma dificuldade do modelo em distinguir as classes A e B.

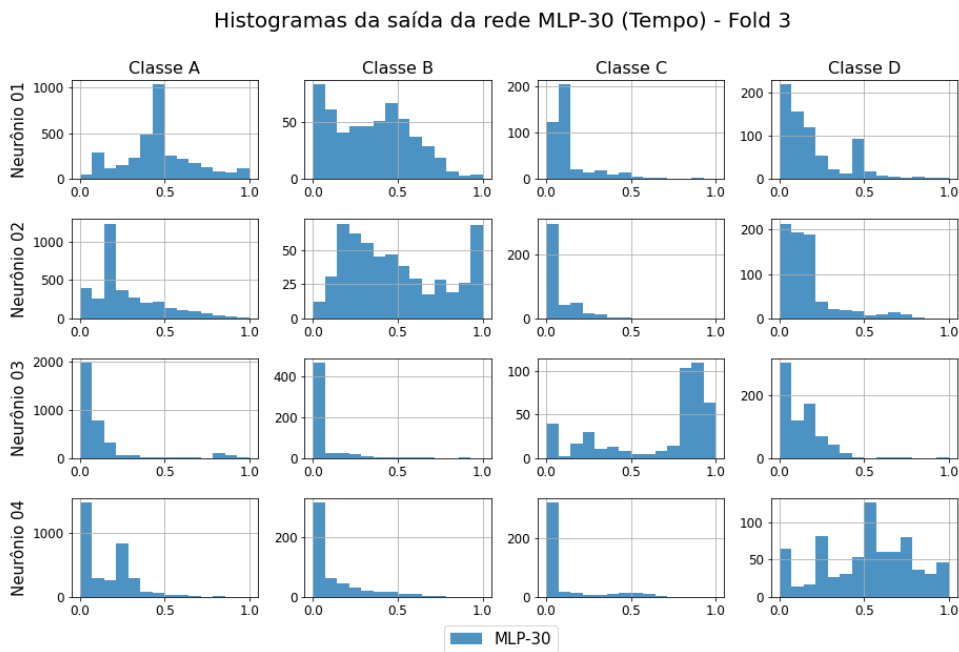
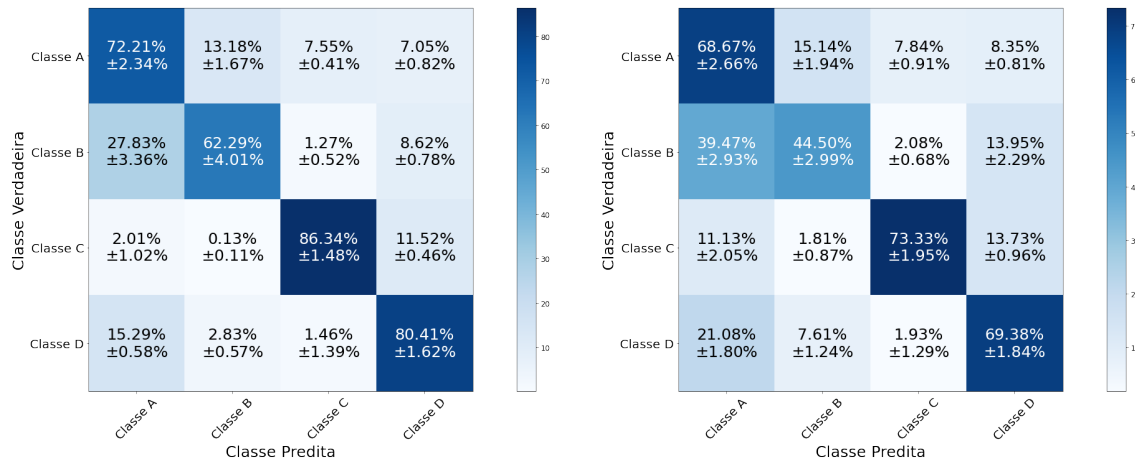


Figura 5.6: Histogramas da saída do modelo temporal MLP-30, *fold* 3 (águas profundas).

A Figura 5.7 mostra a Matriz de confusão dos conjuntos de treinamento e validação do modelo temporal com 30 neurônios. A classe B continua tendo o pior resultado dentre as classes com a acurácia de $(44,50 \pm 2,99)\%$ no conjunto de validação. De fato, a dificuldade do modelo em discernir as classes A e B chama a atenção. A classe C foi a que teve o melhor índice de acerto com $(73,33 \pm 1,95)\%$ no conjunto de validação. As curvas de treinamento do modelo encontram-se no Apêndice C.1.1.

Em complemento à análise do modelo temporal com 30 neurônios, os dados de classificação são: acurácia de $(66.58 \pm 1.77)\%$, precisão de $(55.2 \pm 1.35)\%$ e *recall* de $(63.97 \pm 0.75)\%$.



(a) Conjunto de treinamento MLP-30.

(b) Conjunto de validação MLP-30.

Figura 5.7: Matriz de Confusão do modelo temporal MLP com 30 neurônios (águas profundas).

Resultados para a Rede Neural CNN

Como mencionado na Seção 4.3, a estrutura da rede CNN possui diversos parâmetros a serem observados, tais como: filtros na camada convolucional, quantidade de camadas, *kernel* de convolução e neurônios na camada densa. A Figura 5.8 apresenta a curva do valor médio, com margem de erro do índice SP aplicado sobre o conjunto de validação para cada modelo convolucional implementado com base na variação dos parâmetros citados. Leia-se, a nomenclatura “cnn_16_16_mlp_10_kernel_3”, como duas camadas convolucionais com 16 filtros cada, uma camada *fully connected* com 10 neurônios e um *kernel* igual à 3. Essa leitura é similar para os outros modelos. Assim como ocorreu na rede MLP, o modelo LOFAR obteve resultados melhores que o temporal. Em ambos os domínios, a presença de duas camadas convolucionais para extração de características melhora a classificação.

Para uma análise mais detalhada, é preciso observar numericamente os resultados do índice SP. Devido à quantidade de modelos, algumas tabelas de resultados serão apresentadas de forma resumida e suas versões completas estão no Apêndice ???. A Tabela 5.7 (versão completa na Tabela C.1) mostra o valor médio, com o erro associado do índice SP aplicado sobre o conjunto de validação de alguns modelos gerados a partir da variação dos parâmetros citados anteriormente. Na coluna “Modelos” as informações numéricas (16,8,10,3) são respectivamente, o número de filtros na primeira camada convolucional, número de filtros na segunda camada convolucional, número de neurônios na camada *Fully Connected* e o número do *kernel*. Os resultados do índice SP melhoram com o aumento da quantidade de filtros das camadas CNN e do número de neurônios na camada MLP.

Alguns modelos gerados pela busca paramétrica não podem ser utilizados por

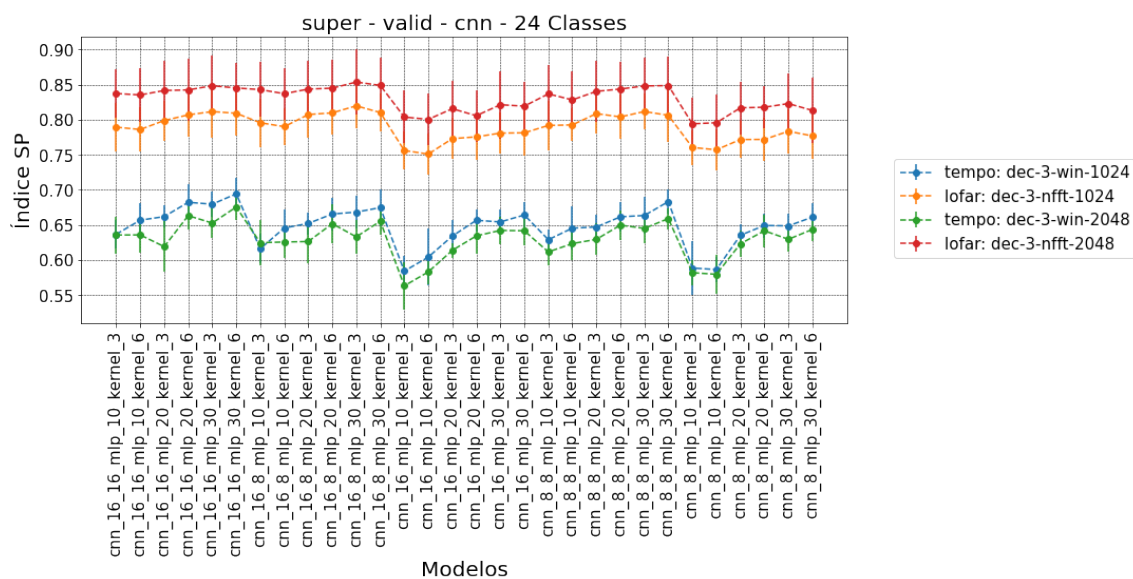


Figura 5.8: Gráfico do índice SP pela variação do número de modelos CNN (águas profundas).

Tabela 5.7: Resultados experimentais do índice SP para o modelo CNN do conjunto de validação (águas profundas).

Modelos	Índice SP (%)			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
(16,16,10,3)	78,93 ± 3,47	83,73 ± 3,49	63,65 ± 2,05	63,55 ± 2,60
(16,16,10,6)	78,60 ± 3,18	83,52 ± 3,83	65,67 ± 2,48	63,57 ± 2,58
(16,16,30,3)	81,15 ± 3,71	84,83 ± 4,33	67,95 ± 1,76	65,25 ± 2,18
(16,8,30,3)	81,97 ± 3,15	85,36 ± 4,67	66,82 ± 2,34	63,29 ± 2,39
(16,-,10,3)	75,60 ± 2,65	80,37 ± 3,76	58,40 ± 2,18	56,30 ± 3,42
(8,8,10,3)	79,17 ± 3,50	83,67 ± 4,11	62,89 ± 1,43	61,14 ± 1,96
(8,8,20,3)	80,86 ± 2,90	84,03 ± 4,30	64,70 ± 1,68	62,94 ± 2,25
(8,8,30,6)	80,62 ± 3,83	84,86 ± 4,11	68,32 ± 1,79	65,90 ± 1,49
(8,-,10,6)	75,71 ± 2,91	79,56 ± 3,96	58,63 ± 1,80	57,95 ± 2,83

apresentarem uma quantidade de parâmetros muito acima da quantidade de amostras do *dataset*. Para reduzir a quantidade de parâmetros dos modelos e maximizar a extração de características, após as camadas convolucionais foi adicionada uma camada *MaxPolling* (descrita com mais detalhes no Apêndice B.1.2). No modelo LOFAR, a camada *MaxPolling* foi adicionada após a primeira camada convolucional. No modelo temporal, devido à dimensão de entrada da rede ser maior, a camada foi adicionada após todas as camadas convolucionais. A Tabela 5.8 (versão completa na Tabela C.2) mostra o número de parâmetros dos modelos criados com a busca paramétrica. Mesmo com a estratégia da camada *MaxPolling*, a grande maioria dos modelos com janelas de 2048 pontos ainda apresentam uma quantidade de parâmetros bem acima do número de amostras do *dataset*. Assim como no modelo MLP, a escolha dos modelos de ser realizada com parcimônia em relação a quantidade de parâmetros. Sendo assim, os modelos com janelas de 1024 pontos são mais apropriados, pois permitem uma maior gama de escolhas.

Tabela 5.8: Quantidade de parâmetros para cada modelo CNN (águas profundas)

Modelos	Parâmetros			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
(16,16,30,6)	97.818	194.818	124.698	247.578
(16,8,30,3)	48.610	96.610	62.050	123.490
(16,8,30,6)	49.042	98.042	62.482	123.922
(16,-,10,3)	32.118	64.118	82.038	163.958
(16,-,30,6)	96.266	192.266	246.026	491.786
(8,8,10,3)	16.286	32.286	20.766	41.246
(8,8,20,3)	32.336	64.336	41.296	82.256
(8,8,30,3)	48.386	96.386	61.826	123.266
(8,8,30,6)	48.602	96.602	62.042	123.482
(8,-,30,3)	48.186	96.186	123.066	245.946
(8,-,30,6)	48.210	96.210	123.090	245.970

Os gráficos de violino das Figuras 5.9 e 5.12 apresentam os índices SP dos dados do conjunto de validação para cada modelo proposto com a variação de parâmetros da rede convolucional. Os valores são das janelas de 1024 pontos dos modelos temporal e LOFAR. No modelo LOFAR, observa-se um aumento do valor médio com o uso de duas camadas convolucionais e uma diminuição da barra de erro para os modelos menos complexos (menor quantidade de parâmetros). Desta forma, é interessante analisar o modelo com duas camadas convolucionais de oito filtros cada, uma camada *fully conneted* com 10 neurônios e *Kernel* 3 (8,8,10,3). O valor do índice SP deste modelo é $(79, 17 \pm 3, 50)\%$. Entretanto, as curvas de treinamento dos *folds* 3 e 4 apresentadas na Figura C.3 mostram que o modelo com a mesma estrutura convolucional e com 20 neurônios na camada *fully connected* pode ser capaz de entender melhor as características extraídas das camadas convolucionais. Portanto, o modelo com 20 neurônios na camada *fully connected* também será analisado. O

índice SP do modelo (8,8,20,3) é $(80,86 \pm 2,90)\%$.

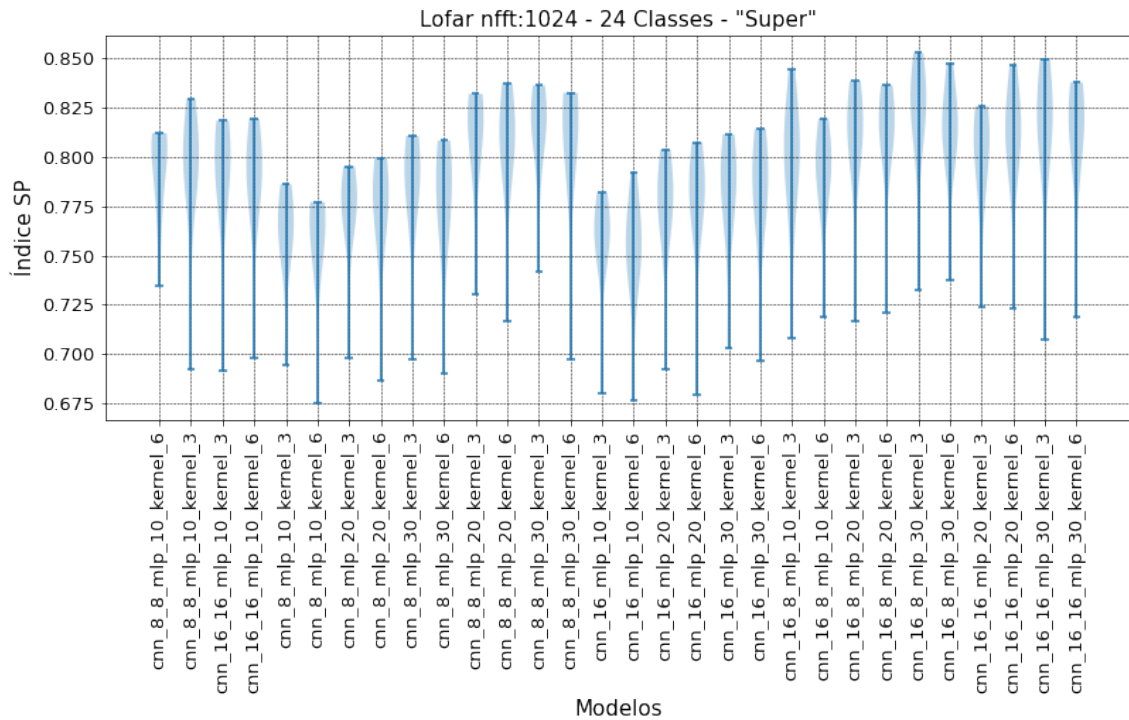


Figura 5.9: Gráfico de Violino do desempenho dos modelos LOFAR CNN no conjunto de validação (águas profundas).

A Figura 5.10 mostra a matriz de histogramas da saída dos modelos CNN citados acima com 10 e 20 neurônios na camada *fully connected*. As distribuições estão bem concentradas nos extremos. A primeira e segunda linha das colunas “Classe A” e “Classe B” mostram alguns valores na parte central dos histogramas, isso reflete uma dificuldade de distinção entre as classes A e B. O modelo (8,8,20,3) apresenta uma maior quantidade de eventos localizada nos extremos em relação ao modelo (8,8,10,3).

A Figura 5.11 apresenta a matriz de confusão dos conjuntos de treinamento e validação dos modelos LOFAR convolucionais com 10 e 20 neurônios da camada *fully connected*. A matriz de confusão reforça a observação das curvas de treinamento. Se olharmos os conjuntos de treinamento dos dois modelos, o modelo com 20 neurônios teve um desempenho melhor no valor médio e na margem de erro também. Isso reflete em uma ligeira melhora no conjunto de validação. Assim como no modelo MLP, o CNN também apresentou uma dificuldade de distinguir as classes A e B. A classe B teve o menor desempenho na classificação com uma acurácia de $(70,55 \pm 7,12)\%$ e $(69,47 \pm 6,97)\%$ para os modelos de 10 e 20 neurônios respectivamente. As curvas de treinamento dos modelos encontram-se no Apêndice C.1.2.

A Tabela 5.9 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com 10 e 20 neurônios na camada *fully connected*. Nota-se uma melhora

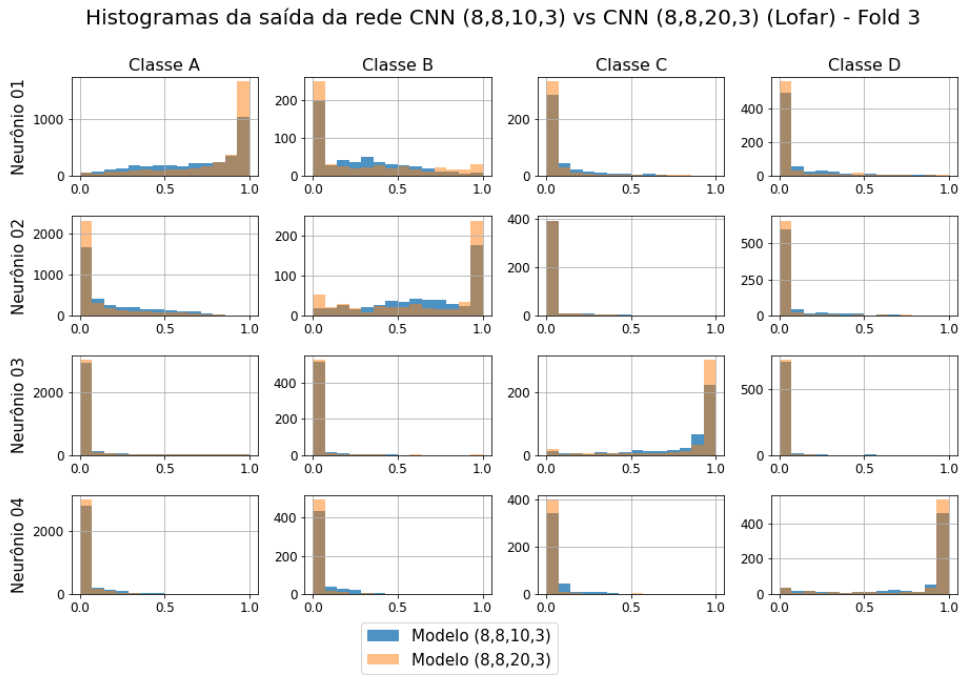


Figura 5.10: Histogramas da saída dos modelos LOFAR CNN (8,8,10,3) e (8,8,20,3), *fold* 3 (águas profundas).

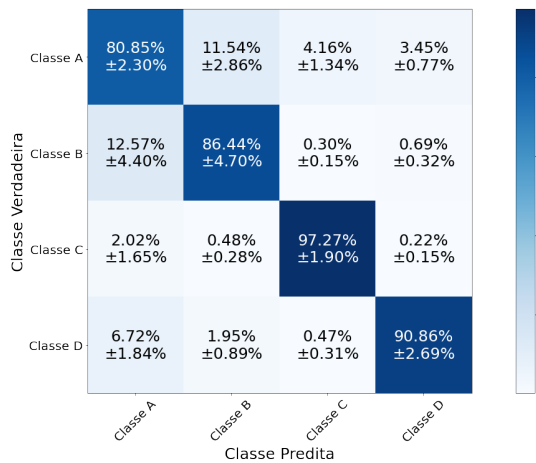
nos resultados de acurácia e precisão, além de uma diminuição do erro do índice SP.

Tabela 5.9: Resultados experimentais da análise LOFAR (1024) dos modelos CNN para o conjunto de validação (águas profundas)

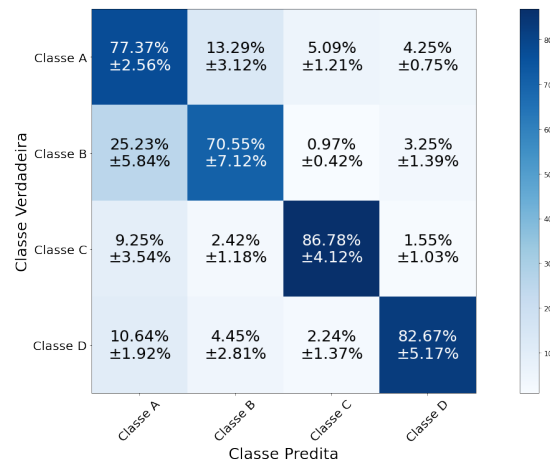
Modelos	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
(8, 8, 10, 3)	79,17 ± 3,50	78,17 ± 2,07	69,77 ± 2,47	79,34 ± 3,40
(8, 8, 20, 3)	80,86 ± 2,90	81,63 ± 2,07	73,27 ± 2,87	81,06 ± 2,76

Os modelos temporais apresentam resultados do índice SP semelhantes em torno de 65%, com exceção de alguns modelos. Sendo assim, a quantidade de parâmetros se torna um fator muito importante na escolha do modelo. Modelos muito complexos não melhoram a margem de erro do índice SP e a melhora no valor médio é quase imperceptível (vide Tabela 5.8). Conseqüentemente, assim como ocorreu no LOFAR, o modelo temporal analisado será com duas camadas convolucionais de 8 filtros, uma camada *fully connected* de 10 neurônios e um *kernel* 3. Os resultados serão comparados com o modelo com a mesma estrutura convolucional e com 20 neurônios na camada *fully connected*, pois esse modelo apresentou uma melhora no valor médio. O índice SP dos modelos com 10 e 20 neurônios são (62,89 ± 1,43)% e (64,70 ± 1,68)% respectivamente.

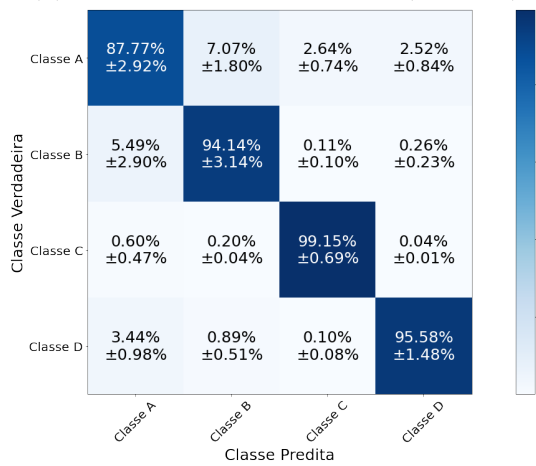
A Figura 5.13 mostra a matriz de histogramas da saída dos modelos temporais CNN citados acima, com 10 e 20 neurônios na camada *fully connected*. As distribuições estão mais dispersas em comparação ao modelo LOFAR. A primeira e segunda



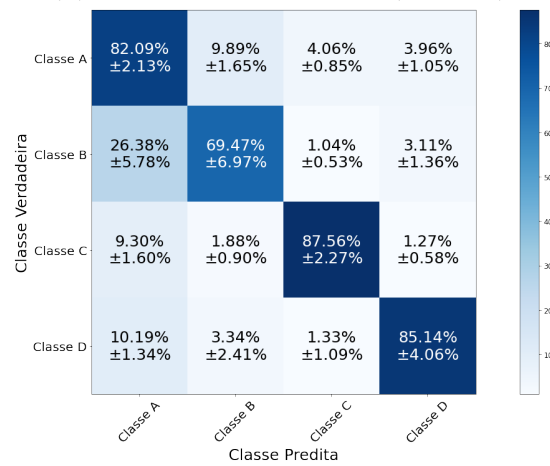
(a) Conjunto de treinamento (8,8,10,3).



(b) Conjunto de validação (8,8,10,3).



(c) Conjunto de treinamento (8,8,20,3).



(d) Conjunto de validação (8,8,20,3).

Figura 5.11: Matriz de Confusão dos modelos LOFAR CNN com 10 e 20 neurônios na camada *fully connected* (águas profundas).

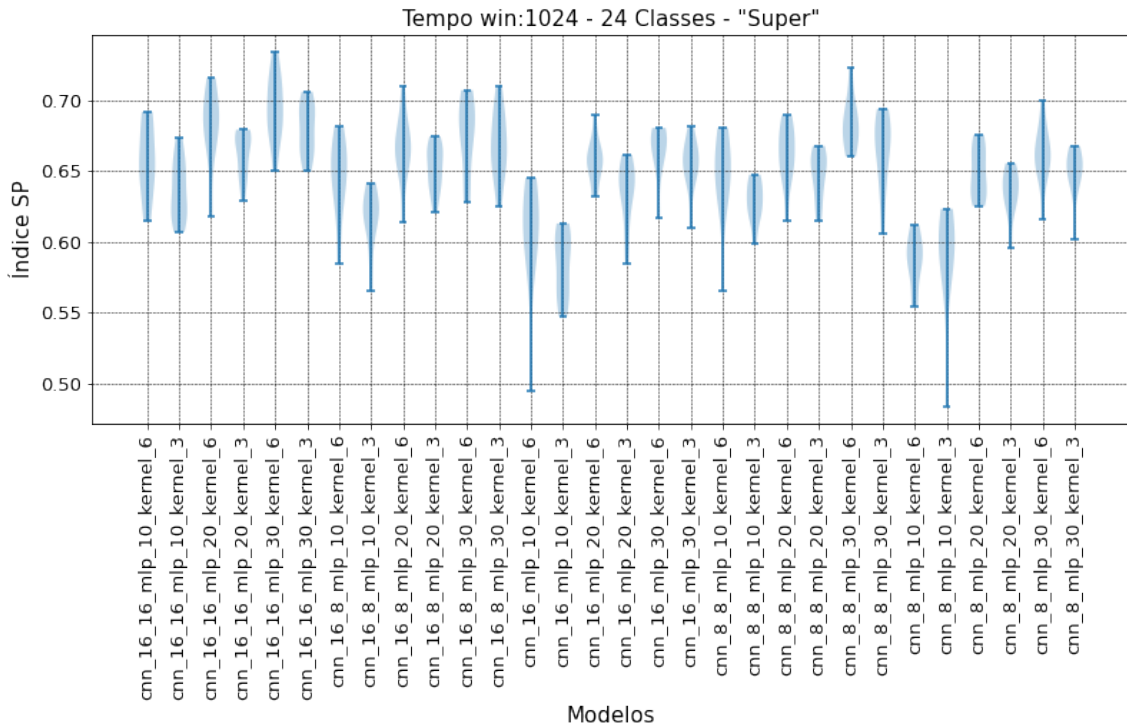


Figura 5.12: Gráfico de Violino do desempenho dos modelos temporais CNN no conjunto de validação (águas profundas).

linhas das colunas “Classe A” e “Classe B” mostram que o modelo teve dificuldades em diferenciar as classes A e B. Os valores do modelo com 20 neurônios apresentam resultados melhores do que o modelo com 10 neurônios.

A Figura 5.14 apresenta a matriz de confusão dos conjuntos de treinamento e validação dos modelos temporais convolucionais com 10 e 20 neurônios da camada *fully connected*. A matriz de confusão reforça a observação das curvas de treinamento. Se olharmos os conjuntos de treinamento dos dois modelos, o modelo com 20 neurônios teve um desempenho melhor no valor médio e na margem de erro também. Os modelos apresentaram uma dificuldade de distinguir as classes A e B. A classe A teve o menor desempenho na classificação com uma acurácia de $(42, 20 \pm 5, 89)\%$ e $(46, 99 \pm 4, 97)\%$ para os modelos de 10 e 20 neurônios respectivamente no conjunto de validação. A classe B também teve um comportamento incomum, durante o treinamento ela teve uma acurácia de cerca de 80%, mas no conjunto de validação esse valor diminuiu bastante, ficando por volta de 65%. As curvas de treinamento dos modelos encontram-se no Apêndice C.1.2.

A Tabela 5.10 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com 10 e 20 neurônios na camada *fully connected*. Os índices apresentados melhoraram com o uso de 20 neurônios.

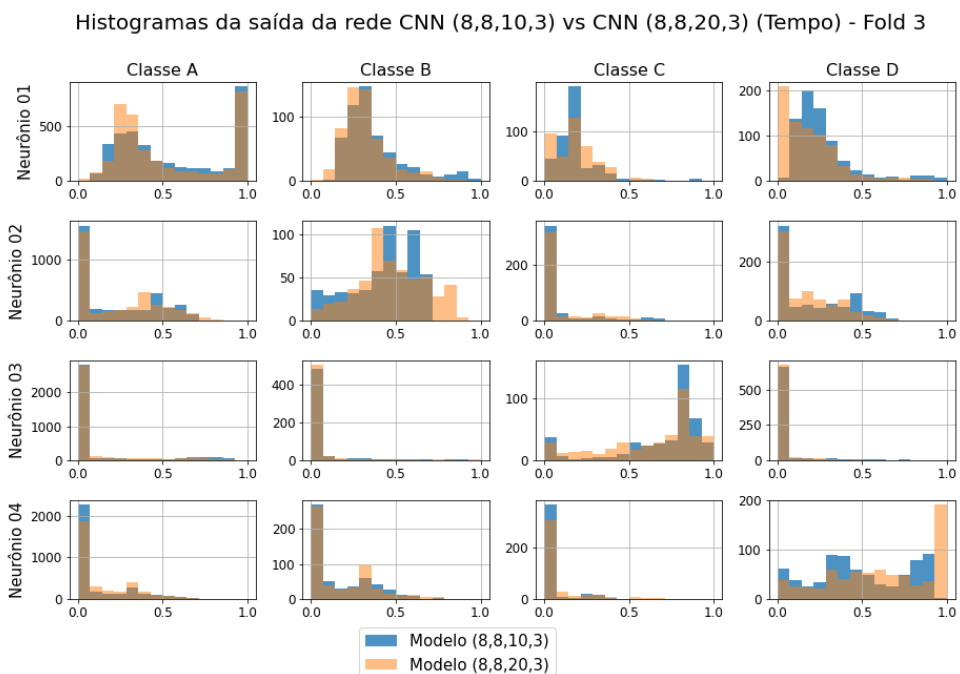
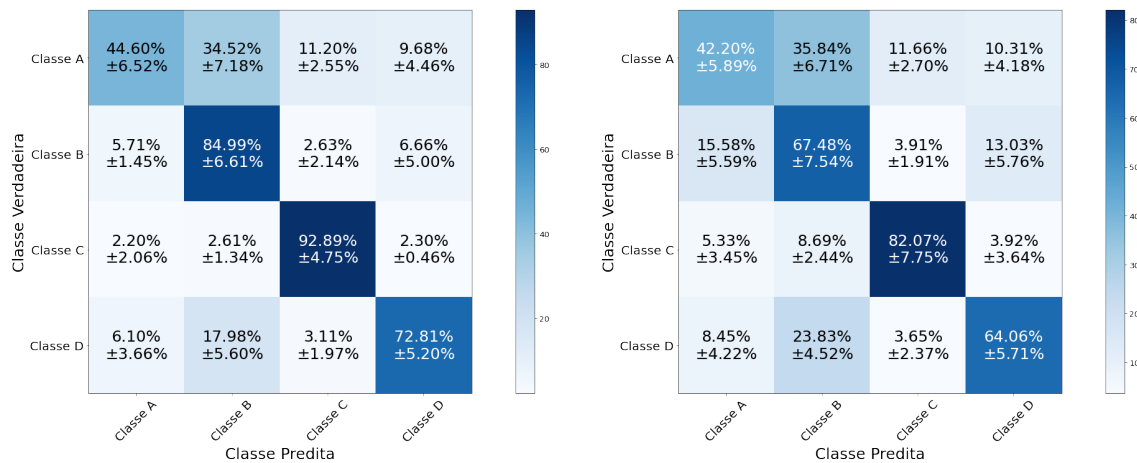


Figura 5.13: Histogramas da saída dos modelos temporais CNN (8,8,10,3) e (8,8,20,3), *fold* 3 (águas profundas).

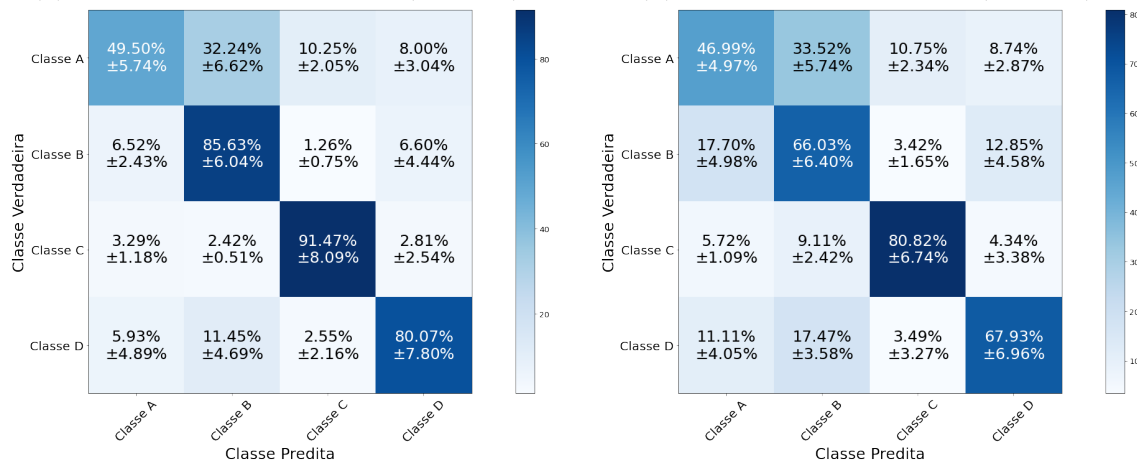
Tabela 5.10: Resultados experimentais da análise Temporal (1024) dos modelos CNN para o conjunto de validação (águas profundas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
(8, 8, 10, 3)	62,89 ± 1,43	51,30 ± 3,52	52,04 ± 2,68	63,95 ± 1,31
(8, 8, 20, 3)	64,70 ± 1,68	54,79 ± 2,65	53,45 ± 1,03	65,44 ± 1,83



(a) Conjunto de treinamento (8,8,10,3).

(b) Conjunto de validação (8,8,10,3).



(c) Conjunto de treinamento (8,8,20,3).

(d) Conjunto de validação (8,8,20,3).

Figura 5.14: Matriz de Confusão dos modelos temporais CNN com 10 e 20 neurônios na camada *fully connected* (águas profundas).

Resultados para a Rede Neural LSTM

A Figura 5.15 mostra a curva do valor médio, com margem de erro do índice SP aplicado sobre o conjunto de validação pela variação de células da camada oculta da rede LSTM. Analisando o comportamento das curvas, o índice SP melhora com o aumento do número de células. Esse comportamento ocorre nas janelas de 1024 e 2048 pontos. Como a rede LSTM é adequada para dados sequenciais (vide o Apêndice B.1.3), apenas o domínio tempo foi utilizado.

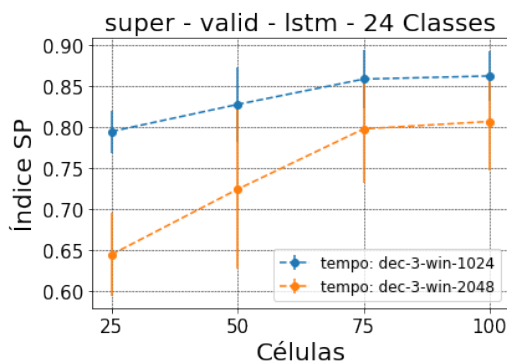


Figura 5.15: Gráfico do índice SP pela variação do número de células LSTM (águas profundas).

A Tabela 5.11 apresenta os valores médios e os erros do índice SP de acordo com o número de células. Comparando as duas janelas, 1024 e 2048 pontos, é possível observar que as janelas de 1024 pontos apresentam valores médios do índice SP melhores e uma diminuição da margem de erro.

Tabela 5.11: Resultados experimentais do índice SP para o modelo LSTM do conjunto de validação (águas profundas)

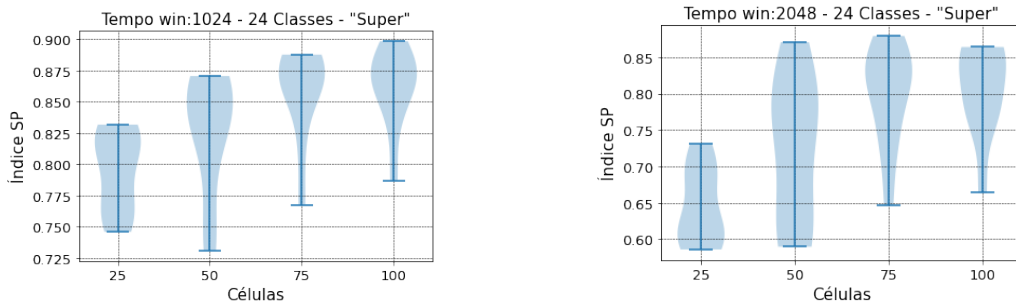
Células	Índice SP (%)	
	tempo-1024	tempo-2048
25	79,41 ± 2,59	64,41 ± 5,03
50	82,71 ± 4,59	72,37 ± 9,63
75	85,80 ± 3,51	79,73 ± 6,65
100	86,18 ± 3,04	80,64 ± 5,99

Como mencionado anteriormente nesta Seção, o número de parâmetros, o comportamento das curvas de treinamento (apresentado no Apêndice C.3), dentre outros fatores devem ser analisados na escolha do modelo. Lembrando que, o número total de amostras de treinamento do *dataset* para as janelas de 1024 pontos é de 40.128 e 20.064 para as janelas de 2048. A Tabela 5.12 mostra a variação do número de células LSTM pela quantidade de parâmetros. Nota-se que o aumento do número de parâmetros não reflete especificamente em um aumento da margem de erro.

Tabela 5.12: Quantidade de parâmetros para cada modelo LSTM

Células	Parâmetros
	tempo-1024 & 2048
25	2.804
50	10.604
75	23.404
100	41.204

A Figura 5.16 mostra os gráficos de violino do índice SP sobre os dados do conjunto de validação pela variação de células da camada oculta do modelo. Os dados das janelas de 1024 pontos tiveram valores médios do índice SP superiores aos de 2048 pontos, além da diminuição da margem de erro. Com base nesses resultados, observamos dois valores possíveis para a escolha das células LSTM. Os valores são 50 ou 75 células. Os valores do SP no conjunto de validação são respectivamente, $(82,71 \pm 4,59)\%$ e $(85,80 \pm 3,51)\%$. Acima de 75 células não há uma melhora significativa no índice SP.



(a) Tempo: janela de 1024 pontos.

(b) Tempo: janela de 2048 pontos.

Figura 5.16: Gráfico de Violino do desempenho dos modelos temporais LSTM no conjunto de validação (águas profundas).

A Figura 5.17 mostra uma matriz de histogramas da saída dos modelos LSTM com 50 e 75 células na camada oculta. As duas distribuições estão bem parecidas e seus eventos na maioria dos casos estão bem concentrados nos extremos dos histogramas. Anteriormente na rede MLP, os eventos dos histogramas das classes A e B dos neurônios 01 e 02 estavam mais distribuídos com uma grande quantidade de eventos no centro. Como na rede LSTM os eventos estão mais nos extremos, significa que a dificuldade de distinção entre as classes A e B foi reduzida. A rede com 75 células teve um desempenho levemente melhor do que o do modelo com 50 células.

A Figura 5.18 apresenta a matriz de confusão dos conjuntos de treinamento e validação dos modelos temporais com 50 e 75 células LSTM. Das técnicas apresentadas até o momento, MLP e CNN, o modelo temporal LSTM foi o que obteve os melhores resultados de classificação. O que já era esperado devido aos bons resulta-

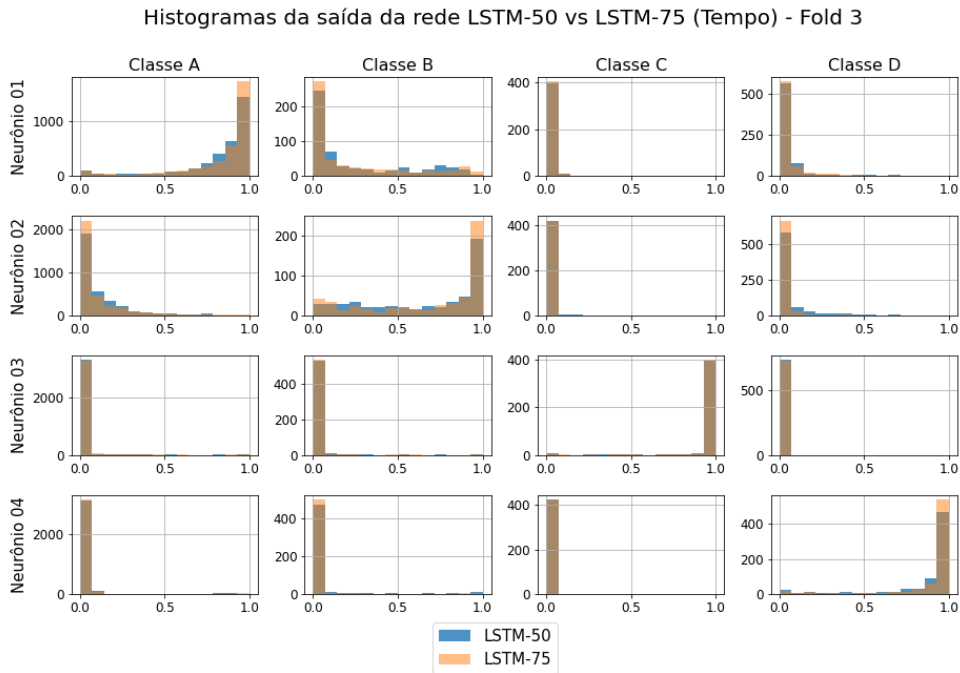


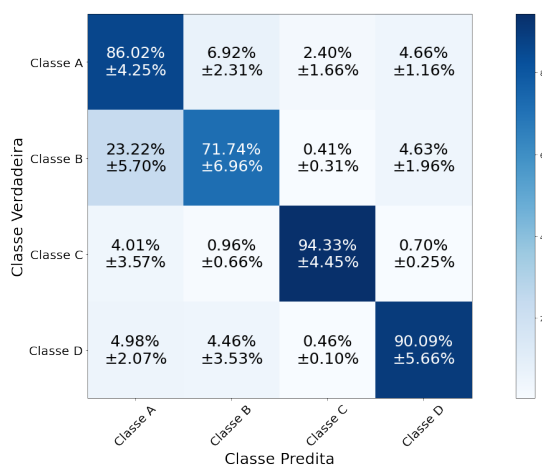
Figura 5.17: Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75, *fold* 3 (águas profundas).

dos das redes recorrentes com dados sequenciais. O problema de diferenciação entre as classes A e B permanece, mas foi bastante mitigado. A presença desse resultado nas três técnicas representa uma possível similaridade estatística nas corridas das duas classes. Houve uma redução bastante significativa na dificuldade de distinção das classes A e B, e os Falsos Positivos da classe C foram quase zerados. Com uma acurácia de $(93,47 \pm 4,21)\%$ e $(95,38 \pm 1,95)\%$ em ambos os modelos para o conjunto de validação, a classe C obteve a melhor eficiência de classificação. Superando até os valores apresentados pelo modelo LOFAR convolucional. As curvas de treinamento dos modelos encontram-se no Apêndice C.1.3.

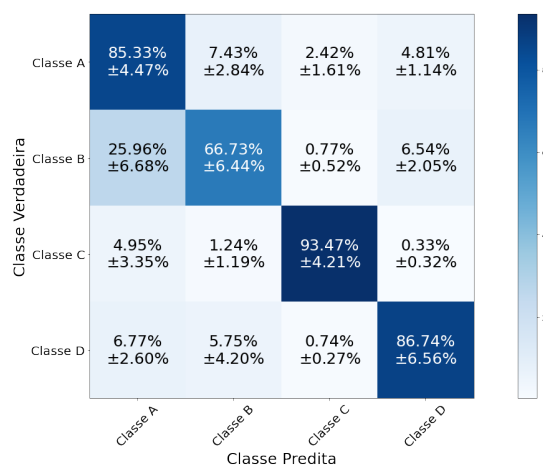
A Tabela 5.13 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com 50 e 75 células. O modelo com 75 células apresentou melhores valores médios e diminuição do erro. Somado ao fato de ter apenas 23.404 parâmetros, ele pode ser aproveitado para compor o modelo de fusão proposto neste trabalho.

Tabela 5.13: Resultados experimentais da análise Temporal (1024) dos modelos LSTM para o conjunto de validação (águas profundas)

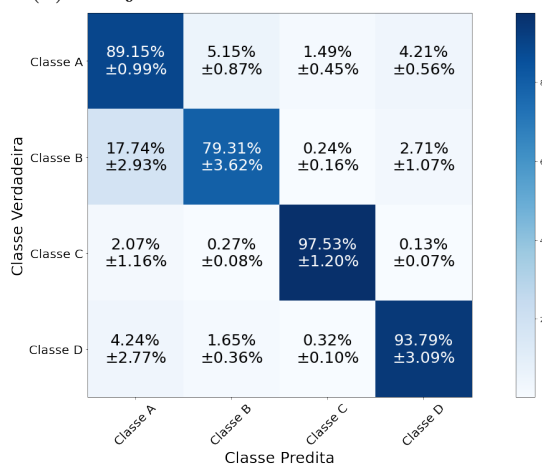
Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
50	$82,71 \pm 4,59$	$84,23 \pm 4,27$	$76,98 \pm 5,57$	$83,07 \pm 4,51$
75	$85,80 \pm 3,51$	$86,99 \pm 2,31$	$80,43 \pm 3,49$	$86,10 \pm 3,27$



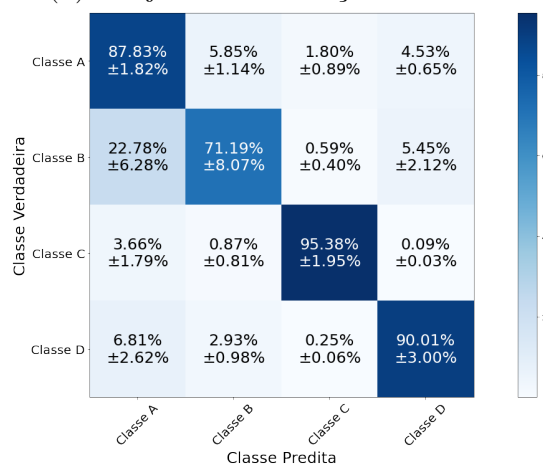
(a) Conjunto de treinamento LSTM-50.



(b) Conjunto de validação LSTM-50.



(c) Conjunto de treinamento LSTM-75.



(d) Conjunto de validação LSTM-75.

Figura 5.18: Matriz de Confusão dos modelos temporais LSTM com 50 e 75 células (águas profundas).

Avaliação dos Modelos

Após o desenvolvimento dos modelos LOFAR e temporal MLP, CNN e LSTM é importante analisar os resultados obtidos. Os modelos escolhidos foram:

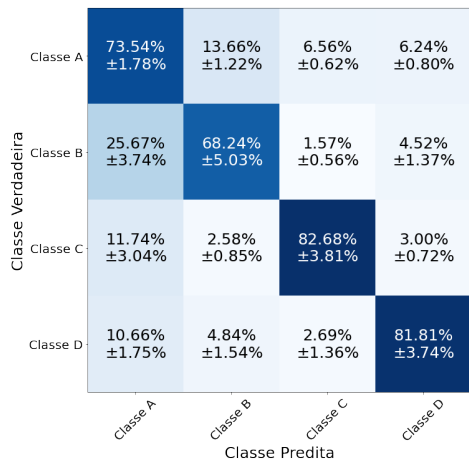
- MLP: Os modelos LOFAR e temporal possuem 30 neurônios na camada oculta.
- CNN: Os modelos LOFAR e temporal possuem duas camadas convolucionais de 8 filtros, uma camada *fully connected* de 20 neurônios e um *kernel* 3.
- LSTM: O modelo temporal possui 75 células na camada oculta.

A Figura 5.19 apresenta a matriz de confusão do conjunto de teste dos modelos LOFAR e temporal das redes mencionadas acima. Esses resultados em conjunto com as informações de Acurácia, Precisão e *Recall* do conjunto de teste apresentados na Tabela 5.14, além de todas as informações discutidas nas escolhas dos modelos, nos permite dizer que:

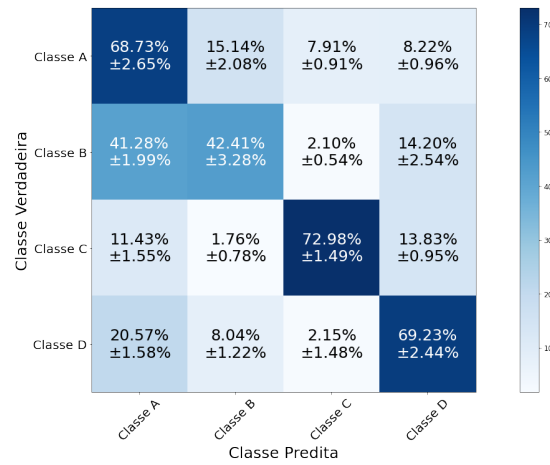
- O modelo temporal CNN não agrega valor em relação ao desempenho apresentado no modelo MLP. O valor médio é um pouco diferente, mas não há ganho no modelo convolucional, dada as incertezas apresentadas. Além do mais, não podemos esquecer a maior complexidade dos modelos convolucionais.
- No modelo LOFAR, a CNN não apresenta grandes vantagens em relação a rede MLP. O valor médio do índice SP é um pouco melhor, mas não há ganho considerando a barra de erro. Vale ressaltar que não temos dados suficientes para avaliar melhor esta diferença de desempenho.
- O modelo temporal usando a rede LSTM obteve o melhor desempenho entre os modelos abordados. Os valores são bem superiores aos apresentados pelos modelos LOFAR e temporal com a rede MLP. Em comparação com o modelo LOFAR CNN, o modelo temporal LSTM tem ligeira vantagem.

Tabela 5.14: Resultados experimentais do conjunto de teste da fusão dos modelos utilizando a rede MLP (águas profundas)

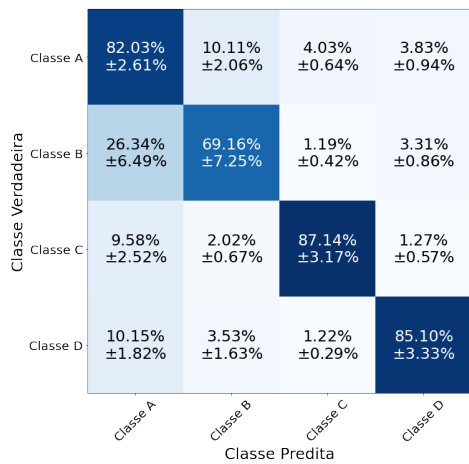
Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
<i>MLP(LOFAR)</i>	76,42 ± 2,69	74,91 ± 1,21	65,26 ± 1,47	76,56 ± 2,66
<i>CNN(LOFAR)</i>	80,64 ± 2,81	81,51 ± 2,03	73,20 ± 2,52	80,85 ± 2,68
<i>MLP(Tempo)</i>	62,61 ± 1,53	66,35 ± 1,81	54,79 ± 1,68	63,33 ± 1,39
<i>CNN(Tempo)</i>	64,49 ± 1,65	54,60 ± 2,23	53,36 ± 1,71	65,24 ± 1,92
<i>LSTM(Tempo)</i>	85,55 ± 3,28	86,85 ± 2,06	80,33 ± 3,11	85,89 ± 3,02



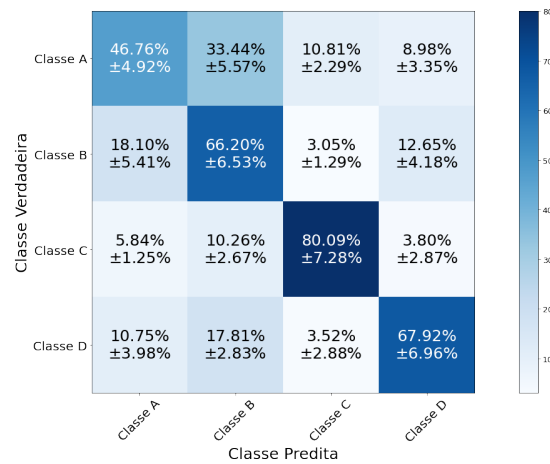
(a) Modelo MLP (LOFAR).



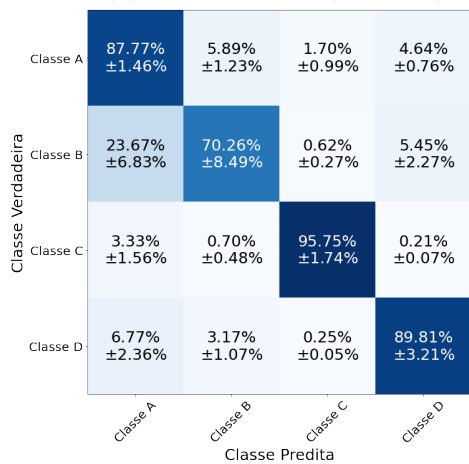
(b) Modelo MLP (Temporal).



(c) Modelo CNN (LOFAR).



(d) Modelo CNN (Temporal).



(e) Modelo LSTM (Temporal).

Figura 5.19: Matriz de Confusão do conjunto de teste do modelo MLP, CNN e LSTM (águas profundas).

Modelo de Fusão Proposto

Aqui iremos analisar o desenvolvimento e os resultados da fusão dos modelos temporal e LOFAR das redes MLP, CNN e LSTM. As fusões implementadas foram: MLP (LOFAR) com MLP (tempo), CNN (LOFAR) com CNN (tempo), MLP (LOFAR) com LSTM (tempo) e CNN (LOFAR) com LSTM (tempo).

Neste Seção, apenas alguns dados serão apresentados com o propósito de comparar os modelos desenvolvidos. Para maiores detalhes, o desenvolvimento das fusões pode ser vistas no Apêndice A. Os modelos LOFAR e temporal utilizados são os citados na última subseção (5.3.1). Lembrando que os modelos LOFAR e temporal são entradas para uma rede de fusão com a arquitetura MLP e uma camada oculta. A variação da quantidade de neurônios da camada oculta é 10, 30, 50 e 60, quando os modelos LOFAR e temporal não utilizam modelos profundos, e quando os modelos profundos são utilizados a variação do número de neurônios da rede de fusão é 10, 30, 50, 100. A Tabela 5.15 mostra o resultado do índice SP para o conjunto de validação de cada uma das fusões de acordo com a variação de neurônios descrita e a quantidade de parâmetros treináveis/não-treináveis. Os parâmetros treináveis são da rede de fusão e os não-treináveis são dos modelos LOFAR e temporal, pois estão congelados durante o processo de treinamento da fusão. A informação “-” mostra o resultado sem a rede de fusão, conforme método descrito na Subseção 4.5.

Tabela 5.15: Resultados experimentais do índice SP e dos parâmetros da fusão dos modelos nos conjuntos de validação (águas profundas)

Neurônios da Rede de fusão	MLP com MLP	CNN com CNN	MLP com LSTM	CNN com LSTM
	SP(%)			
-	66,63 ± 1,87	78,29 ± 2,25	88,18 ± 3,28	88,84 ± 3,38
10	75,86 ± 2,38	81,37 ± 2,62	89,07 ± 3,30	88,81 ± 3,47
30	78,71 ± 2,43	82,22 ± 2,39	89,50 ± 3,39	88,66 ± 3,27
50	79,15 ± 2,43	81,94 ± 2,63	89,64 ± 3,21	88,49 ± 3,35
60	79,51 ± 2,25			
100		81,61 ± 2,69	89,54 ± 3,50	88,57 ± 3,33
Parâmetros Treináveis - Não Treináveis				
-	244 - 42.780	14.596 - 464	424 - 35.130	6.604 - 10.632
10	654 - 42.780	36.534 - 464	1.104 - 35.130	16.554 - 10.632
30	1.954 - 42.780	109.594 - 464	3.304 - 35.130	49.654 - 10.632
50	3.254 - 42.780	182.654 - 464	5.504 - 35.130	82.754 - 10.632
60	3.904 - 42.780			
100		365.304 - 464	11.004 - 35.130	165.504 - 10632

Na rede de fusão com os modelos LOFAR e temporal utilizando as redes MLP como entrada, dois modelos se destacam, o de 10 e 30 neurônios. Acima de 30 neurônios, não há uma melhora na margem de erro em comparação com outros modelos. A distinção entre os dois resultados pode ser vista com mais detalhe no Apêndice A. Em suma, a Tabela 5.16 mostra os resultados do índice SP, acurácia, precisão e *recall* dos modelos de fusão. É possível, observar um desempenho superior na acurácia e precisão do modelo com 30 neurônios.

Na rede de fusão com os modelos LOFAR e temporal utilizando as redes CNN

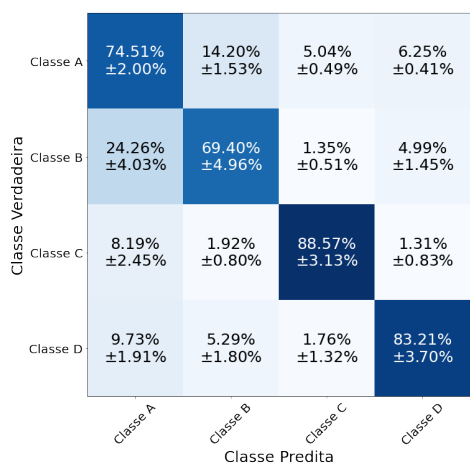
como entrada, nota-se que não há melhora na margem de erro com o aumento do número de neurônios da rede e o valor médio gira em torno de 81%. A fusão sem a rede diminui um pouco o valor médio do índice SP. A Tabela 5.16 mostra que sem a rede de fusão, a margem de erro dos valores é um pouco menor, mas ocorre uma perda também no valor médio.

Na rede de fusão utilizando como entrada o modelo LOFAR com a rede MLP e o modelo temporal com a rede LSTM, vemos que a utilização ou não da rede, não apresenta uma mudança nos valores do índice SP. Essa análise também pode ser observada na rede de fusão que utiliza o modelo LOFAR CNN em conjunto com o modelo temporal LSTM. Nota-se na Tabela 5.16 que além do índice SP, os outros resultados também não mostram uma diferença com a adição da rede na composição da fusão. O Apêndice A.1 mostra com mais detalhes esses resultados. Com base nos resultados apresentados, as redes de fusão utilizadas para avaliar o conjunto de teste serão: As redes com 10 neurônios na camada oculta, apesar de não mostrarem uma melhora em relação a fusão dos modelos sem a rede de fusão, elas serão mantidas para aferir o método proposto. Apenas a rede de fusão do modelo MLP (LOFAR) com MLP (tempo) utilizará 30 neurônios na camada oculta, pois nela é possível ver uma melhora nos resultados.

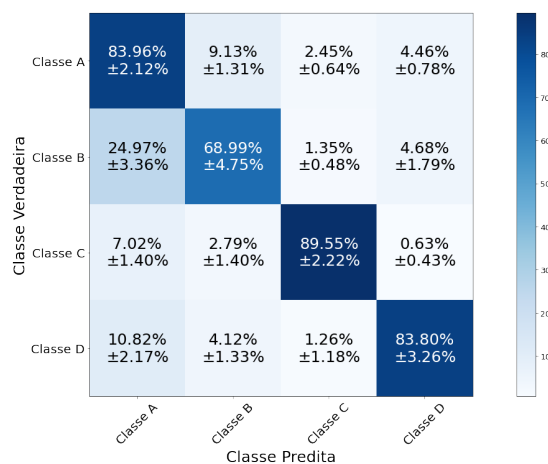
Tabela 5.16: Resultados experimentais do conjunto de validação da fusão dos modelos (águas profundas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
MLP com MLP				
10	75,86 ± 2,38	70,31 ± 1,06	61,93 ± 1,44	76,15 ± 2,37
30	78,71 ± 2,43	76,36 ± 1,64	67,58 ± 1,80	78,92 ± 2,40
CNN com CNN				
–	78,29 ± 2,25	77,21 ± 1,23	68,05 ± 1,78	78,49 ± 2,22
10	81,37 ± 2,62	82,80 ± 2,25	75,06 ± 2,92	81,57 ± 2,55
MLP com LSTM				
–	88,18 ± 3,28	86,75 ± 2,27	80,34 ± 3,37	88,34 ± 3,12
10	89,07 ± 3,30	88,24 ± 2,13	82,24 ± 3,07	89,22 ± 3,14
CNN com LSTM				
–	88,84 ± 3,38	88,78 ± 2,29	83,35 ± 3,39	88,99 ± 3,24
10	88,81 ± 3,47	88,98 ± 2,60	83,62 ± 3,80	88,97 ± 3,34

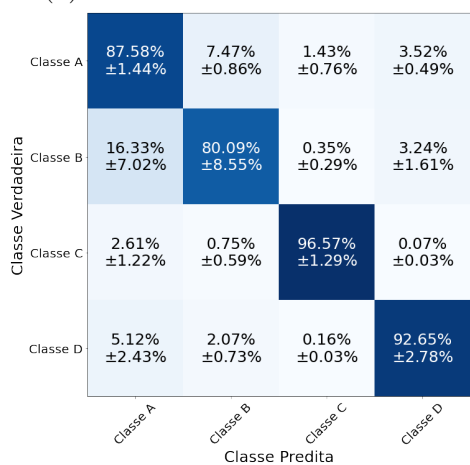
A Figura 5.20 apresenta a matriz de confusão do conjunto de validação da fusão dos modelos. De forma geral, todas elas apresentaram resultados melhores de classificação em relação aos resultados apresentados individualmente em cada modelo (LOFAR e temporal). A adição do modelo temporal LSTM claramente melhorou os resultados da fusão, basta observar as Figuras 5.20c e 5.20d. A classe C, por exemplo, teve uma melhora de cerca de 7% com o uso da rede LSTM. Mesmo com a fusão, nenhum modelo conseguiu mitigar a confusão entre as classes A e B, apesar de ter reduzido bastante os Falsos Positivos da classe A.



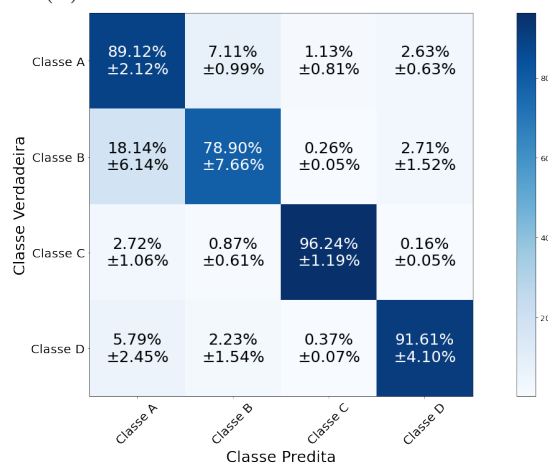
(a) Fusão do modelo MLP com MLP.



(b) Fusão do modelo CNN com CNN.



(c) Fusão do modelo MLP com LSTM.



(d) Fusão do modelo CNN com LSTM.

Figura 5.20: Matriz de Confusão do conjunto de validação da fusão dos modelos (águas profundas).

Outra maneira de comparar as fusões dos modelos é analisando a distribuição de probabilidade de saída. A Figura 5.21 mostra uma matriz de histogramas da saída da fusão dos modelos MLP com MLP e MLP com LSTM do conjunto de validação. A fusão MLP com LSTM apresentou uma frequência maior de valores próximos de “1” para os Verdadeiros Positivos e próximos de “0” para os Verdadeiros Negativos. Assim como citado anteriormente, o modelo apresenta dificuldades de distinguir as classes A e B. Essa observação pode ser vista na diagonal principal da matriz de histogramas.

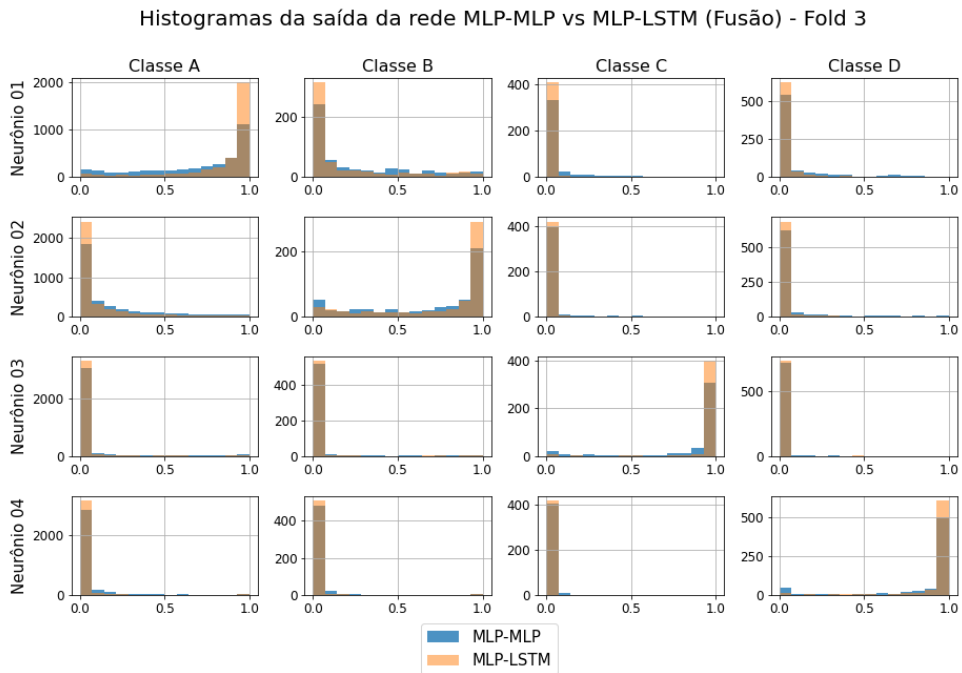


Figura 5.21: Histogramas da saída da fusão dos modelos MLP-MLP e MLP-LSTM do conjunto de validação (águas profundas).

A Figura 5.22 mostra uma matriz de histogramas da saída dos modelos CNN com CNN e CNN com LSTM do conjunto de validação. Os Verdadeiros Positivos da fusão utilizando o modelo LSTM estão na diagonal principal. Ela apresenta uma frequência maior de valores próximos de “1”, em comparação ao outro modelo de fusão. Para os Verdadeiros Negativos esse comportamento se repete para valores próximos de “0”.

A Figura 5.23 apresenta a matriz de confusão do conjunto de teste da fusão dos modelos. O desempenho do conjunto de teste foi semelhante ao apresentado no conjunto de validação, com o desempenho um pouco abaixo. A adição dos modelos recorrentes na composição da fusão apresentou resultados melhores que a fusão que utiliza os modelos MLP e convolucionais como entrada temporal.

A Tabela 5.17 apresenta o resultado do conjunto de teste do índice SP, Acurácia, Precisão e *Recall* das redes de fusão. Os modelos híbridos mantém o valor médio

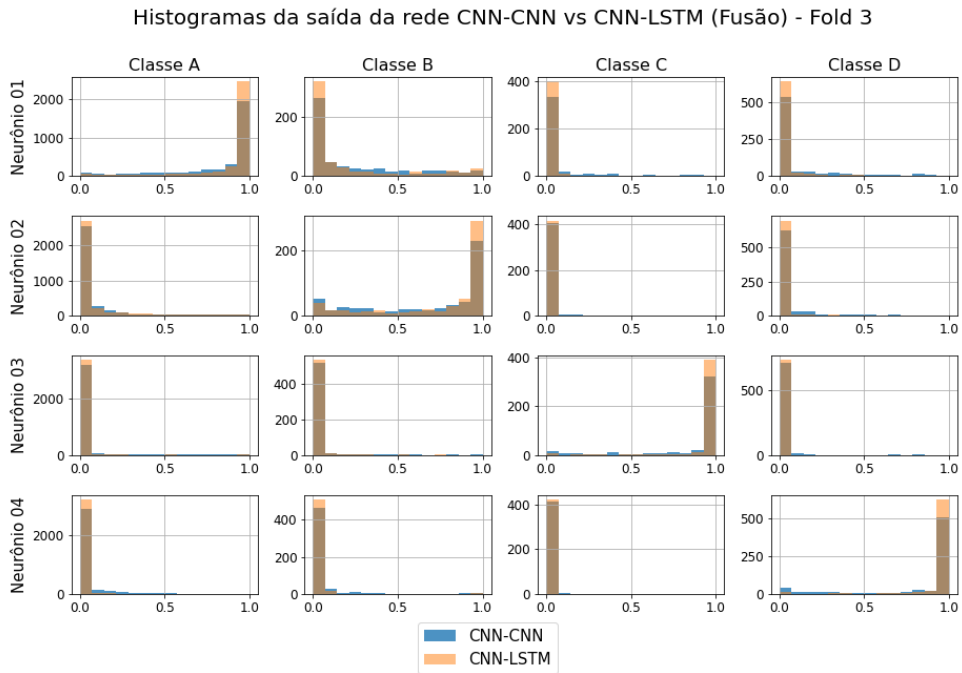


Figura 5.22: Histogramas da saída da fusão dos modelos CNN-CNN e CNN-LSTM do conjunto de validação (águas profundas).

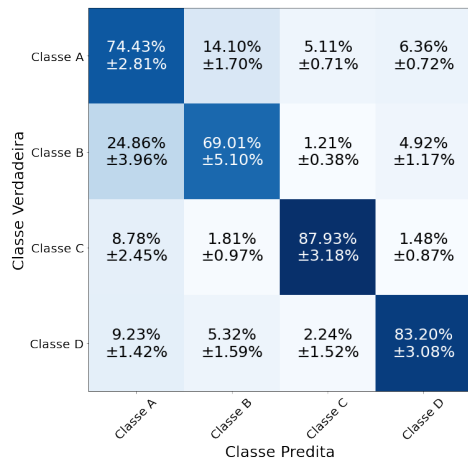
acima de 80% em todos os índices, mostrando uma superioridade no desempenho.

Tabela 5.17: Resultados experimentais do conjunto de teste da fusão dos modelos (águas profundas)

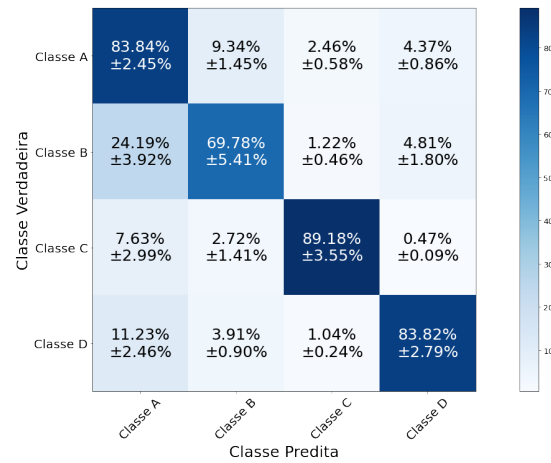
Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
<i>MLP – MLP</i>	78,44 ± 2,48	76,21 ± 2,22	67,33 ± 2,27	78,64 ± 2,44
<i>CNN – CNN</i>	81,45 ± 2,71	82,78 ± 1,94	75,19 ± 2,49	81,65 ± 2,62
<i>MLP – LSTM</i>	88,76 ± 2,81	88,02 ± 1,70	82,00 ± 2,34	88,92 ± 2,66
<i>CNN – LSTM</i>	88,88 ± 3,04	89,01 ± 1,94	83,69 ± 2,77	89,04 ± 2,90

A Tabela 5.18 mostra o resultado do índice SP do conjunto de validação e teste dos modelos escolhidos em cada técnica nos dois domínios, além de suas fusões. Esses resultados em conjunto com as informações discutidas anteriormente nos permite dizer que:

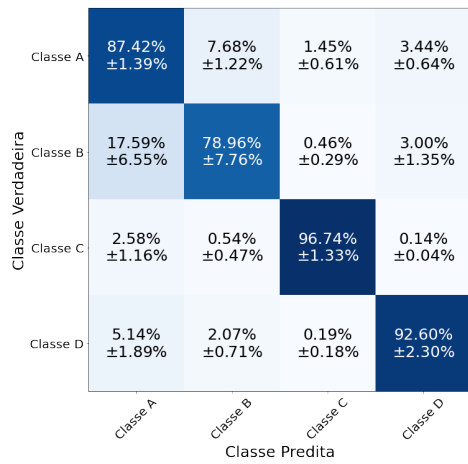
- O uso da mesma tecnologia como entrada da rede de fusão (exemplo: MLP com MLP) não apresenta vantagens em relação ao uso individual dos modelos. Por exemplo, a fusão MLP com MLP não melhora os valores obtidos com o modelo LOFAR com a rede MLP.
- O uso de tecnologias híbridas como entrada da rede de fusão (exemplo: MLP com LSTM) apresenta ganhos no valor médio, entretanto, o bom desempenho do modelo temporal LSTM impossibilita afirmar que rede de fusão melhora os resultados, dada a margem de erro.



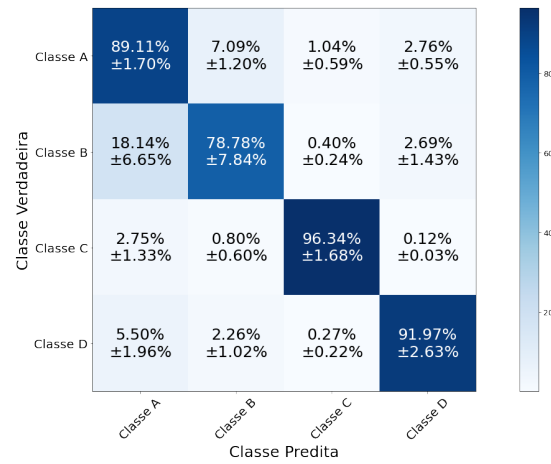
(a) Fusão do modelo MLP com MLP.



(b) Fusão do modelo CNN com CNN.



(c) Fusão do modelo MLP com LSTM.



(d) Fusão do modelo CNN com LSTM.

Figura 5.23: Matriz de Confusão do conjunto de teste da fusão dos modelos (águas profundas).

- Não há diferença nos resultados ao substituir a rede MLP pela CNN na entrada LOFAR da rede de fusão.

Tabela 5.18: Resultados experimentais do índice SP dos modelos MLP, CNN , LSTM e da fusão nos conjuntos de validação e teste (águas profundas)

Modelos	Índice SP (%)					
	Validação			Teste		
	LOFAR	Temporal	Fusão	LOFAR	Temporal	Fusão
MLP	76,30 ± 2,77	63,35 ± 0,79		76,42 ± 2,69	62,61 ± 1,53	
CNN	80,86 ± 2,90	64,70 ± 1,68		80,64 ± 2,81	64,49 ± 1,65	
LSTM		85,79 ± 3,51			85,55 ± 3,28	
MLP - MLP			78,71 ± 2,43			78,44 ± 2,48
CNN - CNN			81,37 ± 2,62			81,45 ± 2,71
MLP - LSTM			89,07 ± 3,30			88,76 ± 2,81
CNN - LSTM			88,81 ± 3,47			88,88 ± 3,04

5.3.2 Águas Rasas (31 Classes)

Resultados para a Rede Neural MLP

A Figura 5.24 apresenta a curva do valor médio com margem de erro do índice SP aplicado sobre o conjunto de validação pela variação de neurônios da camada oculta da rede MLP. Analisando o comportamento das curvas, o índice SP tem uma ligeira melhora com o aumento do número de neurônios.

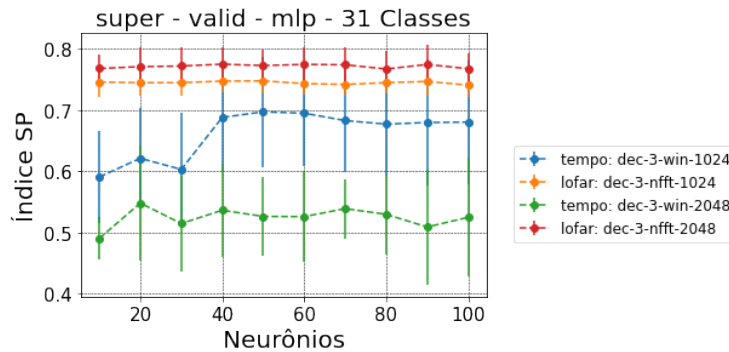


Figura 5.24: Gráfico do índice SP pela variação do número de neurônios (águas rasas).

A Tabela 5.19 apresenta o valor médio e o erro do índice SP de acordo com o número de neurônios. É possível observar que o modelo LOFAR obteve resultados melhores que o temporal. No LOFAR, a janela de 2048 pontos apresenta um valor médio ligeiramente melhor do que os valores obtidos com a janela de 1024 pontos, mas considerando a variância estatística os valores se sobrepõem. Esta análise pode ser replicada para o modelo temporal, sendo que neste domínio, a janela com 1024 pontos obteve os melhores valores. Em comparação com os resultados apresentados no conjunto de validação dos dados de águas profundas (vide Tabela 5.4), os dados de águas rasas no modelo temporal apresentam uma margem de erro maior.

Tabela 5.19: Resultados experimentais do índice SP para o modelo MLP do conjunto de validação (águas rasas)

Neurônios	Índice SP (%)			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
10	74,60 ± 2,54	76,81 ± 2,36	59,06 ± 7,51	49,02 ± 3,50
20	74,47 ± 2,12	77,10 ± 3,08	62,12 ± 8,17	54,80 ± 9,34
30	74,49 ± 2,24	77,20 ± 3,08	60,30 ± 9,26	51,46 ± 7,79
40	74,73 ± 1,78	77,51 ± 2,85	68,80 ± 8,9	53,62 ± 7,54
50	74,77 ± 2,03	77,28 ± 2,67	69,72 ± 9,02	52,62 ± 6,36
60	74,33 ± 1,83	77,47 ± 2,89	69,49 ± 8,59	52,58 ± 7,40
70	74,19 ± 1,93	77,41 ± 2,92	68,28 ± 8,36	53,89 ± 4,84
80	74,49 ± 1,81	76,72 ± 2,94	67,71 ± 8,53	52,96 ± 6,55
90	74,69 ± 2,37	77,46 ± 3,16	67,97 ± 10,25	50,89 ± 9,44
100	74,08 ± 1,85	76,77 ± 2,53	68,03 ± 10,09	52,47 ± 9,72

Como mencionado anteriormente na Seção 5.3.2, é importante analisar as curvas de treinamento (apresentadas no Apêndice C.3.2) e a quantidade de parâmetros. O número total de amostras do *dataset* para as janelas de 1024 pontos é 48.060. Como o conjunto de dados é subdividido em 10 *folds*, em cada iteração da validação cruzada 80% dos dados são para treinamento, 10% para validação e 10% para teste. Portanto, 38.448 amostras são destinadas ao treinamento. A classe A com 6.272 amostras para o treinamento é a classe minoritária. A classe B é a classe majoritária com 32.176 amostras para o treinamento. Com as janelas de 2048 pontos, esses valores apresentados reduzem pela metade. A Tabela 5.20 apresenta o número de parâmetros de acordo com a variação do número de neurônios da camada oculta da rede MLP. Assim como mencionado no conjunto de águas profundas, o número de parâmetros dos modelos serão escolhidos com parcimônia.

Tabela 5.20: Quantidade de parâmetros para cada modelo MLP (águas rasas)

Neurônios	Parâmetros			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
10	4.274	8.274	10.514	20.754
20	8.524	16.524	21.004	41.484
30	12.774	24.774	31.494	62.214
40	17.024	33.024	41.984	82.944
50	21.274	41.274	52.474	103.674
60	25.524	49.524	62.964	124.404
70	29.774	57.774	73.454	145.134
80	34.024	66.024	83.944	165.864
90	38.274	74.274	94.434	186.594
100	42.524	82.524	104.924	207.324

Na Figura 5.25 são representados os gráficos de violino do índice SP dos dados do conjunto de validação pela variação de neurônios da camada oculta do modelo. Como mencionado anteriormente na Subseção B.1.1, os dados considerados serão das janelas de 1024 pontos. No modelo LOFAR, todos os valores do índice SP são similares se considerarmos a barra de erro. Sendo assim, pensando na menor quantidade de parâmetros, podemos escolher o modelo com 10 neurônios na camada oculta, o qual tem o índice SP de $(74, 60 \pm 2, 54)\%$. Os resultados de classificação deste modelo são: Acurácia de $(77, 73 \pm 2, 00)\%$, Precisão de $(66, 50 \pm 1, 81)\%$ e *Recall* de $(74, 69 \pm 2, 46)\%$.

A Figura 5.26 mostra uma matriz de histogramas da saída do modelo LOFAR MLP com 10 neurônios na camada oculta. Na coluna “Classe B” é possível observar que o modelo identifica bem esses dados. Esse fator pode ser atribuído à quantidade maior de amostras da classe. Claramente vemos uma confusão no modelo ao tentar classificar os sinais da classe A. Muito dos valores das saídas estão concentrados na parte central do histograma, o que indica uma imprecisão do modelo.

A Figura 5.27 mostra a matriz de confusão dos conjuntos de treinamento e va-

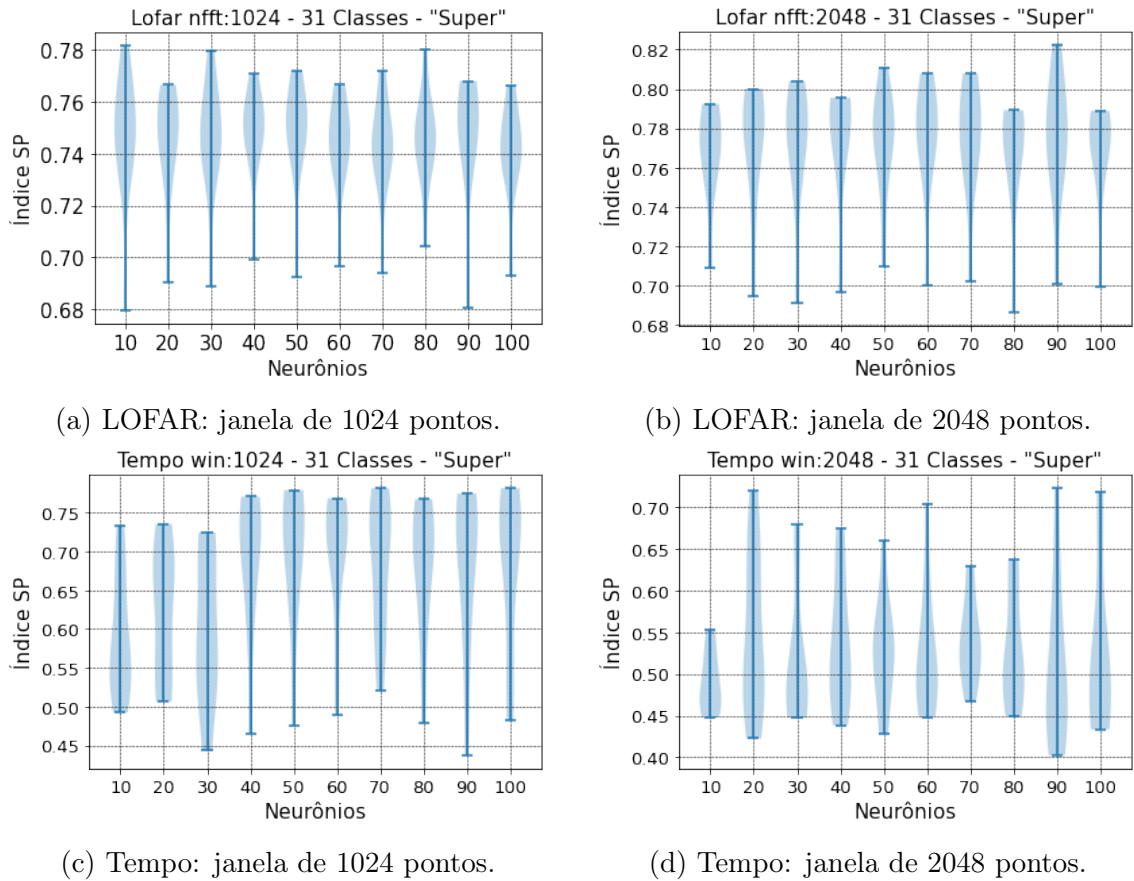


Figura 5.25: Gráfico de Violino do desempenho dos modelos MLP no conjunto de validação (águas rasas).

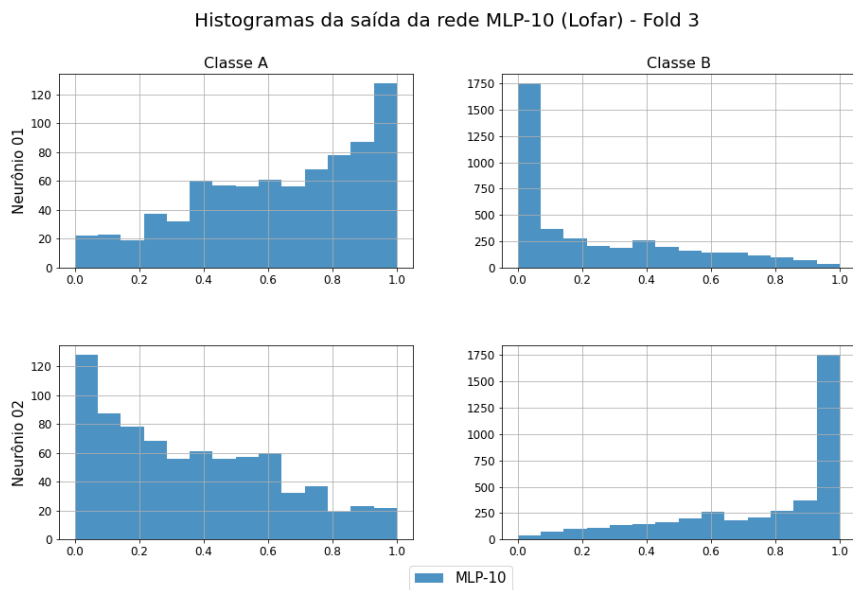
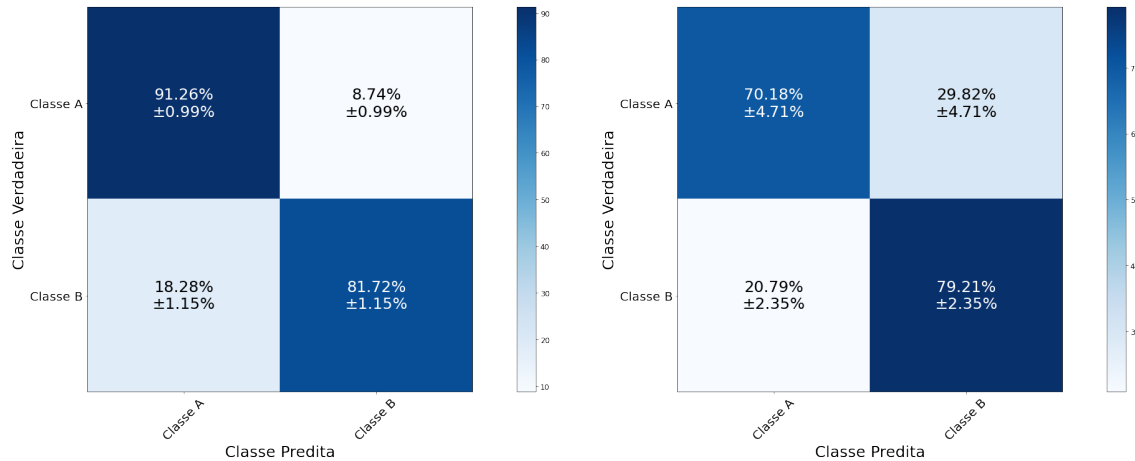


Figura 5.26: Histogramas da saída do modelo LOFAR MLP com 10 neurônios (águas rasas).

lidação do modelo LOFAR com 10 neurônios. O maior problema de classificação é com a classe A. O conjunto de validação obteve apenas $(70,18 \pm 4,71)\%$ de acurácia para a classe. O alto resultado do índice SP é sustentado pela ótima classificação da classe B, com acurácia de $(79,21 \pm 2,35)\%$ no conjunto de validação. As curvas de treinamento do modelo encontram-se no Apêndice C.2.1.



(a) Conjunto de treinamento MLP-10.

(b) Conjunto de validação MLP-10.

Figura 5.27: Matriz de Confusão do modelo LOFAR MLP com 10 neurônios (águas rasas).

Para o modelo temporal, deve-se analisar de forma mais detalhada os resultados de 10 e 20 neurônios na camada oculta pois, observando os parâmetros da Tabela 5.20, vemos que esses neurônios apresentam parâmetros compatíveis com o número de amostras do treinamento. Os índices SP deles são $(59,06 \pm 7,51)\%$ e $(62,12 \pm 8,17)\%$ respectivamente (vide a Tabela 5.19). O gráfico de violino da Figura 5.25c mostra que os resultados com 10 neurônios estão mais concentrados na parte inferior violino, já com 20 neurônios os resultados do índice SP nos *folds* estão mais distribuídos. A Figura 5.28 mostra a matriz de histogramas da saída dos modelos temporais MLP com 10 e 20 neurônios na camada oculta. As distribuições estão mais centralizadas, o que era de se esperar, tendo em vista que os resultados do SP estão em torno de 60%. Observando a diagonal principal, vemos que o modelo com 20 neurônios na camada oculta apresenta mais eventos próximos ao extremo da distribuição do que os modelos com 10 neurônios.

A Figura 5.29 mostra a matriz de confusão dos conjuntos de treinamento e validação dos modelos temporais com 10 e 20 neurônios. Os resultados de ambos os modelos possuem uma barra de erro muito grande associada a identificação da classe A. Entretanto, há uma pequena melhora no valor médio de acertos e na margem de erro quando usamos 20 neurônios. Mesmo assim, ela continua apresentando o pior resultado de classificação. As curvas de treinamento dos modelos encontram-se no Apêndice C.2.1.

Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 3

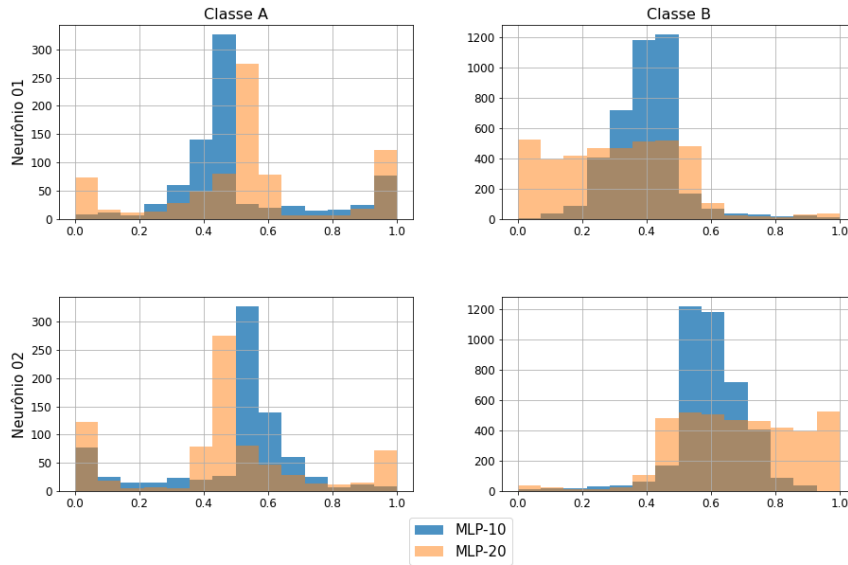


Figura 5.28: Histogramas da saída dos modelos temporais MLP-10 e MLP-20, *fold* 3 (águas rasas).

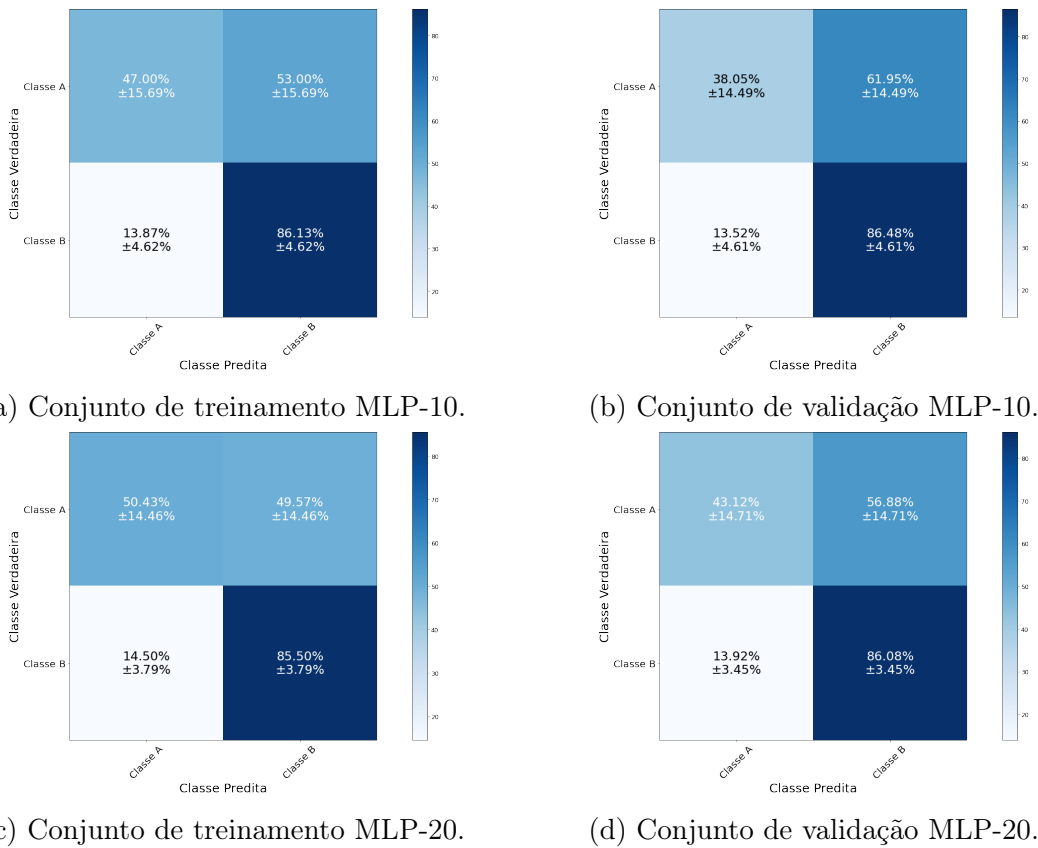


Figura 5.29: Matriz de Confusão dos modelos temporais MLP com 10 e 20 neurônios (águas rasas).

Dada a análise do treinamento dos modelos temporais com 10 e 20 neurônios na camada oculta, é possível observar um desempenho um pouco superior no modelo com 20 neurônios, mas como a quantidade de amostras da classe A para o treinamento não ultrapassa sete mil, o modelo com 10 neurônios será utilizado na fusão por causa da quantidade reduzida de parâmetros. A Tabela 5.21 mostra o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos. Ela reforça a conclusões obtidas pelas análises anteriores.

Tabela 5.21: Resultados experimentais da análise Temporal (1024) para o modelo MLP para o conjunto de validação (águas rasas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
10	59,06 ± 7,51	78,58 ± 2,37	61,68 ± 3,15	62,27 ± 5,53
20	62,12 ± 8,17	79,07 ± 2,10	62,98 ± 4,01	64,60 ± 6,27

Resultados para a Rede Neural CNN

A Figura 5.30 apresenta a curva do valor médio com a margem de erro do índice SP aplicado sobre o conjunto de validação para cada modelo convolucional implementado. Os modelos são implementados tendo como base a variação dos parâmetros já mencionado na Seção 5.3.1. Assim como ocorreu na rede MLP, o modelo LOFAR obteve resultados melhores que o temporal. Em ambos, a presença de duas camadas convolucionais para extração de características melhora a classificação.

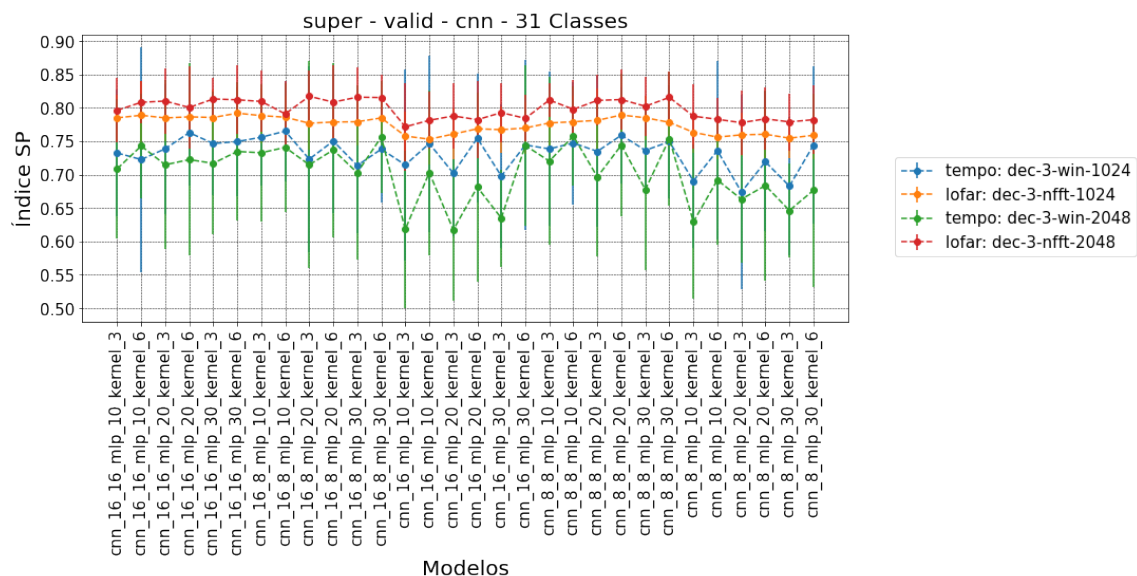


Figura 5.30: Gráfico do índice SP pela variação do número de modelos CNN (águas rasas).

A Tabela 5.22 (versão completa na Tabela C.3) mostra o valor médio, com o erro associado, do índice SP aplicado sobre o conjunto de validação de alguns modelos

gerados a partir da variação dos parâmetros citados anteriormente. Os resultados do valor médio do índice SP melhoram com o aumento da quantidade de filtros das camadas CNN e o número de neurônios na camada MLP, entretanto, devido à margem de erro, os valores são semelhantes.

Tabela 5.22: Resultados experimentais do índice SP para o modelo CNN do conjunto de validação (águas rasas)

Modelos	Índice SP (%)			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
(16,16,10,3)	78,51 ± 3,50	79,64 ± 4,82	73,25 ± 9,44	70,88 ± 10,41
(16,16,10,6)	78,90 ± 5,07	80,84 ± 3,16	72,26 ± 16,80	74,30 ± 7,77
(16,16,30,3)	78,56 ± 4,21	81,33 ± 3,17	74,67 ± 7,75	71,70 ± 10,68
(16,8,10,6)	78,60 ± 3,22	79,10 ± 4,85	76,52 ± 7,55	74,09 ± 9,65
(16,8,30,3)	77,93 ± 3,54	81,60 ± 4,40	71,37 ± 10,11	70,30 ± 13,03
(16,-,10,3)	75,78 ± 3,72	77,17 ± 6,55	71,46 ± 14,31	61,80 ± 11,82
(8,8,10,3)	77,74 ± 3,84	81,17 ± 2,48	73,86 ± 11,55	72,03 ± 12,57
(8,8,20,3)	78,13 ± 4,84	81,12 ± 3,80	73,49 ± 11,26	69,66 ± 11,84
(8,8,20,6)	78,93 ± 3,03	81,25 ± 4,44	75,91 ± 7,19	74,43 ± 10,71
(8,8,30,6)	77,84 ± 3,67	81,58 ± 3,92	74,94 ± 8,43	75,39 ± 10,09
(8,-,10,6)	75,62 ± 3,24	78,29 ± 3,27	73,61 ± 13,35	69,20 ± 9,74

Assim como ocorreu na criação dos modelos para o conjunto de dados de águas profundas na Seção 5.3.1, a camada *MaxPolling* (descrita com mais detalhes no Apêndice B.1.2) foi utilizada para reduzir a quantidade de parâmetros dos modelos e maximizar a extração de características. A Tabela 5.23 (versão completa na Tabela C.2) mostra o número de parâmetros dos modelos criados com a busca paramétrica. As janelas de 2048 pontos para a maioria dos modelos apresentam uma quantidade de parâmetros bem acima do número de amostras do *dataset*, o que dificulta a escolha do modelo. Sendo assim, os modelos com janelas de 1024 pontos são mais apropriados pois permitem uma maior gama de escolhas.

Tabela 5.23: Quantidade de parâmetros para cada modelo CNN (águas rasas)

Modelos	Parâmetros			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
(16,16,30,6)	97.818	194.818	124.698	247.578
(16,8,10,6)	16.942	32.942	21.422	41.902
(16,8,30,3)	48.610	96.610	62.050	123.490
(16,8,30,6)	49.042	98.042	62.482	123.922
(16,-,10,3)	32.118	64.118	82.038	163.958
(16,-,30,6)	96.266	192.266	246.026	491.786
(8,8,10,3)	16.286	32.286	20.766	41.246
(8,8,20,3)	32.336	64.336	41.296	82.256
(8,8,20,6)	32.552	64.552	41.512	82.472
(8,8,30,3)	48.386	96.386	61.826	123.266
(8,8,30,6)	48.602	96.602	62.042	123.482
(8,-,30,3)	48.186	96.186	123.066	245.946
(8,-,30,6)	48.210	96.210	123.090	245.970

Os gráficos de violino das Figuras 5.31 e 5.34 apresentam os índices SP dos dados do conjunto de validação para cada modelo proposto com a variação de parâmetros da rede convolucional. Os valores são das janelas de 1024 pontos dos

modelos LOFAR e temporal. Os modelos LOFAR que apresentam uma quantidade de parâmetros próximos a quantidade de amostras do *dataset* e altos valores médios de SP com baixo erro são os mesmos modelos apresentados no conjunto de dados de águas profundas, com duas camadas convolucionais de oito filtros cada, uma camada *fully conneted* com 10 neurônios e *Kernel* 3 (8,8,10,3). O modelo com 20 neurônios na camada *fully connected* também será analisado. Os valores do índice SP para os modelos com 10 e 20 neurônios são $(77, 74 \pm 3, 84)\%$ e $(78, 13 \pm 4, 84)\%$ respectivamente.

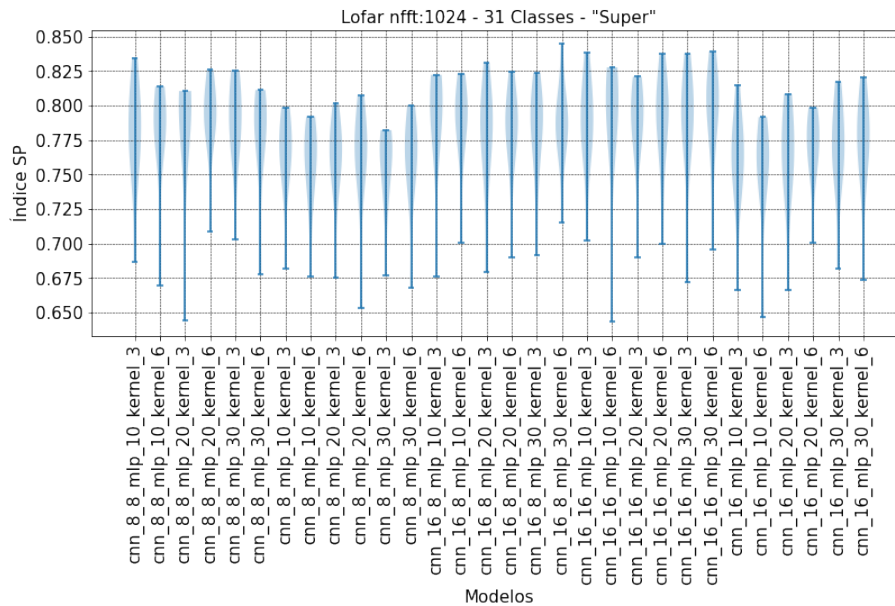


Figura 5.31: Gráfico de Violino do desempenho dos modelos LOFAR CNN no conjunto de validação (águas rasas).

A Figura 5.32 mostra a matriz de histogramas da saída dos modelos CNN citados acima com 10 e 20 neurônios na camada *fully connected*. A distribuição da classe B nos dois modelos estão concentradas nos extremos. Entretanto, a classe A apresenta uma distribuição um pouco mais dispersa em relação a classe B. Neste caso, podemos ver que o modelo com 20 neurônios teve melhores resultados.

A Figura 5.33 apresenta a matriz de confusão dos conjuntos de treinamento e validação dos modelos LOFAR convolucionais com 10 e 20 neurônios da camada *fully connected*. Se olharmos os conjuntos de treinamento e validação dos dois modelos, o modelo com 20 neurônios não apresentou uma melhora significativa em relação ao modelo com 10 neurônios. Durante o treinamento, a classe A teve um desempenho melhor do que a classe B, mas durante a validação o desempenho foi inferior, esse resultado está associado a baixa quantidade de amostras da classe A nos conjuntos de treinamento e validação. As curvas de treinamento dos modelos encontram-se no Apêndice C.2.2.

A Tabela 5.24 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall*

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 3

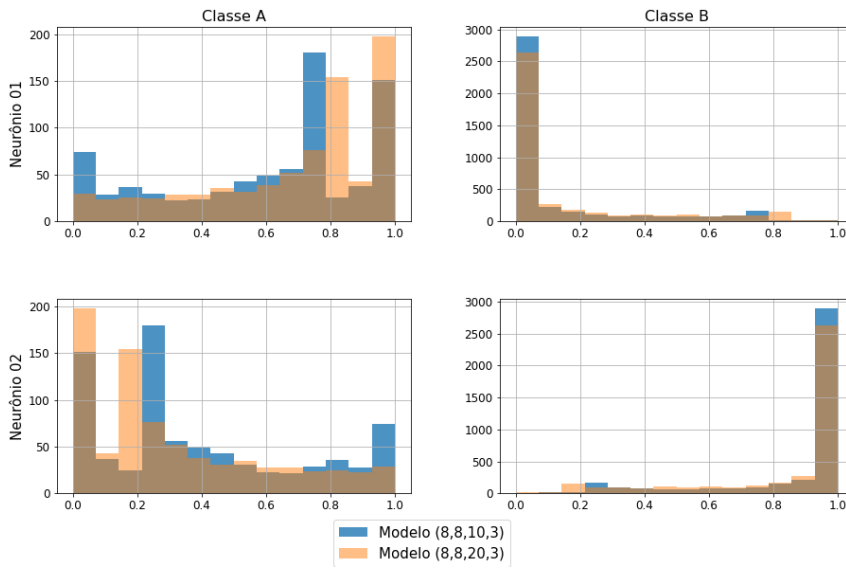


Figura 5.32: Histogramas da saída dos modelos LOFAR CNN (8,8,10,3) e (8,8,20,3), fold 3 (águas rasas).

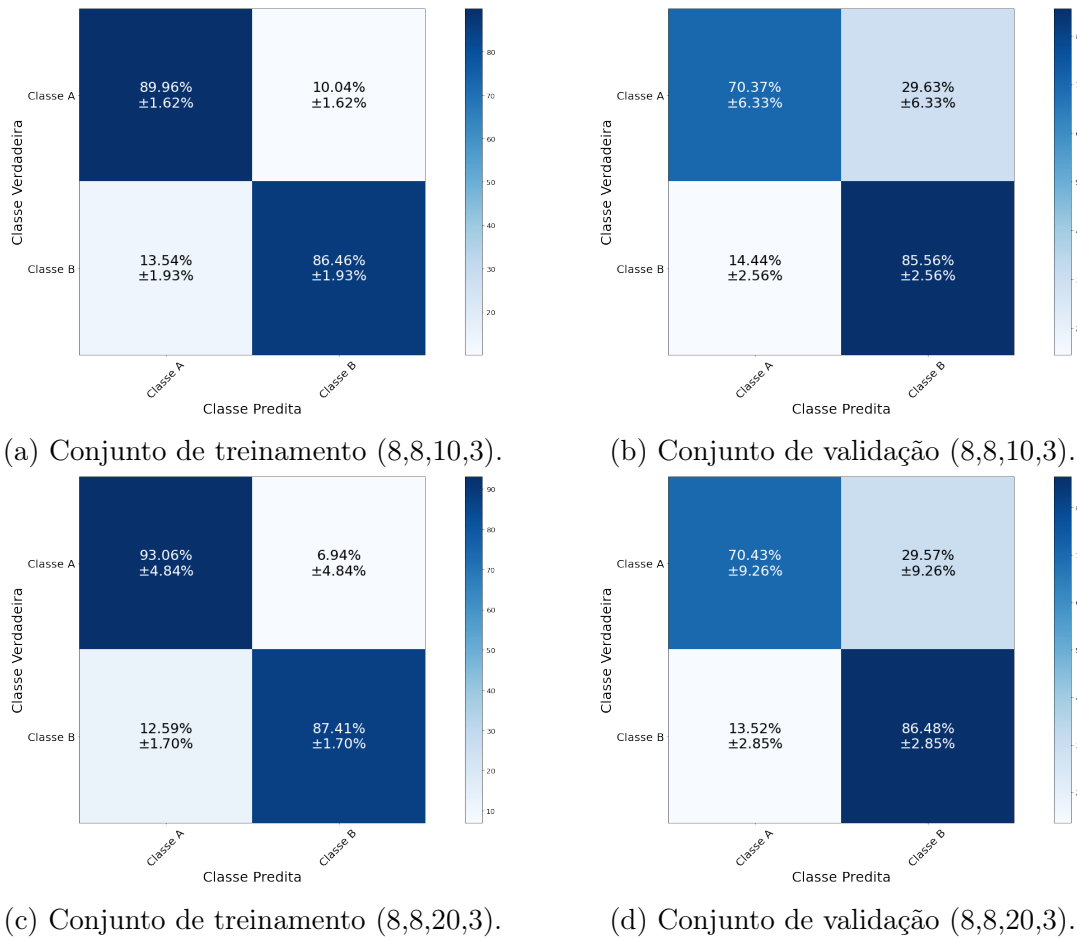


Figura 5.33: Matriz de Confusão dos modelos LOFAR CNN com 10 e 20 neurônios na camada *fully connected* (águas rasas).

dos modelos com 10 e 20 neurônios na camada *fully connected*. Os valores são semelhantes. Sendo assim, o modelo com 10 neurônios será utilizado na rede de fusão.

Tabela 5.24: Resultados experimentais da análise LOFAR (1024) dos modelos CNN para o conjunto de validação (águas rasas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
(8, 8, 10, 3)	77,74 ± 3,84	83,08 ± 2,61	71,36 ± 3,13	77,96 ± 3,71
(8, 8, 20, 3)	78,13 ± 4,84	83,85 ± 2,41	72,24 ± 3,34	78,45 ± 4,42

Os modelos temporais apresentam resultados do índice SP em torno de 70%. O que mais chama a atenção é a barra de erro que fica em cerca de 10%, o que mostra a incerteza durante o processo de treinamento. Dada a semelhança no comportamento dos valores, a quantidade de parâmetros se torna um fator importante na escolha do modelo. Assim como mencionado na análise no modelo LOFAR, o modelo com a menor quantidade de parâmetros é o modelo com duas camadas convolucionais de 8 filtros, uma camada *fully connected* de 10 neurônios e um *kernel* (20.766 parâmetros). Para ter certeza que essa escolha é a mais adequada, iremos fazer uma comparação com o modelo com 20 neurônios na camada *fully connected*, como analisado anteriormente. O índice SP dos modelos com 10 e 20 neurônios são (73,86 ± 11,55)% e (73,49 ± 11,26)% respectivamente.

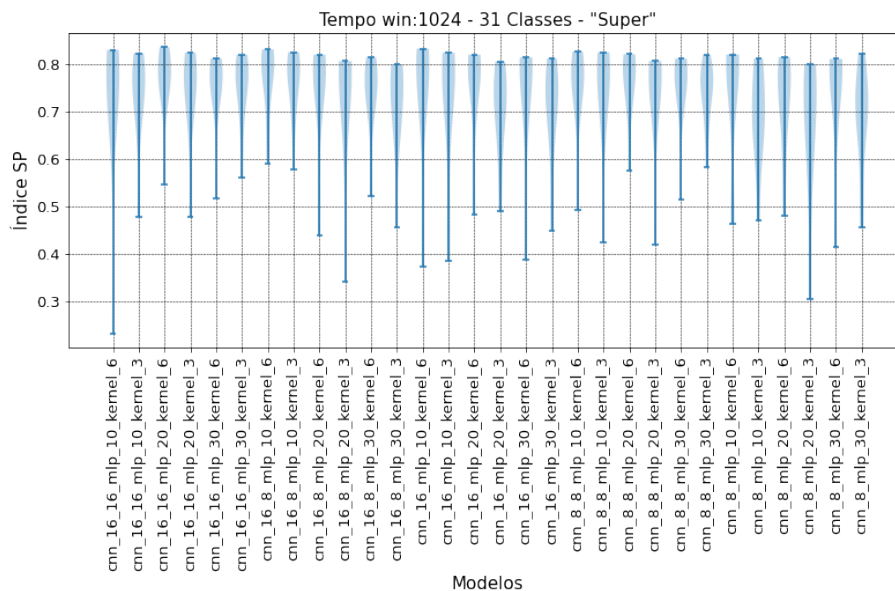


Figura 5.34: Gráfico de Violino do desempenho dos modelos temporais CNN no conjunto de validação (águas rasas).

A Figura 5.13 mostra a matriz de histogramas da saída dos modelos temporais CNN citados acima com 10 e 20 neurônios na camada *fully connected*. As distribuições estão bem concentradas nos extremos. A primeira e segunda linhas das colunas

“Classe A” e “Classe B” mostram alguns valores na parte central dos histogramas, isso reflete uma confusão entre as classes A e B. Os valores do modelo (8,8,20,3) apresentam resultados levemente superior ao modelo (8,8,10,3).

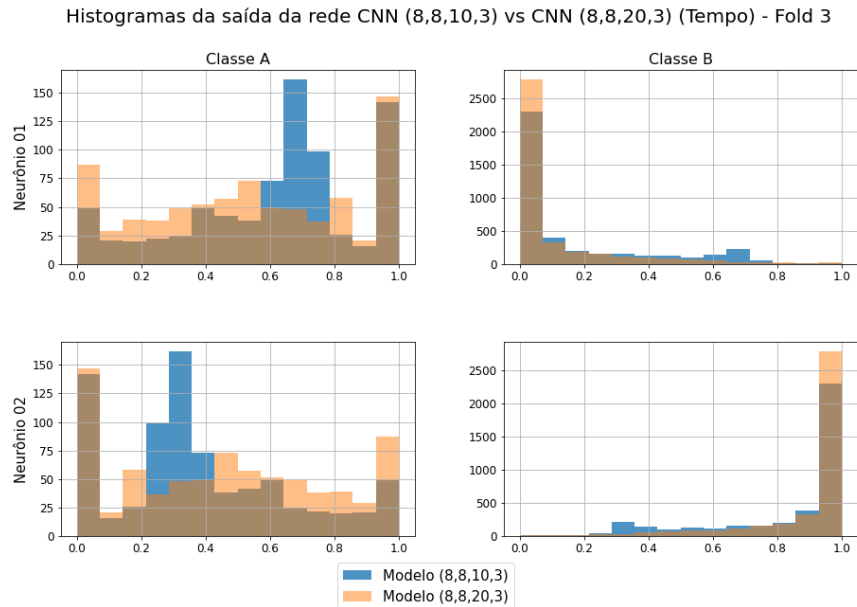


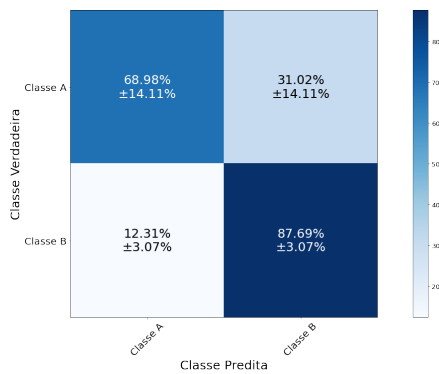
Figura 5.35: Histogramas da saída dos modelos temporais CNN (8,8,10,3) e (8,8,20,3), *fold* 3 (águas rasas).

A Figura 5.36 apresenta a matriz de confusão dos conjuntos de treinamento e validação dos modelos temporais convolucionais com 10 e 20 neurônios da camada *fully connected*. O desempenho foi inferior ao apresentado no modelo LOFAR. A classe A foi a classe de menor desempenho, além de ter uma grande margem de erro. As curvas de treinamento dos modelos encontram-se no Apêndice C.2.2.

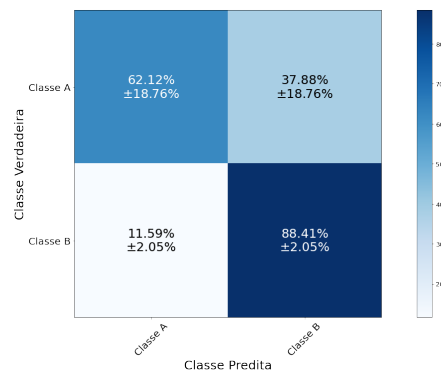
A Tabela 5.25 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com 10 e 20 neurônios na camada *fully connected*. Assim como apresentado no modelo LOFAR, o modelo temporal não apresentou melhoras significativas ao utilizar 20 neurônios na camada *fully connected*. Sendo assim, o modelo com 10 neurônios é o modelo escolhido para ser utilizado na rede de fusão.

Tabela 5.25: Resultados experimentais da análise Temporal (1024) dos modelos CNN para o conjunto de validação (águas rasas)

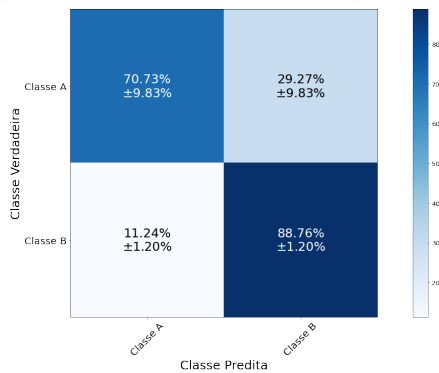
Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
(8, 8, 10, 3)	73, 86 ± 11, 55	84, 12 ± 2, 47	70, 99 ± 6, 30	75, 26 ± 8, 81
(8, 8, 20, 3)	73, 49 ± 11, 26	84, 61 ± 4, 31	72, 08 ± 7, 99	74, 77 ± 9, 18



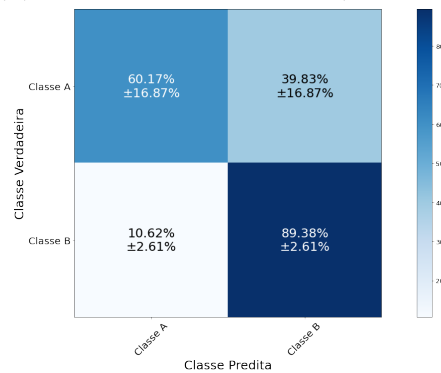
(a) Conjunto de treinamento (8,8,10,3).



(b) Conjunto de validação (8,8,10,3).



(c) Conjunto de treinamento (8,8,20,3).



(d) Conjunto de validação (8,8,20,3).

Figura 5.36: Matriz de Confusão dos modelos temporais CNN com 10 e 20 neurônios na camada *fully connected* (águas rasas).

Resultados para a Rede Neural LSTM

A Figura 5.37 mostra a curva do valor médio com margem de erro do índice SP aplicado sobre o conjunto de validação pela variação de células da camada oculta do modelo temporal LSTM. Analisando o comportamento das curvas, o valor médio do índice SP melhora com o aumento do número de células, com exceção ao do modelo com 75 células.

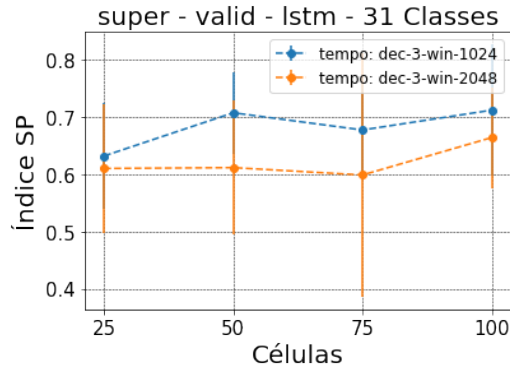


Figura 5.37: Gráfico do índice SP pela variação do número de células LSTM (águas rasas).

A Tabela 5.26 apresenta os valores médios e os erros do índice SP de acordo com o número de células do modelo. Comparando as duas janelas, 1024 e 2048 pontos, é possível observar que os valores médios das janelas de 1024 pontos apresentam resultados melhores, mas considerando a margem de erro, os valores são bem próximos. Assim como observado nos modelos MLP e CNN.

Tabela 5.26: Resultados experimentais do índice SP para o modelo LSTM do conjunto de validação (águas rasas)

Células	Índice SP (%)	
	tempo-1024	tempo-2048
25	63,19 ± 9,17	61,01 ± 11,26
50	70,75 ± 7,15	61,16 ± 11,74
75	67,74 ± 8,13	59,87 ± 21,29
100	71,21 ± 11,47	66,41 ± 8,79

Como mencionado anteriormente nesta seção, o número de parâmetros é um fator que deve ser considerado na escolha do modelo. Lembrando que o número total de amostras de treinamento do *dataset* para as janelas de 1024 pontos é 38.448. A Tabela 5.27 mostra a variação do número de células LSTM pela quantidade de parâmetros. Até 75 células, a quantidade de parâmetros não ultrapassam a quantidade de amostras do *dataset*. Sendo assim, qualquer modelo temporal até 75 células pode ser escolhido.

Tabela 5.27: Quantidade de parâmetros para cada modelo LSTM

Células	Parâmetros
	tempo-1024 & 2048
25	2.804
50	10.604
75	23.404
100	41.204

A Figura 5.38 mostra os gráficos de violino do índice SP sobre os dados do conjunto de validação pela variação de células da camada oculta do modelo para as janelas de 1024 pontos. Com base nesses resultados, observamos que a partir de 50 células, o índice SP não teve uma melhora significativa. Portanto, a escolha desse parâmetro para a camada oculta LSTM é a mais apropriada. O índice SP é $(70, 75 \pm 7, 15)\%$.

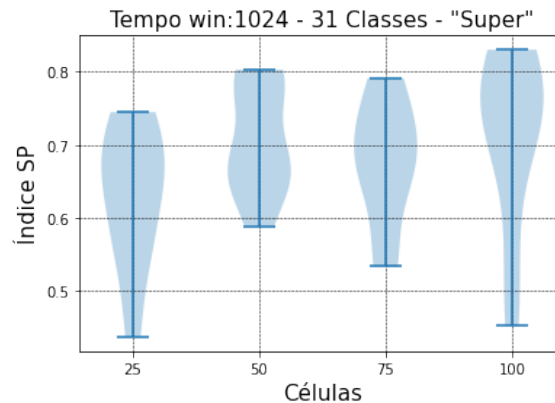


Figura 5.38: Gráfico de Violino do desempenho dos modelos temporais LSTM no conjunto de validação (águas rasas).

A Figura 5.39 mostra uma matriz de histogramas da saída do modelo LSTM com 50 células na camada oculta. Os eventos não estão concentrados nos extremos, mas a maior quantidade dos eventos estão concentrados entre 0,6 e 0,8 quando as classes são Verdadeiras Positivas. Os Verdadeiros Negativos estão concentrados entre 0,2 e 0,4.

A Figura 5.40 apresenta a matriz de confusão dos conjuntos de treinamento e validação do modelo temporal com 50 células LSTM. Diferente dos resultados apresentados no conjunto de dados de águas profundas, o conjunto de dados de águas rasas não obteve o melhor resultado com o modelo temporal LSTM. A rede superou os resultados da rede MLP, porém, ficou abaixo do modelo CNN. A incerteza com relação a acurácia da classe A, reflete na margem de erro do índice SP. As curvas de treinamento do modelo encontram-se no Apêndice C.2.3.

Os resultados de classificação deste modelo são: Acurácia de $(82, 55 \pm 2, 67)\%$,

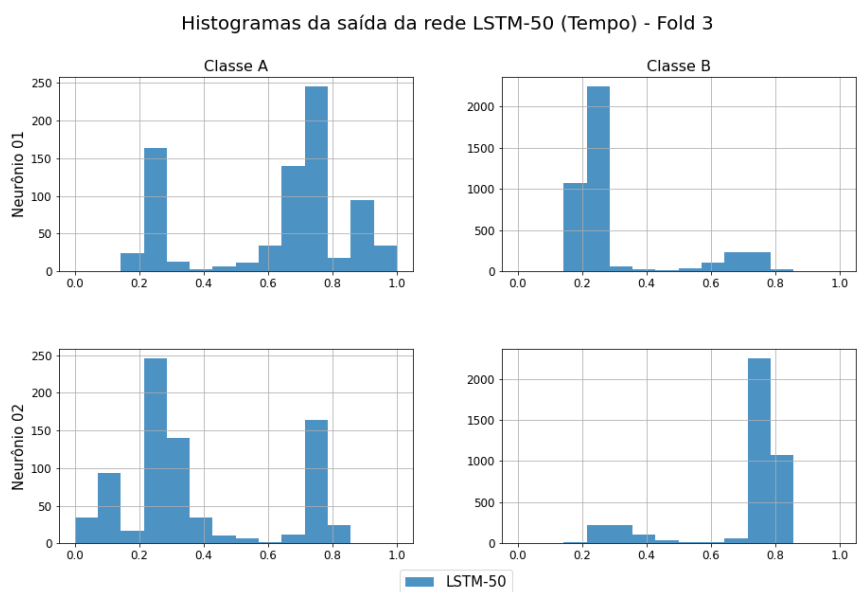
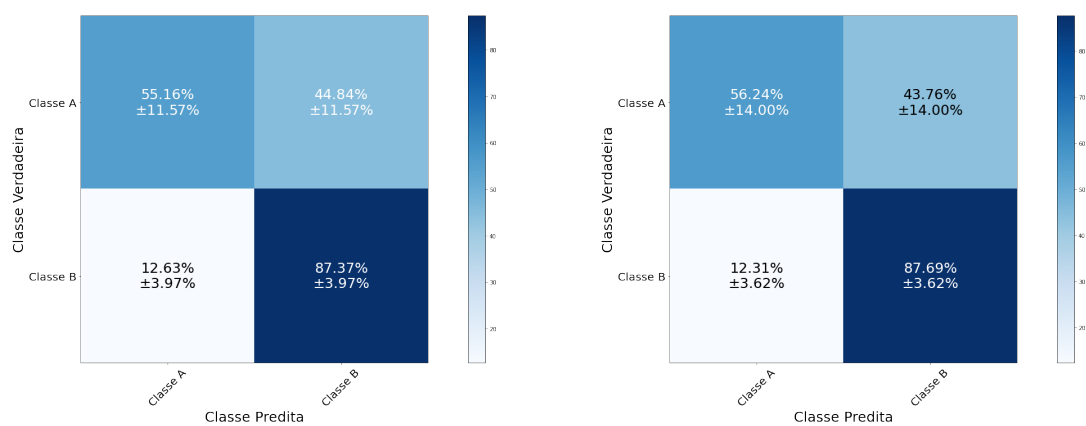


Figura 5.39: Histogramas da saída do modelo temporal LSTM-50, *fold* 3 (águas rasas).



(a) Conjunto de treinamento LSTM-50.

(b) Conjunto de validação LSTM-50.

Figura 5.40: Matriz de Confusão do modelo temporal LSTM com 50 células (águas rasas).

Precisão de $(69,47 \pm 4,16)\%$ e *Recall* de $(71,96 \pm 6,24)\%$.

Avaliação dos Modelos

Após o desenvolvimento dos modelos LOFAR e temporal MLP, CNN e LSTM é importante analisar os resultados obtidos. Os modelos escolhidos foram:

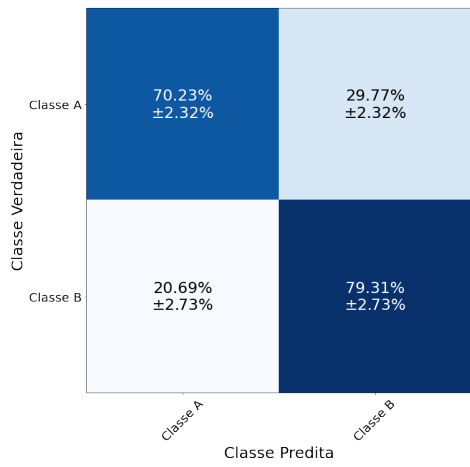
- MLP: Os modelos LOFAR e temporal possuem 10 neurônios na camada oculta.
- CNN: Os modelos LOFAR e temporal possuem duas camadas convolucionais de 8 filtros, uma camada *fully connected* de 10 neurônios e um *kernel* 3.
- LSTM: O modelo temporal possui 50 células na camada oculta.

A Figura 5.41 apresenta a matriz de confusão do conjunto de teste dos modelos LOFAR e temporal das redes mencionadas acima. Esses resultados em conjunto com as informações de Acurácia, Precisão e *Recall* do conjunto de teste apresentados na Tabela 5.28, além de todas as informações discutidas nas escolhas dos modelos, nos permite dizer que:

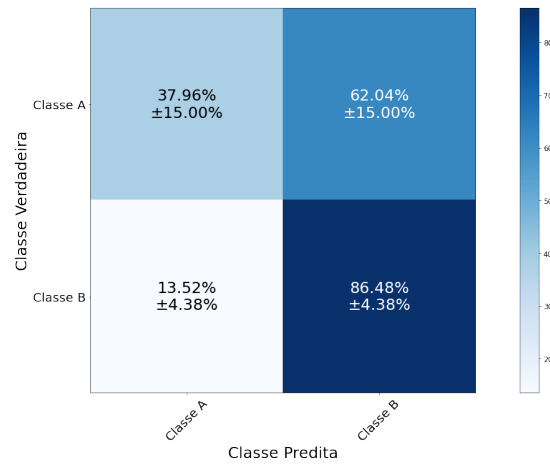
- Comparando o modelo temporal MLP e CNN, nota-se um aumento muito grande do valor do índice SP com o uso da rede CNN, entretanto, a margem de erro dos valores é muito grande, o que ocasiona a sobreposição entre o valor mínimo apresentado pelo rede CNN com o valor máximo da rede MLP. O modelo temporal LSTM apresentou valores semelhantes aos apresentados pela rede CNN.
- Assim como ocorreu no modelo temporal, o modelo LOFAR CNN aumentou o valor médio do índice SP em comparação com a rede MLP, mas considerando a margem de erro, o valor mínimo do SP com a rede CNN sobrepõe o valor máximo do SP da rede MLP. Portanto, a melhora é pouco significativa.
- O modelo temporal usando a rede LSTM obteve resultados superiores aos apresentados com a rede MLP e semelhantes aos apresentados na rede CNN.

Tabela 5.28: Resultados experimentais do conjunto de teste da fusão dos modelos utilizando a rede MLP (águas rasas)

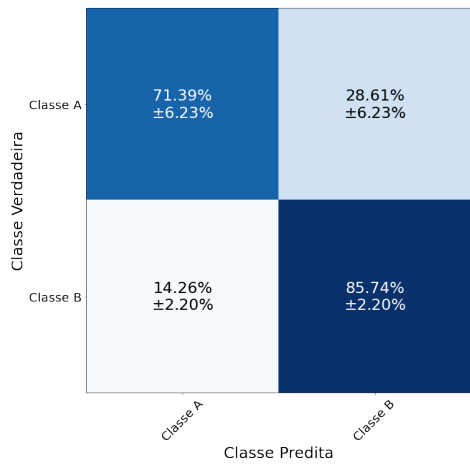
Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
<i>MLP(LOFAR)</i>	$74,69 \pm 1,64$	$77,83 \pm 2,25$	$66,61 \pm 1,64$	$74,77 \pm 1,64$
<i>CNN(LOFAR)</i>	$78,34 \pm 3,10$	$83,39 \pm 1,76$	$71,74 \pm 2,02$	$78,56 \pm 2,91$
<i>MLP(Tempo)</i>	$58,91 \pm 7,90$	$78,56 \pm 1,84$	$61,49 \pm 3,18$	$62,22 \pm 5,63$
<i>CNN(Tempo)</i>	$73,36 \pm 10,86$	$83,74 \pm 3,91$	$71,10 \pm 5,85$	$74,65 \pm 8,90$
<i>LSTM(Tempo)</i>	$69,55 \pm 9,83$	$82,24 \pm 2,45$	$68,34 \pm 5,63$	$71,18 \pm 7,69$



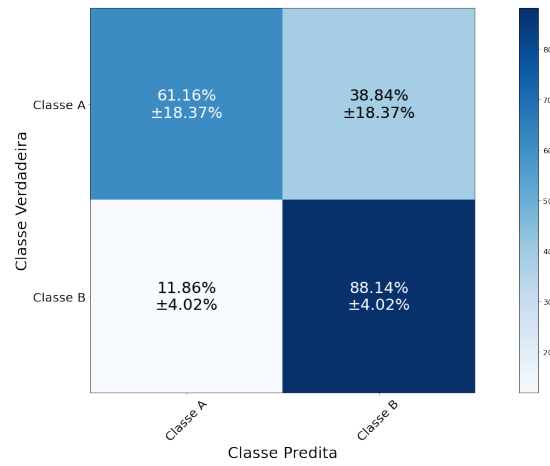
(a) Modelo MLP (LOFAR).



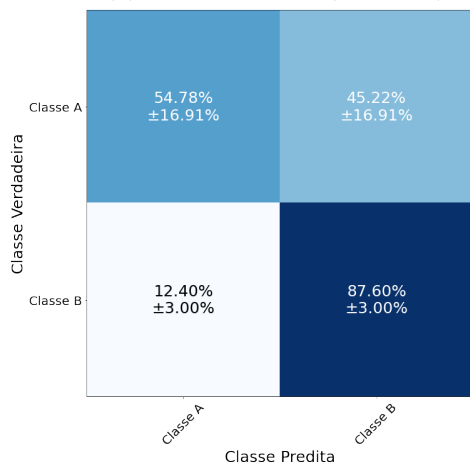
(b) Modelo MLP (Temporal).



(c) Modelo CNN (LOFAR).



(d) Modelo CNN (Temporal).



(e) Modelo LSTM (Temporal).

Figura 5.41: Matriz de Confusão do conjunto de teste do modelo MLP, CNN e LSTM (águas rasas).

Modelo de Fusão Proposto

Assim como no conjunto de águas profundas, nesta seção iremos analisar o desenvolvimento e os resultados da fusão dos modelos temporal e LOFAR das redes MLP, CNN e LSTM. As fusões implementadas foram: MLP (LOFAR) com MLP (tempo), CNN (LOFAR) com CNN (tempo), MLP (LOFAR) com LSTM (tempo) e CNN (LOFAR) com LSTM (tempo). Os dados apresentados nesta seção para escolha das redes de fusão podem ser visto com mais detalhes no Apêndice A. O método de escolha é similar ao apresentado na Subseção 5.3.1. A Tabela 5.29 mostra o resultado do índice SP para o conjunto de validação de cada uma das fusões de acordo com a variação de neurônios descrita e a quantidade de parâmetros treináveis/não-treináveis.

Tabela 5.29: Resultados experimentais do índice SP e dos parâmetros da fusão dos modelos nos conjuntos de validação (águas rasas)

Neurônios da Rede de fusão	MLP com MLP	CNN com CNN	MLP com LSTM	CNN com LSTM
	SP(%)			
–	71,74 ± 2,28	80,92 ± 5,38	76,35 ± 3,42	80,76 ± 5,16
10	77,96 ± 3,65	81,33 ± 6,06	80,17 ± 3,82	80,66 ± 5,75
30	77,97 ± 3,92	80,89 ± 5,19	79,83 ± 3,78	81,52 ± 4,87
50	78,08 ± 3,79	80,94 ± 5,71	80,14 ± 3,75	80,57 ± 4,90
60	77,77 ± 3,34			
100		80,64 ± 5,13	80,02 ± 3,94	80,41 ± 4,48
Parâmetros Treináveis - Não Treináveis				
–	42 – 14.260	7.298 – 464	122 – 14.410	3.302 – 10.632
10	232 – 14.260	36.512 – 464	632 – 14.410	16.532 – 10.632
30	692 – 14.260	109.532 – 464	1.892 – 14.410	49.592 – 10.632
50	1.152 – 14.260	182.552 – 464	3.152 – 14.410	82.652 – 10.632
60	1.382 – 14.260			
100		365.102 – 464	6.302 – 14.410	165.302 – 10.632

Na rede de fusão com os modelos LOFAR e temporal utilizando as redes MLP como entrada, é possível observar uma melhora com a adição da rede com 10 neurônios na camada oculta, mas com o aumento do número de neurônios não foi possível constatar um aumento no valor médio do índice SP e nem uma melhora na margem de erro. Os resultados do índice SP, acurácia, precisão e *recall* dos modelos estão descritos na Tabela 5.30. Nas outras redes de fusão é possível notar que a fusão com a rede não melhorou o desempenho da fusão em relação aos dados apresentados sem a rede. Os resultados de classificação reafirmam a análise do índice SP. Baseado nos resultados apresentados, as redes de fusão utilizadas para avaliar o conjunto de teste são as redes com 10 neurônios na camada oculta em todas as composições. A rede de fusão foi utilizada nos casos em que a rede não melhorou a classificação, para que seja possível analisar o método proposto.

A Figura 5.42 apresenta a matriz de confusão do conjunto de validação da fusão dos modelos. Os modelos híbridos (exemplo: MLP com LSTM) não melhoraram o desempenho em relação aos modelos não híbridos (exemplo: MLP com MLP), houve uma melhora no valor médio da acurácia, mas considerando a margem de erro, o

Tabela 5.30: Resultados experimentais do conjunto de validação da fusão dos modelos (águas rasas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
MLP com MLP				
10	77,96 ± 3,65	81,26 ± 1,51	69,74 ± 2,16	78,08 ± 3,48
CNN com CNN				
–	80,92 ± 5,38	84,15 ± 2,14	72,87 ± 3,26	81,07 ± 5,10
10	81,33 ± 6,06	85,31 ± 2,92	74,18 ± 4,18	81,51 ± 5,78
MLP com LSTM				
–	76,35 ± 3,42	76,61 ± 1,79	66,64 ± 1,87	76,41 ± 3,35
10	80,17 ± 3,82	81,75 ± 1,84	70,78 ± 2,36	80,24 ± 3,70
CNN com LSTM				
–	80,76 ± 5,16	83,56 ± 2,41	72,36 ± 3,35	80,88 ± 4,93
10	80,66 ± 5,75	83,88 ± 2,49	72,60 ± 3,69	80,82 ± 5,43

valores são semelhantes.

A Figura 5.43 mostra uma matriz de histogramas da saída da fusão dos modelos MLP com MLP e MLP com LSTM. As distribuições são bem parecidas. O modelo híbrido apresentou uma frequência maior de valores próximos de “1” para os Verdadeiros Positivos e próximos de “0” para os Verdadeiros Negativos, mas essa pequena diferença não reflete em uma melhora na classificação.

A Figura 5.44 mostra uma matriz de histogramas da saída dos modelos CNN com CNN e CNN com LSTM. As distribuições estão concentradas nos extremos, mas não há uma diferença significativa entre os modelos.

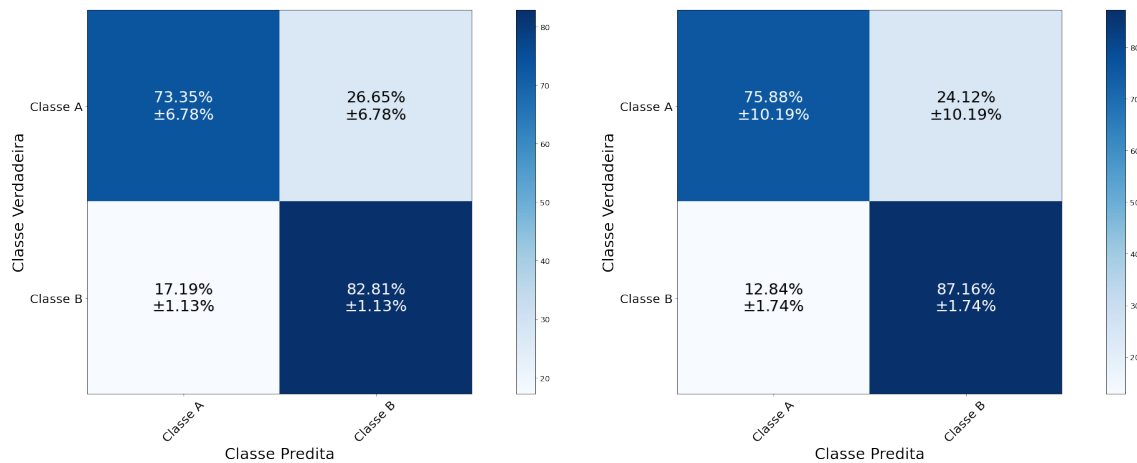
A Figura 5.45 apresenta a matriz de confusão do conjunto de teste da fusão dos modelos. O maior valor de acurácia da classe A é obtido com a fusão MLP com LSTM, (77,32 ± 6,40)%. A acurácia da classe B em todos os modelos obteve um valor acima de 80%.

A Tabela 5.31 mostra os resultados do conjunto de teste do índice SP, acurácia, precisão e *recall* da fusão dos modelos. Todos os modelos apresentaram resultados semelhantes em torno de 80%, apesar de valores médios e margem de erros distintos, não há diferença entre os modelos.

Tabela 5.31: Resultados experimentais do conjunto de teste da fusão dos modelos (águas profundas)

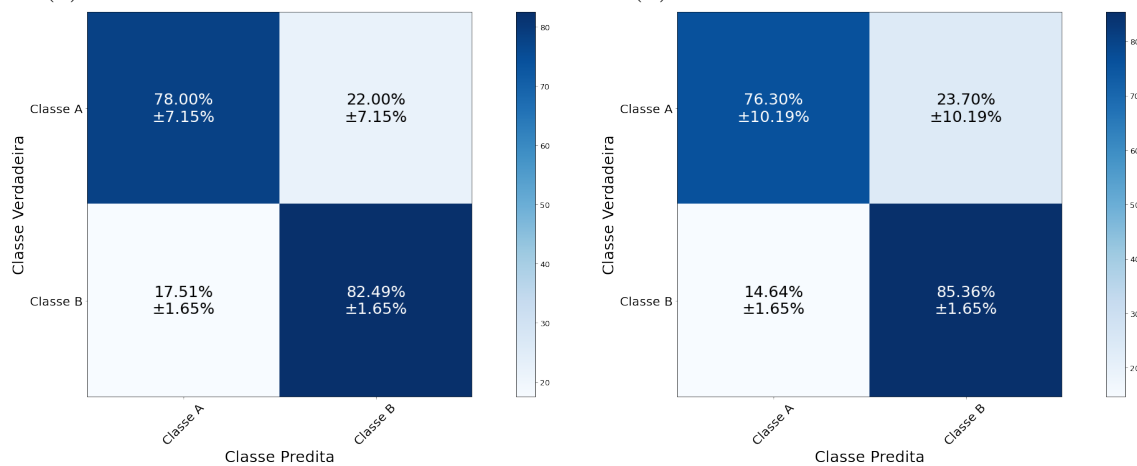
Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
<i>MLP – MLP</i>	78,11 ± 2,44	81,14 ± 1,36	69,73 ± 1,56	78,19 ± 2,36
<i>CNN – CNN</i>	81,56 ± 5,01	85,17 ± 2,21	74,04 ± 3,28	81,70 ± 4,82
<i>MLP – LSTM</i>	79,75 ± 3,53	81,49 ± 2,11	70,51 ± 2,43	79,81 ± 3,45
<i>CNN – LSTM</i>	80,36 ± 4,84	83,87 ± 2,24	72,54 ± 3,16	80,50 ± 4,64

A Tabela 5.32 mostra o resultado do índice SP do conjunto de validação e teste dos modelos LOFAR e temporal escolhidos, além de suas fusões. Considerando esses



(a) Fusão do modelo MLP com MLP.

(b) Fusão do modelo CNN com CNN.



(c) Fusão do modelo MLP com LSTM.

(d) Fusão do modelo CNN com LSTM.

Figura 5.42: Matriz de Confusão do conjunto de validação da fusão dos modelos (águas rasas).

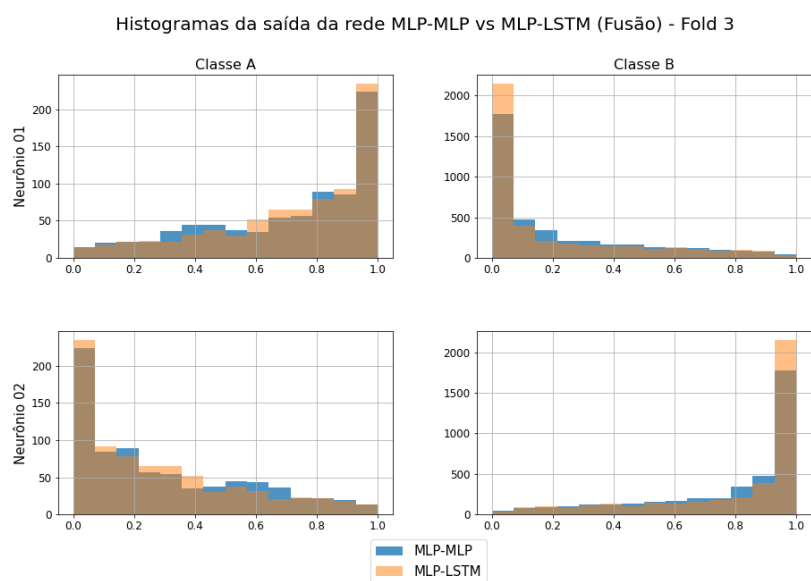


Figura 5.43: Histogramas da saída da fusão dos modelos MLP-MLP e MLP-LSTM (águas rasas).

Histogramas da saída da rede CNN-CNN vs CNN-LSTM (Fusão) - Fold 3

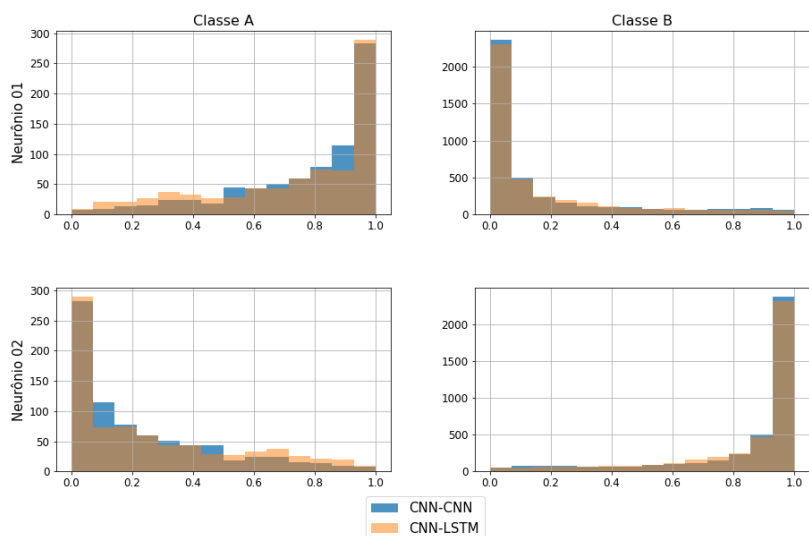
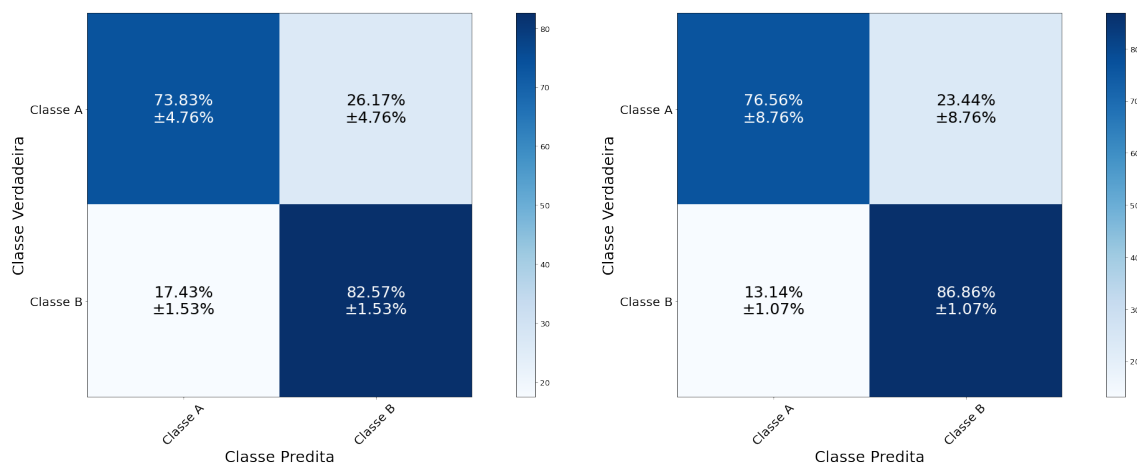
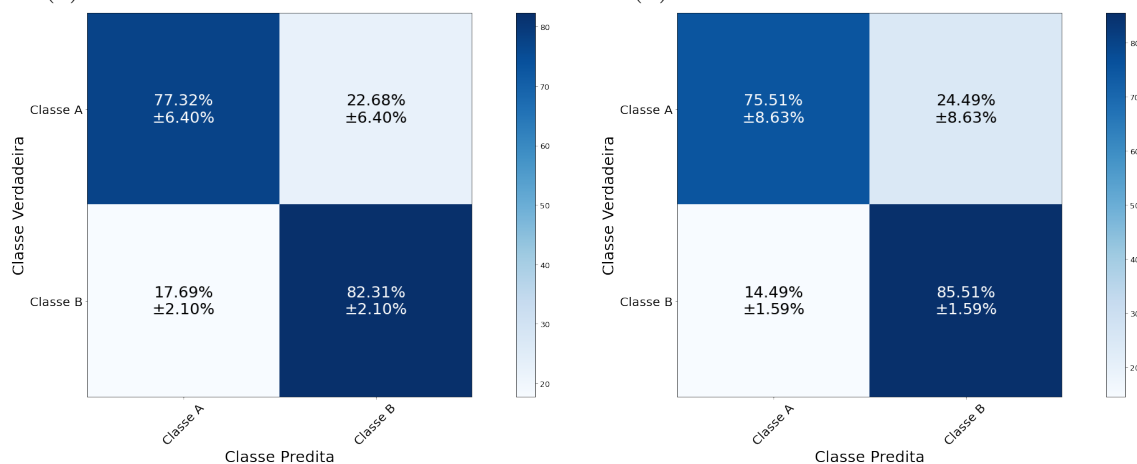


Figura 5.44: Histogramas da saída da fusão dos modelos CNN-CNN e CNN-LSTM (águas rasas).



(a) Fusão do modelo MLP com MLP.

(b) Fusão do modelo CNN com CNN.



(c) Fusão do modelo MLP com LSTM.

(d) Fusão do modelo CNN com LSTM.

Figura 5.45: Matriz de Confusão do conjunto de teste da fusão dos modelos (águas rasas).

resultados e os outros apresentados anteriormente podemos dizer que:

- O uso da mesma tecnologia como entrada da rede de fusão (exemplo: MLP com MLP) não apresenta vantagens em relação ao uso individual dos modelos. A fusão MLP com MLP apresentou valores dentro da margem de erro do modelo LOFAR MLP e a fusão CNN com CNN apresentou valores dentro da margem de erro do modelo LOFAR CNN.
- O uso de tecnologias híbridas como entrada da rede de fusão (exemplo: MLP com LSTM) apresenta ganhos quando a entrada LOFAR é composta pela rede MLP, mas quando usamos a rede CNN, os resultados são semelhantes aos apresentados no modelo LOFAR.
- O uso da rede de fusão com as entradas MLP com LSTM apresentou ganhos em relação aos modelos alimentados por dados processados por apenas uma análise individual.

Tabela 5.32: Resultados experimentais do índice SP dos modelos MLP, CNN , LSTM e da fusão nos conjuntos de validação e teste (águas rasas)

Modelos	Índice SP (%)					
	Validação			Teste		
	LOFAR	Temporal	Fusão	LOFAR	Temporal	Fusão
MLP	74,60 ± 2,54	59,06 ± 7,51		74,69 ± 1,64	58,91 ± 7,90	
CNN	77,74 ± 3,84	73,86 ± 11,55		78,34 ± 3,10	73,36 ± 10,86	
LSTM		70,75 ± 7,15			69,55 ± 9,83	
MLP - MLP			77,96 ± 3,65			78,11 ± 2,44
CNN - CNN			81,33 ± 6,06			81,56 ± 5,01
MLP - LSTM			80,17 ± 3,82			79,75 ± 3,53
CNN - LSTM			80,66 ± 5,75			80,36 ± 4,84

Capítulo 6

Conclusões e Trabalhos Futuros

Neste Seção iremos apresentar a conclusão dos resultados discutidos ao longo da Seção 5.

6.1 Conclusões

Este trabalho propôs o desenvolvimento de classificadores hierárquicos não-especialistas de navios com o uso de sinais de sistema de sonar passivo. Os classificadores utilizam técnicas de modelos profundos (redes convolucionais e recorrentes) com dados processados pelas análises LOFAR e Temporal. Além disso, foi utilizado um modelo de classificação capaz de fundir as informações temporais e espectrais dos sinais. Para isso, dois bancos de dados foram utilizados, o primeiro contendo 24 classes, chamado de águas profundas e o segundo contendo 31 classes, chamado de águas rasas ou VIPP. Os dados de cada conjunto foram organizados hierarquicamente em vários níveis e o método proposto foi aplicado apenas no nível “Super” da hierarquia.

Todos os classificadores desenvolvidos tiveram mais dificuldade em classificar o conjunto de dados de águas rasas do que o conjunto de águas profundas, como visto ao longo das discussões e sintetizadas nas Tabelas 5.18 e 5.32. O baixo desempenho está associado a dificuldade de classificação da classe A do conjunto de dados de águas rasas. Um dos motivos é que os dados do conjunto de águas rasas (VIPP) são coletados de um sistema instalado em um canal de navegação muito estreito, impossibilitando a gravação dos ruídos por um longo período de tempo, acarretando uma quantidade menor de amostras, como observado na Tabela 5.2.

Analisando o conjunto de águas profundas nota-se que o modelo convolucional não apresentou uma melhora significativa nos dados processados pela análise LOFAR em relação ao modelo MLP. Como observado na Tabela 5.18, apesar da melhora do valor médio do índice SP os resultados estão dentro da mesma incerteza. Ressalta-se que os modelos convolucionais são dotados de uma grande quantidade de parâmetros

e maior complexidade, aumentando o tempo de processamento. Uma vez operando juntamente com o OS em um submarino, o sistema de classificação automática tem que ser o mais ágil possível. Sendo assim, a complexidade do modelo deve ser levada em consideração, dado que a melhora na classificação não foi tão expressiva. Os modelos alimentados pela análise temporal mostraram resultados inferiores aos modelos alimentados pela análise LOFAR nas redes MLP e CNN. Apesar disso, o uso do modelo temporal LSTM apresentou um resultado superior aos apresentados pelo modelo LOFAR (em todas as redes estudadas). Esse resultado deve ser olhado com devido cuidado já que no conjunto de águas rasas, os resultados do modelo LSTM ficaram próximos aos obtidos com a rede convolucional alimentada por dados espectrais. A fusão das informações espectrais e temporais para o conjunto de águas profundas não trouxe benefícios na classificação, apesar dos resultados ficarem acima dos obtidos nos modelos MLP e CNN alimentados pelos dados da análise LOFAR. O modelo Temporal LSTM apresentou resultados semelhantes, o que impossibilita afirmar que a rede de fusão melhorou a classificação. Salienta-se o fato que o uso da mesma tecnologia como entrada da rede de fusão (exemplo MLP com MLP) não apresenta vantagens em relação ao uso de tecnologias híbridas como entrada da rede de fusão (exemplo: MLP com LSTM).

Analisando o conjunto de águas rasas nota-se que o modelo convolucional apresentou uma melhora pouco significativa nos dados processados pela análise LOFAR em relação ao modelo MLP, como observado na Tabela 5.32. Os modelos MLP e CNN alimentados pela análise Temporal tiveram resultados abaixo dos obtidos pela análise LOFAR, com exceção do modelo temporal LSTM, que obteve resultados equivalentes ao do modelo LOFAR CNN. O bom resultado da rede LSTM no tempo era previsto, dado que elas são adequadas para dados sequenciais. A rede de fusão com as entradas MLP e LSTM apresentou ganhos em relação aos modelos alimentados por dados processados por apenas uma análise. Neste conjunto de dados as fusões abordando tecnologia híbrida ou não híbridas obtiveram o mesmo desempenho.

O trabalho desenvolvido por GOLTZ [15] apresenta o valor médio do SP para o comitê hierárquico de $73,00\% \pm 1,00\%$ no conjunto de águas profundas e $61,00\% \pm 2,00\%$ no conjunto de águas rasas. Neste trabalho os valores obtidos com a rede de fusão (MLP com LSTM) apresentaram no nível hierárquico “Super”, o índice SP de $88,76\% \pm 2,81\%$ e $79,75\% \pm 3,53\%$ para os dados de águas profundas e rasas respectivamente no conjunto de teste.

Em suma, o método proposto apresentou bons resultados em comparação aos observados isoladamente com a análise LOFAR e a Temporal. Os resultados apresentam margem de melhora dada a capacidade do modelo proposto em incluir outras análises no processo de classificação. A rede LSTM com os dados da análise Tem-

poral apresentou bons resultados, mas deve ser melhor avaliada no futuro, dado a diferença de resultados obtidos nos dois *datasets*. A rede convolucional não apresentou vantagens em relação a rede MLP.

6.2 Trabalhos Futuros

A análise espectral utilizada neste trabalho foi desenvolvida por SOARES FILHO [21] com foco em sinais coletados em raia acústica com o mínimo de interferência possível. Entretanto, neste trabalho utilizamos dados adquiridos por submarinos em operação real e sinais de águas rasas sujeitos a inúmeras interferências devido a sua propagação em multi-percursos. A análise temporal foi baseada nesta análise e ainda têm a desvantagem de não possuir um sistema de filtragem de ruído. Os resultados obtidos neste trabalho podem melhorar com uma análise mais detalhada e modificação das técnicas de processamento para as condições operativas dos conjuntos de dados utilizados.

Apesar deste trabalho se pautar no conceito de comitê hierárquico, apenas o nível “Super” foi estudado. Como trabalho futuro sugere-se a implementação do método utilizado neste trabalho para os outros níveis da hierarquia, possibilitando a comparação dos resultados do comitê hierárquico com uma estrutura não hierárquica.

Neste trabalho foram utilizadas duas técnicas de processamento dos dados, mas o método proposto apresenta a possibilidade de adicionar inúmeras técnicas. Sugere-se a adição de outras técnicas, tal como a análise DEMON citada na Seção 4.2, a fim de melhorar os índices encontrados.

A rede LSTM com os dados da análise Temporal apresentou bons resultados em águas profundas, sendo assim, pode-se aproveitar a informação temporal para melhorar a eficiência de classificação. Outrossim, torna-se interessante avaliar o desempenho da rede LSTM com dados processados pela análise LOFAR. Essa abordagem pode ajudar a entender melhor o processo que levou a rede a obter bons desempenhos.

Referências Bibliográficas

- [1] MARINHA DO BRASIL. “Missão e Visão de Futuro”. Disponível em: <https://www.marinha.mil.br/content/missao-e-visao-de-futuro-da-marinha>. Acesso em: 28 novembro 2021, .
- [2] MARINHA DO BRASIL, B. “O que é a Amazônia Azul e por que o Brasil quer se tornar potência militar no Atlântico”. Disponível em: <https://www.marinha.mil.br/economia-azul/noticias/o-que-%C3%A9-amaz%C3%B4nia-azul-e-por-que-o-brasil-quer-se-tornar-pot%C3%A2ncia-militar-no-atl%C3%A2ntico>. Acesso em: 28 março 2022, Novembro 2019.
- [3] COMISSÃO INTERMINISTERIAL PARA OS RECURSOS DO MAR, B. “AMAZÔNIA AZUL”. Disponível em: <https://www.marinha.mil.br/secirm/amazoniaazul>. Acesso em: 30 novembro 2021.
- [4] MARINHA, B. “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/>. Acesso em: 15 novembro 2021.
- [5] GRANDE, U. F. D. R. “Ciências do Mar Brasil”. Disponível em: <https://cienciasdomarbrasil.furg.br/>. Acesso em: 11 abril 2022, 2019.
- [6] MARINHA DO BRASIL, B. “Programa de Submarinos”. Disponível em: <https://www.marinha.mil.br/prosub/institucional>. Acesso em: 28 março 2022, .
- [7] MINISTÉRIO DA DEFESA NACIONAL, B. *Estratégia nacional de defesa*. 2008.
- [8] DE MOURA JUNIOR, N. N. *Detecção de Novidade para Sistemas de Sonar Passivo*. Tese de Doutorado, D. Sc. dissertation, Universidade Federal do Rio de Janeiro, 2018.

- [9] URICK, R. J. “Principles of underwater sound-2”, 1975.
- [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.
- [11] BLOUIN, S. “Simultaneous design of underwater acoustic sensor and communication networks”. In: *Proceedings of the International Conference on Underwater Networks & Systems*, pp. 1–5, 2017.
- [12] NIELSEN, R. O. “Sonar Signal Processing”. Norwood, MA, USA: Artech House. 1991.
- [13] LI, Q. *Digital sonar design in underwater acoustics: principles and applications*. Springer Science & Business Media, 2012.
- [14] MOHRI, M., ROSTAMIZADEH, A., TALWALKAR, A. “Foundations of machine learning.[Sl]”. 2012.
- [15] GOLTZ, G. A. M. *Classificação de navios com uso de sinais de sonar passivo*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2019.
- [16] DE PESQUISAS DA MARINHA, I. “Sistema de Vigilância & Informações Passivas em Portos”. Disponível em: <https://www.marinha.mil.br/ipqm>. Acesso em: 11 abril 2022.
- [17] DE MOURA, N. N. *Detecção e classificação de sinais de sonar passivo usando métodos de separação cega de fontes*. Tese de Doutorado, COPPE/UFRJ, 2013.
- [18] DE CASTRO VARGAS FERNANDES, J., DE MOURA JUNIOR, N. N., DE SEIXAS, J. M. “Análise de Amostras Sintéticas de Sinais de Sonar Passivo Geradas por Redes Neurais Generativas Adversariais”. In: *Anais do 14 Congresso Brasileiro de Inteligência Computacional*, pp. 1–8, Curitiba, PR, 2019.
- [19] MARLON SOUZA, N. J., SEIXAS., J. “Classificação de Sinais Temporais de Sonar Passivo utilizando Redes Neurais Convolucionais”. In: Filho, C. J. A. B., Siqueira, H. V., Ferreira, D. D., et al. (Eds.), *Anais do 15 Congresso Brasileiro de Inteligência Computacional*, pp. 1–7, Joinville, SC, 2021. SBIC.
- [20] HAYKIN, S., NETWORK, N. “A comprehensive foundation”, *Neural networks*, v. 2, n. 2004, pp. 41, 2004.

- [21] SOARES FILHO, W. *Classificação do Ruído Irrradiado por Navios Usando Redes Neurais*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2001.
- [22] GOODFELLOW, I., BENGIO, Y., COURVILLE, A. *Deep learning*. MIT press, 2016.
- [23] BARINO, F. O., DOS SANTOS, A. B. “Rede Neural Convolutacional 1D aplicada à previsão da vazão no Rio Madeira”, *XXXVIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*, 2020.
- [24] ZHANG, Q., DA, L., ZHANG, Y., et al. “Integrated neural networks based on feature fusion for underwater target recognition”, *Applied Acoustics*, v. 182, pp. 108261, 2021.
- [25] SOUZA FILHO, J. B., DE SEIXAS, J. M. “Class-modular multi-layer perceptron networks for supporting passive sonar signal classification”, *IET Radar, Sonar & Navigation*, v. 10, n. 2, pp. 311–317, 2016.
- [26] CERQUEIRA, A., DE SEIXAS, J. M. “Classificação de sinais de Sonar Passivo utilizando Stacked AutoEncoders”, 2017.
- [27] DA SILVA MUNIZ, V., SOUZA FILHO, J., OLIVEIRA, M. “Classificação Hierárquica de Navios em Sinais de Sonar Passivo”. pp. 1–5, 01 2020. doi: 10.21528/CBIC2019-34.
- [28] VILLANUEVA, W. J. P. “Comitê de máquinas em predição de séries temporais”. In: *Mestrado em Engenharia Elétrica e Computação*, Faculdade de Engenharia Elétrica e de Computação, FEEC/UNICAMP, Campinas, SP, Brasil, 2006.
- [29] KOMARI ALAIE, H., FARSI, H. “Passive sonar target detection using statistical classifier and adaptive threshold”, *Applied Sciences*, v. 8, n. 1, pp. 61, 2018.
- [30] WANG, B., WU, C., ZHU, Y., et al. “Ship Radiated Noise Recognition Technology Based on ML-DS Decision Fusion”, *Computational Intelligence and Neuroscience*, v. 2021, 2021.
- [31] BUBLIC, I., SKELIN, D., VUKADIN, P. “Measurement of vessel underwater noise signature”. In: *2008 50th International Symposium ELMAR*, v. 2, pp. 407–410. IEEE, 2008.

- [32] CHAO, L. “Simulating vessel’s underwater radiated noise in spectrum based on actual measurement data”. In: *2016 IEEE/OES China Ocean Acoustics (COA)*, pp. 1–5. IEEE, 2016.
- [33] SANTOS, F. *Auxílio à Classificação Sonar Passiva por Tons Característicos*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2005.
- [34] JOHNSON, D., DUDGEON, D. “Array Signal Processing: Concepts and Techniques, PTR Prentice-Hall”, *Inc., Upper Saddle River, New Jersey*, 1993.
- [35] 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)*, pp. 1–6. IEEE, 2015.
- [36] DINIZ, P. S., DA SILVA, E. A., NETTO, S. L. *Digital signal processing: system analysis and design*. Cambridge University Press, 2010.
- [37] BRIGHAM, E. O. *The fast Fourier transform and its applications*. Prentice-Hall, Inc., 1988.
- [38] BURDIC, W. S., BARTRAM, J. F. “Underwater acoustic system analysis by William S. Burdic”. 1984.
- [39] PIRKL, R. J., AUGHENBAUGH, J. M. “Bayesian passive sonar tracking with conventional beamformer-level data”. In: *2015 18th International Conference on Information Fusion (Fusion)*, pp. 621–628. IEEE, 2015.
- [40] YOCOM, B. A., LA COUR, B. R., YUDICHAK, T. W. “A Bayesian approach to passive sonar detection and tracking in the presence of interferers”, *IEEE Journal of Oceanic Engineering*, v. 36, n. 3, pp. 386–405, 2011.
- [41] DE SEIXAS, J., DE MOURA, N., OTHERS. “Preprocessing passive sonar signals for neural classification”, *IET Radar, Sonar & Navigation*, v. 5, n. 6, pp. 605–612, 2011.
- [42] RAMOS, R. R. *Redes Neurais para Monitoração de Estacionariedade de Sinais de Sonar Passivo*. Dissertação de M.Sc, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2011.
- [43] STRUZINSKI, W. A., LOWE, E. D. “A performance comparison of four noise background normalization schemes proposed for signal detection systems”, *Acoustical Society of America*, v. 76, n. 6, pp. 1738–1742, jul. 1984.
- [44] DUDA, R. O., HART, P. E., STORK, D. G. *Pattern classification and scene analysis*. 3 ed., Wiley New York, 1973.

- [45] KRAMER, M. “Nonlinear Principal Component Analysis using Autoassociative Neural Networks”, *AICHE*, v. 37, pp. 365–375, fev. 1991.
- [46] CALÔBA, L. P., SEIXAS, J., PEREIRA, F. “Neural discriminating analysis for a second-level trigger system”. In: *Computing in High Energy Physics’ 95: CHEP’95*, London Mathematical Society Lecture Note Series, World Scientific, pp. 870–874, 1996.
- [47] HAYKIN, S. *Neural Networks and Learning Machines*. 3 ed. McMaster University, Ontario Canada, Prentice Hall, 2009.
- [48] ZUREK, E., GAMARRA, M. R., ESCORCIA, J. R., et al. “Acoustic fingerprint recognition using artificial neural networks”, *Polibits*, v. 50, pp. 63–73, 2014.
- [49] SITAMAHALAKSHMI, T., BABU, A. V., JAGADEESH, M., et al. “Performance comparison of radial basis function networks and probabilistic neural networks for Telugu character recognition”, *Global Journal of Computer Science and Technology*, 2011.
- [50] DONALD, F., OTHERS. “Probabilistic neural networks and the polynomial adaline as complementary techniques for classification”, *IEEE Transactions on Neural Networks I (I)(March 1990)*, 1990.
- [51] PIMENTEL MARCO, A. “Clifton Lei and Tarassenko Lionel 2014 A review of novelty detection”, *Signal Processing*, v. 99, pp. 215–249.
- [52] FERNANDEZ, H. L. *CLASSIFICAÇÃO DE NAVIOS BASEADA EM CURVAS PRINCIPAIS*. Dissertação de M.Sc, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2005.
- [53] MURPHY, K. P. *Machine learning: a probabilistic perspective*. MIT press, 2012.
- [54] BABU, T. A., KUMAR, P. R. “Characterization and classification of uterine magnetomyography signals using KNN classifier”. In: *2018 Conference on Signal Processing And Communication Engineering Systems (SPACES)*, pp. 163–166. IEEE, 2018.
- [55] VAHIDPOUR, V., RASTEGARNIA, A., KHALILI, A. “An automated approach to passive sonar classification using binary image features”, *Journal of Marine Science and Application*, v. 14, n. 3, pp. 327–333, 2015.

- [56] FERNANDES, J. D. C. V., DE MOURA JUNIOR, N. N., DE SEIXAS, J. M. “Deep Learning Models for Passive Sonar Signal Classification of Military Data”, *Remote Sensing*, v. 14, n. 11, pp. 2648, 2022.
- [57] GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., et al. “Generative adversarial nets”, *Advances in neural information processing systems*, v. 27, 2014.
- [58] YUE, H., ZHANG, L., WANG, D., et al. “The classification of underwater acoustic targets based on deep learning methods”. In: *2017 2nd International Conference on Control, Automation and Artificial Intelligence (CAAI 2017)*, pp. 526–529. Atlantis Press, 2017.
- [59] SHAMIR, L., ORLOV, N., ECKLEY, D. M., et al. “Wndchrm—an open source utility for biological image analysis”, *Source code for biology and medicine*, v. 3, n. 1, pp. 1–13, 2008.
- [60] CAO, X., ZHANG, X., TOGNERI, R., et al. “Underwater target classification at greater depths using deep neural network with joint multiple-domain feature”, *IET Radar, Sonar & Navigation*, v. 13, n. 3, pp. 484–491, 2019.
- [61] CAO, X., ZHANG, X., YU, Y., et al. “Deep learning-based recognition of underwater target”. In: *2016 IEEE International Conference on Digital Signal Processing (DSP)*, pp. 89–93. IEEE, 2016.
- [62] NG, A., OTHERS. “Sparse autoencoder”, *CS294A Lecture notes*, v. 72, n. 2011, pp. 1–19, 2011.
- [63] BISHOP, C. M., OTHERS. *Neural networks for pattern recognition*. Oxford university press, 1995.
- [64] ZHOU, Z.-H. *Ensemble methods: foundations and algorithms*. CRC press, 2012.
- [65] BREIMAN, L. “Random forests”, *Machine learning*, v. 45, n. 1, pp. 5–32, 2001.
- [66] FRIEDMAN, J., HASTIE, T., TIBSHIRANI, R. “Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)”, *The annals of statistics*, v. 28, n. 2, pp. 337–407, 2000.
- [67] BREIMAN, L. “Bagging predictors”, *Machine learning*, v. 24, n. 2, pp. 123–140, 1996.
- [68] SCHAPIRE, R. E. “A brief introduction to boosting”. In: *Ijcai*, v. 99, pp. 1401–1406. Citeseer, 1999.

- [69] YÜZKAT, M., ILHAN, H. O., AYDIN, N. “Multi-model CNN fusion for sperm morphology analysis”, *Computers in Biology and Medicine*, v. 137, pp. 104790, 2021.
- [70] MELO, R. D. O., OTHERS. *Aplicação de redes neurais convolucionais para detectar vazamentos de gás natural em imagens de cabeça de poço petrolíferos onshore*. Tese (doutorado em engenharia elétrica), Universidade Federal do Amazonas, Manaus (AM), Brasil, 2021.
- [71] MANGAI, U. G., SAMANTA, S., DAS, S., et al. “A survey of decision fusion and feature fusion strategies for pattern classification”, *IETE Technical Review*, v. 27, n. 4, pp. 293–307, 2010.
- [72] SARAIVA, G. A., GONÇALVES, A. D. A. C., PINTO, R. A., et al. “Fusão de Dados de Raio-x com Dados Clínicos para Detectar Covid-19”, *Journal of Health Informatics*, v. 12, 2021.
- [73] SOARES-FILHO, W., SEIXAS, J., CALÔBA, L. P. “Enlarging neural class detection capacity in passive sonar systems”. In: *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No. 02CH37353)*, v. 3, pp. III–III. IEEE, 2002.
- [74] SEIXAS, J., DAMAZIO, D., DINIZ, P. S., et al. “Wavelet transform as a preprocessing method for neural classification of passive sonar signals”. In: *ICECS 2001. 8th IEEE International Conference on Electronics, Circuits and Systems (Cat. No. 01EX483)*, v. 1, pp. 83–86. IEEE, 2001.
- [75] KERAS. “Keras”. Disponível em: <https://keras.io/>. Acesso em: 05 maio 2022.
- [76] SUND, R. “Computer Age Statistical Inference: Algorithms, Evidence, and Data Science Bradley Efron and Trevor Hastie Institute of Mathematical Statistics Monographs Cambridge University Press, 2016, hardcover ISBN: 978-1-107-14989-2”. 2019.
- [77] LI, D., CHEN, D., GOH, J., et al. “Anomaly detection with generative adversarial networks for multivariate time series”, *arXiv preprint arXiv:1809.04758*, 2018.
- [78] IZBICKI, R., DOS SANTOS, T. M. *Aprendizado de máquina: uma abordagem estatística*. 2020. ISBN: 978-65-00-02410-4.

- [79] BARROS, T., SILVA, I., GUEDES, L. “Modelagem e visualização científica de dados educacionais: estudo de caso sobre o desempenho em componentes curriculares”. In: *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, v. 6, p. 654, 2017.
- [80] DA SILVA, I. N., SPATTI, D. H., FLAUZINO, R. A. “Redes neurais artificiais para engenharia e ciências aplicadas-curso prático”, *São Paulo: Artliber*, 2010.
- [81] PEEBLES JR, P. Z. *Probability, random variables, and random signal principles*. McGraw-Hill, 2001.
- [82] GUO, T., DONG, J., LI, H., et al. “Simple convolutional neural network on image classification”. In: *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, pp. 721–724. IEEE, 2017.
- [83] GISELY ALVES. “Entendendo Redes Convolucionais (CNNs)”. Disponível em: <https://medium.com/neuronio-br/entendendo-redes-convolucionais-cnns-d10359f21184>. Acesso em: 10 março 2022, out. 2018.
- [84] AYESHMANTHA PERERA. “What is Padding in Convolutional Neural Network’s(CNN’s) padding”. Disponível em: <https://ayeshmanthaperera.medium.com/what-is-padding-in-cnns-71b21fb0dd7>. Acesso em: 10 março 2022, set. 2018.
- [85] GUILHERME DUARTE. “Redes Neurais | Redes Neurais Recorrentes”. Disponível em: <https://medium.com/turing-talks/turing-talks-26-modelos-de-predi%C3%A7%C3%A3o-redes-neurais-recorrentes-439198e9ecf3>. Acesso em: 03 março 2022, dez. 2019.
- [86] FERNANDO MATSUMOTO. “Redes Neurais | LSTM”. Disponível em: <https://medium.com/turing-talks/turing-talks-27-modelos-de-predi%C3%A7%C3%A3o-lstm-df85d87ad210>. Acesso em: 03 março 2022, dez. 2019.
- [87] DATA SCIENCE ACADEMY. “Deep Learning Book”. Disponível em: <https://www.deeplearningbook.com.br/>. Acesso em: 04 março 2022, 2022.

Apêndice A

Fusão dos Modelos

Este apêndice apresenta os resultados da fusão dos modelos temporal e LOFAR das redes MLP, CNN e LSTM.

A.1 Águas Profundas

A Tabela A.1 apresenta os resultados dos índices SP do conjunto de validação dos modelos LOFAR e temporal escolhidos (vide o Capítulo 5). Esses modelos são entradas de uma rede MLP (rede de fusão).

Tabela A.1: Resultados experimentais do índice SP dos modelos no conjunto de validação

Modelos	Índice SP (%)	
	LOFAR-1024	tempo-1024
MLP	$76,30 \pm 2,77$	$63,35 \pm 0,79$
CNN	$80,86 \pm 2,90$	$64,70 \pm 1,68$
LSTM		$85,79 \pm 3,51$

A.1.1 Fusão MLP com MLP

Os melhores modelos LOFAR e temporal pré-treinados possuem respectivamente: 30 e 30 neurônios na camada oculta. Como descrito na Subseção 4.5, um modelo composto por uma rede de fusão e dois modelos de entradas são criados. Os parâmetros dos modelos temporal e LOFAR pré-treinados são importados para os novos modelos. Durante o processo de treinamento da camada oculta da rede de fusão, os parâmetros dos modelos de entrada são congelados. A Tabela A.2 apresenta o índice SP do conjunto de validação pela variação do número de neurônios da rede de fusão. A primeira linha “–” mostra o resultado sem a rede de fusão, ou seja, as representações das últimas camadas de cada modelo de entrada são concatenadas e

conectadas a uma camada de saída *Softmax*. Analisando os valores médios do índice SP e a margem de erro, é possível observar que os valores estão bem próximos. O gráfico de violino da Figura A.1 mostra os valores do índice SP pela variação de neurônio da rede de fusão. Valores acima de 30 neurônios na camada oculta da rede de fusão não apresentam melhoras significativas. Sendo assim, adotando um critério de parcimônia com relação ao número de parâmetros do modelo, podemos escolher entre 10 e 30 neurônios na camada oculta. Os índices SP do conjunto de validação dos modelos são $(75,87 \pm 2,34)\%$ e $(78,50 \pm 2,65)\%$ respectivamente. Esses resultados são superiores aos obtidos individualmente pelo modelo temporal e similares aos do modelo LOFAR (vide a Tabela A.1).

Tabela A.2: Resultados experimentais do índice SP do conjunto de validação da fusão dos modelos MLP (águas profundas).

Neurônios da Rede de fusão	SP(%)
–	$66,63 \pm 1,87$
10	$75,86 \pm 2,38$
30	$78,71 \pm 2,43$
50	$79,15 \pm 2,43$
60	$79,51 \pm 2,25$

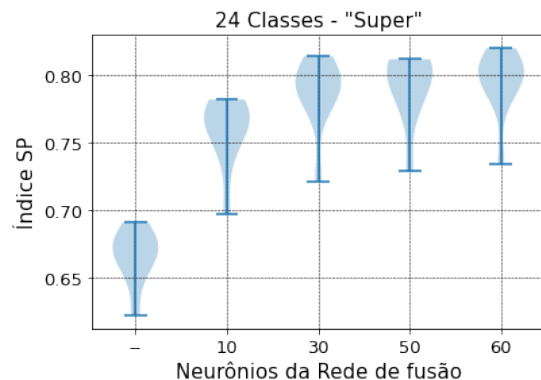


Figura A.1: Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos MLP (águas profundas).

A Figura A.2 mostra uma matriz de histogramas da saída da fusão dos modelos MLP com 10 e 30 neurônios na camada oculta da rede de fusão. Há uma frequência maior de valores próximos de “1” para os Verdadeiros Positivos e de “0” para os Verdadeiros Negativos na rede de fusão de 30 neurônios em comparação com a de 10 neurônios. A diferença mais perceptível está nas classes A e B. Esse comportamento foi observado em todos os *folds* (vide as Figuras C.54 a C.58).

A Figura A.3 apresenta a matriz de confusão dos conjuntos de treinamento e validação dos modelos com 10 e 30 neurônios na camada oculta da rede de fusão. Em ambos os modelos, o problema de diferenciação entre as classes A e B permanece. A classe B teve o pior desempenho no conjunto de validação com $(69,43 \pm 5,84)\%$ e $(69,40 \pm 4,96)\%$ de acurácia nos modelos com 10 e 30 neurônios respectivamente. O melhor desempenho foi na classe C no conjunto de treinamento e validação. A maior concentração dos erros são associados aos Falsos Positivos da classe A. A melhora mais significativa com o aumento dos neurônios é com relação a classe A.

A Figura A.4 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folds* do *dataset* (as curvas dos outros *folds* estão no

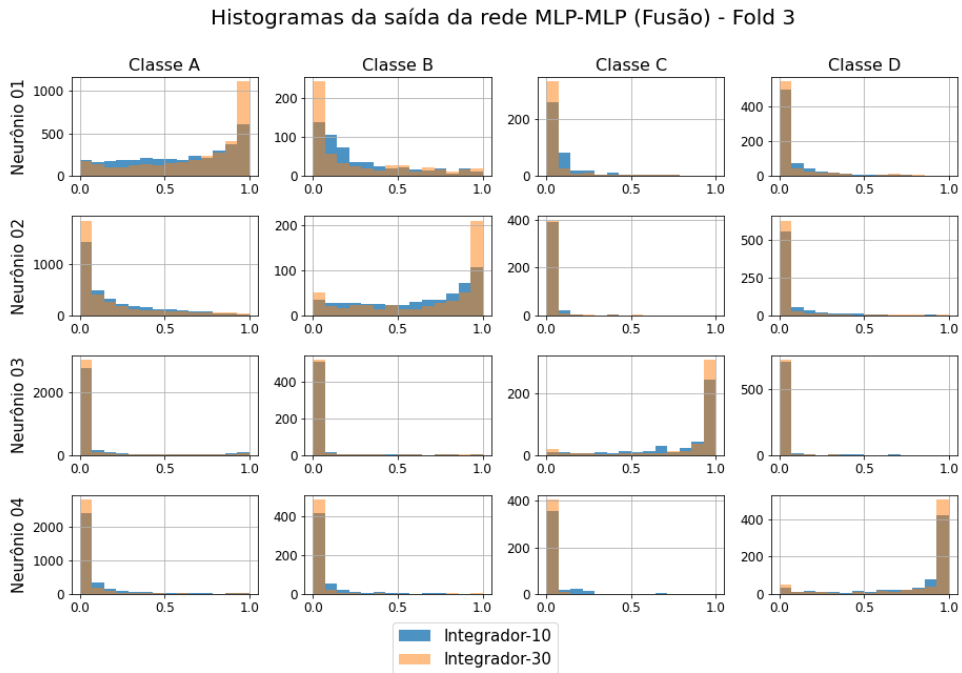


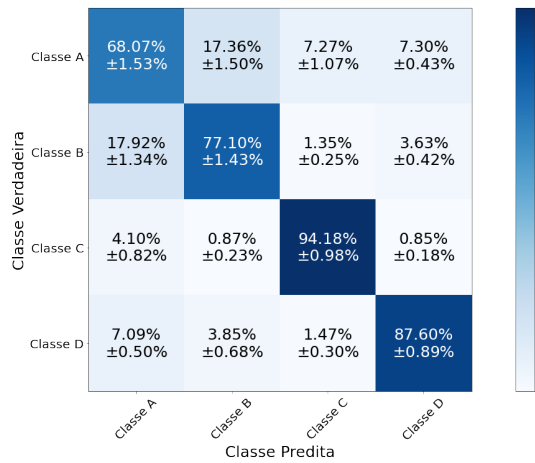
Figura A.2: Histogramas da saída da fusão dos modelos MLP, *fold* 3 (águas profundas).

Apêndice C.3.1. Nestes *folders*, anteriormente nos domínios isolados, os modelos apresentavam *overfitting* em torno de 20 a 30 épocas. Na fusão, muitos dos *folders* não alcançaram um valor mínimo de perda do conjunto de validação. O conjunto de treinamento não fornece informações suficientes para entender o problema, em relação ao conjunto de dados de validação, pois ainda permanece uma lacuna entre as curvas de perda de treinamento e validação. As curvas de perdas estão tendendo para um valor de estabilidade.

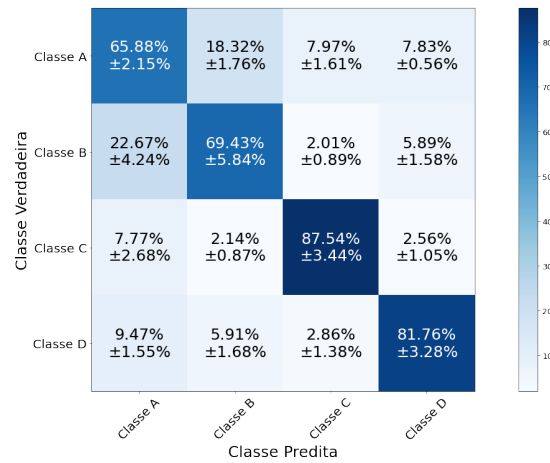
A Tabela A.3 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com 10 e 30 neurônios na camada oculta da rede de fusão. Apesar dos *Recalls* terem valores próximos, o modelo com 30 neurônios na rede de fusão apresenta um desempenho bem superior na acurácia e precisão.

Tabela A.3: Resultados experimentais do conjunto de validação da fusão dos modelos MLP (águas profundas)

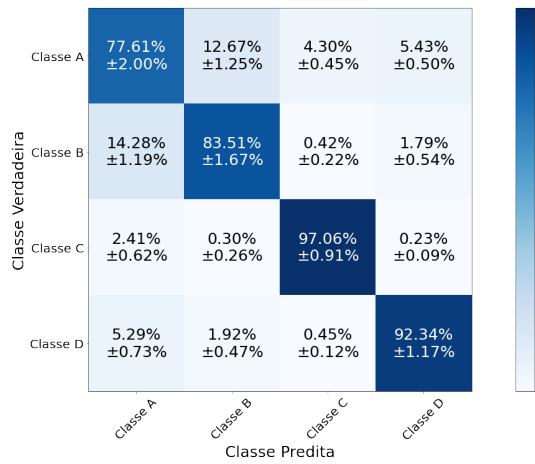
Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
10	$75,86 \pm 2,38$	$70,31 \pm 1,06$	$61,93 \pm 1,44$	$76,15 \pm 2,37$
30	$78,71 \pm 2,43$	$76,36 \pm 1,64$	$67,58 \pm 1,80$	$78,92 \pm 2,40$



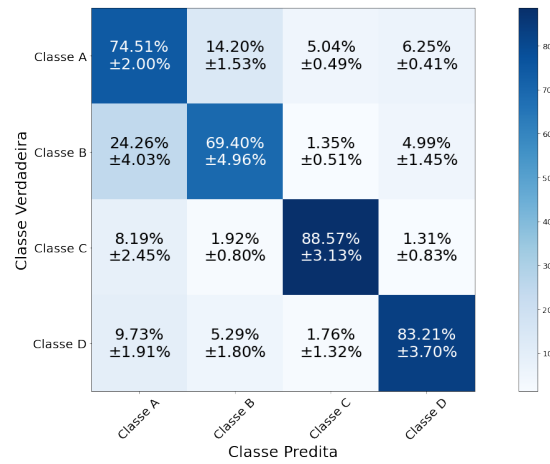
(a) Treinamento: Rede de fusão-10.



(b) Validação: Rede de fusão-10.

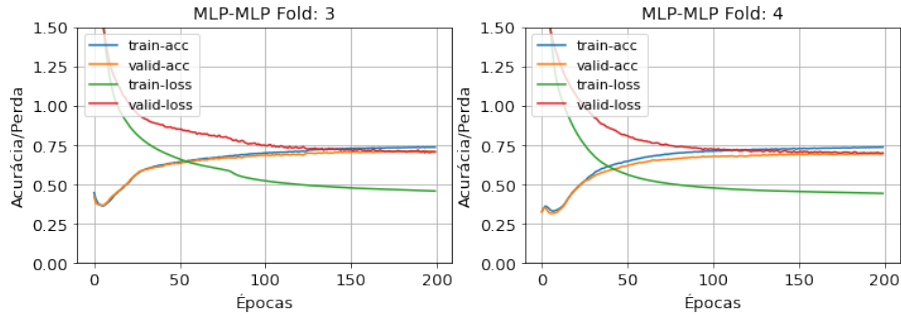


(c) Treinamento: Rede de fusão-30.

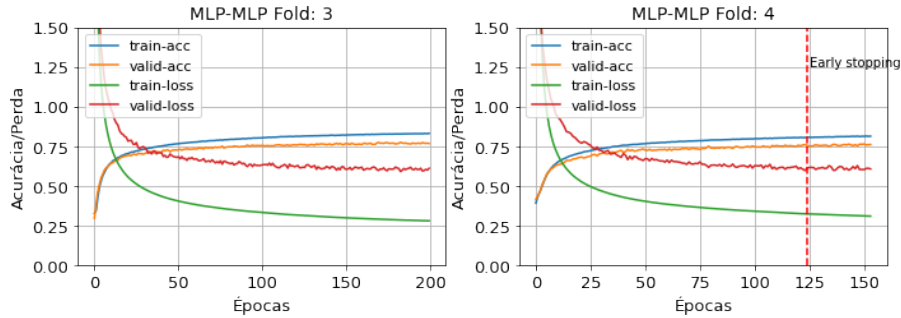


(d) Validação: Rede de fusão-30.

Figura A.3: Matriz de Confusão da fusão dos modelos MLP com 10 e 30 neurônios na rede de fusão (águas profundas).



(a) Rede de fusão com 10 neurônios.



(b) Rede de fusão com 30 neurônios.

Figura A.4: Curvas de treinamento e validação da fusão dos modelos MLP (águas profundas).

A.1.2 Fusão CNN com CNN

O melhor modelo convolucional pré-treinado é o modelo com duas camadas convolucionais de 8 filtros, uma camada *fully connected* de 20 neurônios e *kernel* 3, em ambos os domínios. A Tabela A.4 apresenta o índice SP do conjunto de validação pela variação do número de neurônios da rede de fusão. Considerando os valores médios do índice SP e a margem de erro, é possível observar uma melhora com a adição da rede de fusão, entretanto, o aumento do número de neurônios não melhora significativamente os resultados. O gráfico de violino da Figura A.5 mostra os valores do índice SP pela variação de neurônio da rede de fusão. Muitos dos resultados com a rede de fusão estão concentrados em torno de 82% com *outliers* entre 74% e 76%. Isto posto, o modelo com 10 neurônios na camada oculta da rede de fusão é o mais apropriado. Esses resultados são bem superiores aos resultados obtidos no modelo temporal convolucional (vide a Tabela A.1).

A Figura A.6 mostra uma matriz de histogramas da saída da fusão dos modelos CNN sem a rede de fusão e com 10 neurônios na camada oculta da rede. Os dados sem a rede serão apresentados para comparação da eficiência da rede de fusão. Há uma frequência maior de valores próximos de “1” para os Verdadeiros Positivos no modelo com a rede em comparação ao modelo sem. Esse comportamento foi observado em todos os *folds* (vide as Figuras C.59 a C.63).

Tabela A.4: Resultados experimentais do índice SP do conjunto de validação da fusão dos modelos CNN (águas profundas).

Neurônios da Rede de fusão	SP(%)
–	$78,29 \pm 2,25$
10	$81,37 \pm 2,62$
30	$82,22 \pm 2,39$
50	$81,94 \pm 2,63$
100	$81,61 \pm 2,69$

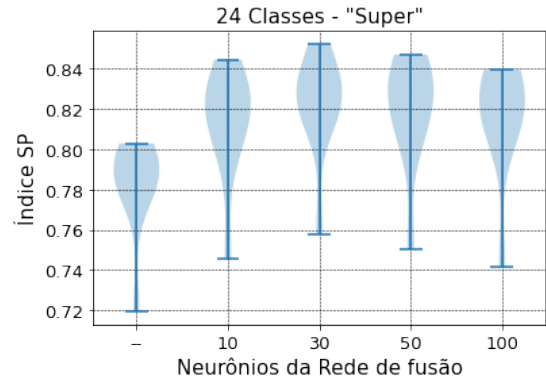


Figura A.5: Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos CNN (águas profundas).

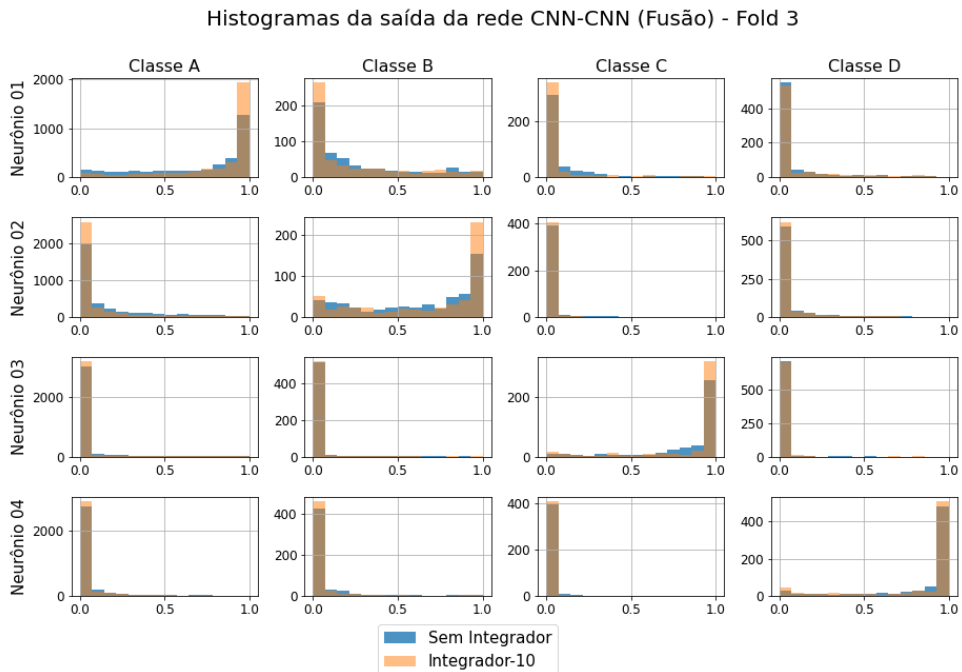


Figura A.6: Histogramas da saída da fusão dos modelos CNN, *fold 3* (águas profundas).

A Figura A.7 apresenta a matriz de confusão dos conjuntos de treinamento e validação dos modelos CNN sem e com a rede de fusão de 10 neurônios. O modelo com a rede teve uma melhora durante o treinamento em todas as classes. No conjunto de validação, a melhora significativa foi na classe A com um aumento de cerca de 7%. A maior concentração dos erros são associados aos Falsos Positivos da classe A.

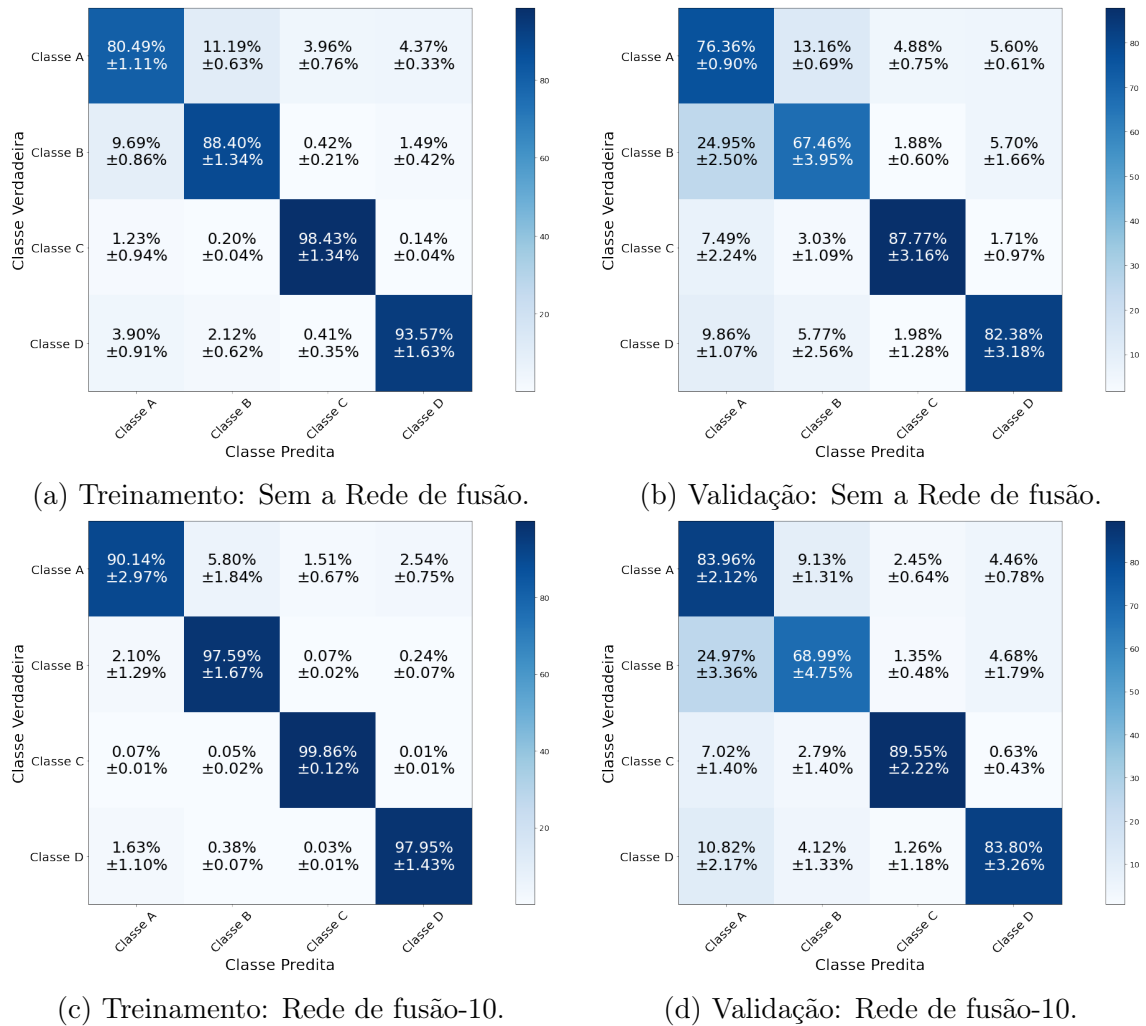
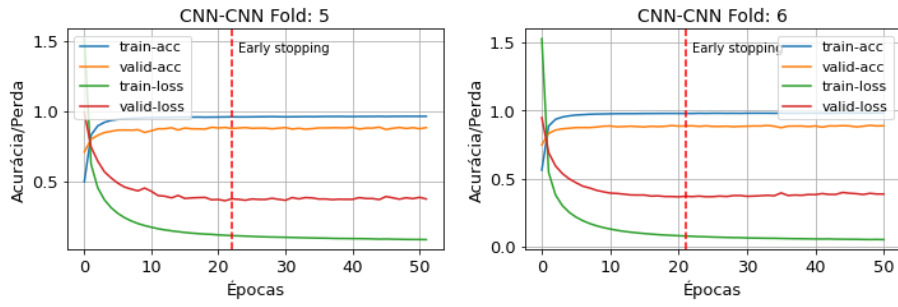


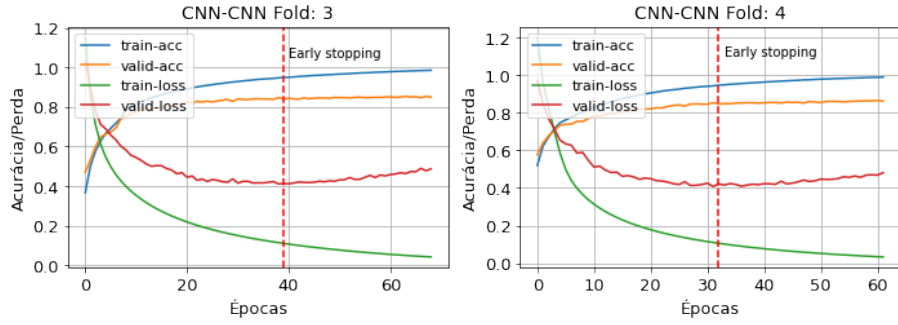
Figura A.7: Matriz de Confusão da fusão dos modelos CNN sem e com rede de fusão de 10 neurônios (águas profundas).

A Figura A.8 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folds* do *dataset* dos modelos CNN sem e com rede de fusão de 10 neurônios (as curvas dos outros *folds* estão no Apêndice C.3.1. O conjunto de treinamento não fornece informações suficientes para entender o problema em relação ao conjunto de dados de validação, pois existe uma lacuna entre as curvas de treinamento e validação.

A Tabela A.5 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com e sem a rede de fusão. O modelo com a rede tem uma acurácia e uma



(a) Sem a Rede de fusão.



(b) Rede de fusão com 10 neurônios.

Figura A.8: Curvas de treinamento e validação da fusão dos modelos CNN (águas profundas).

precisão melhor do que o modelo sem a rede de fusão.

Tabela A.5: Resultados experimentais do conjunto de validação da fusão dos modelos CNN (águas profundas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
--	$78,29 \pm 2,25$	$77,21 \pm 1,23$	$68,05 \pm 1,78$	$78,49 \pm 2,22$
10	$81,37 \pm 2,62$	$82,80 \pm 2,25$	$75,06 \pm 2,92$	$81,57 \pm 2,55$

A.1.3 Fusão MLP com LSTM

Os melhores modelos LOFAR e temporal pré-treinados possuem respectivamente, 30 neurônios e 75 células na camada oculta. A Tabela A.6 apresenta o índice SP do conjunto de validação pela variação do número de neurônios da rede de fusão. O resultado não apresenta melhoras com o aumento do número de neurônios. A utilização da rede de fusão também não causa grande mudança no índice SP, mas nas outras figuras de mérito de classificação iremos observar algumas melhoras. O gráfico violino da Figura A.9 mostra os valores do índice SP pela variação do número de neurônios da rede de fusão. A maioria dos resultados estão concentrados na parte superior do diagrama. Iremos utilizar o modelo com 10 neurônios na camada oculta da rede de fusão, mas em termos de comparação traremos também os resultados do modelo sem a rede de fusão.

Tabela A.6: Resultados experimentais do índice SP do conjunto de validação da fusão dos modelos (águas profundas).

Neurônios da Rede de fusão	SP(%)
–	88,18 ± 3,28
10	89,07 ± 3,30
30	89,50 ± 3,39
50	89,64 ± 3,21
100	89,54 ± 3,50

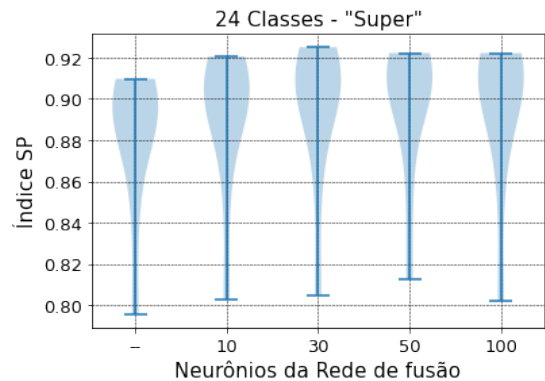


Figura A.9: Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos MLP com LSTM (águas profundas).

A Figura A.10 mostra uma matriz de histogramas da saída da fusão do modelo MLP e LSTM sem e com a rede de fusão. Nos Verdadeiros Negativos, ambos os modelos apresentam desempenhos semelhantes, a diferença está no Verdadeiros Positivos, o modelo com a rede apresenta uma frequência maior de valores próximos de “1”. Esse comportamento foi observado em todos os *folds* (vide as Figuras C.64 a C.68).

A Figura A.7 apresenta a matriz de confusão dos conjuntos de treinamento e validação da fusão do modelo MLP e LSTM sem e com rede de fusão. Os modelos apresentam valores próximos. O melhor desempenho é da classe C com a acurácia de (95,75 ± 2,20)% e (96,03 ± 1,95)% nos modelos sem e com a rede respectivamente. A quantidade de Falsos Positivos da classe A ainda permanecem.

A Figura A.12 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folds* do *dataset* da fusão dos modelos MLP e LSTM com

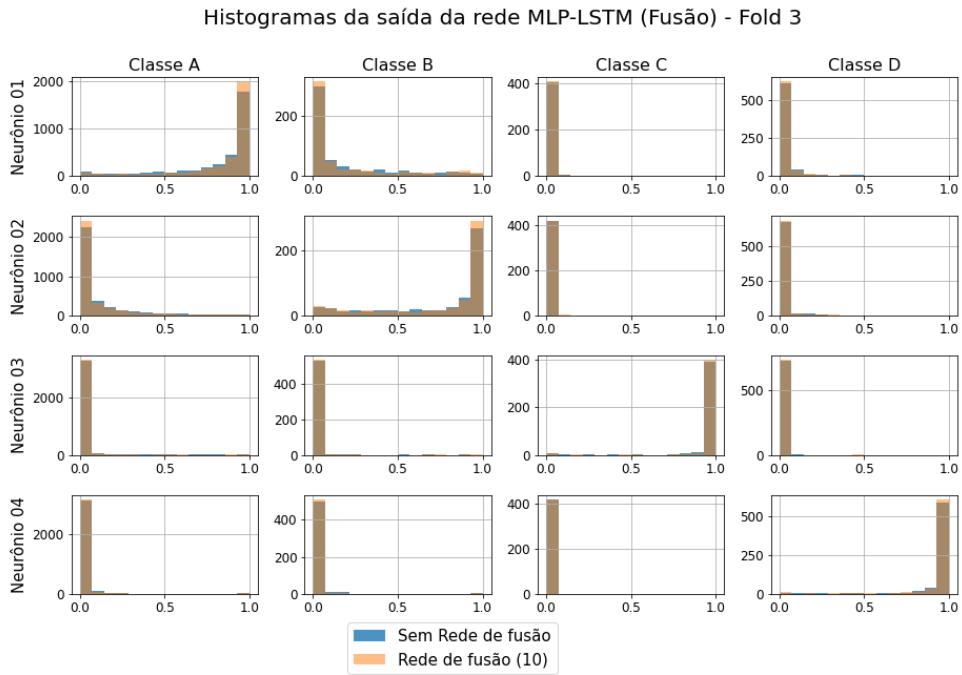


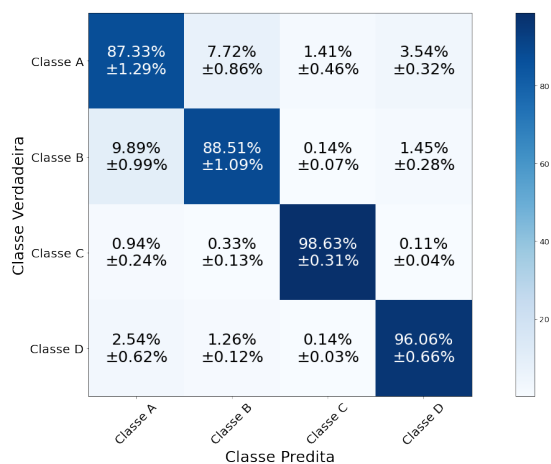
Figura A.10: Histogramas da saída da fusão dos modelos MLP com LSTM, *fold* 3 (águas profundas).

e sem a rede de fusão (as curvas dos outros *folders* estão no Apêndice C.3.1). Em ambos os modelos, o conjunto de treinamento não fornece informações suficientes para entender o problema em relação ao conjunto de dados de validação, pois existe uma lacuna entre as curvas de treinamento e validação. As curvas apresentam tendência de estabilização.

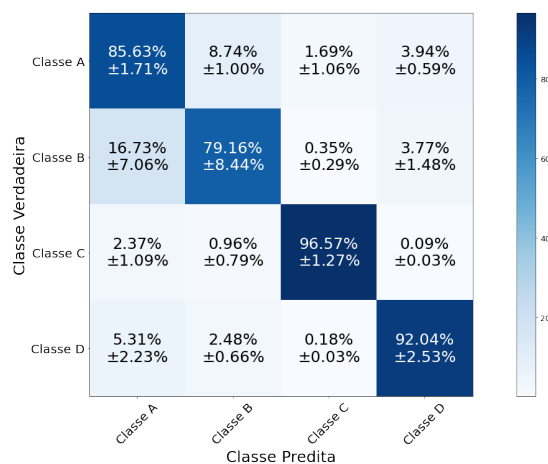
A Tabela A.7 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com e sem a rede de fusão. O modelo com a rede tem valores médios melhores do que o modelo sem a rede de fusão.

Tabela A.7: Resultados experimentais do conjunto de validação da fusão dos modelos MLP com LSTM (águas profundas)

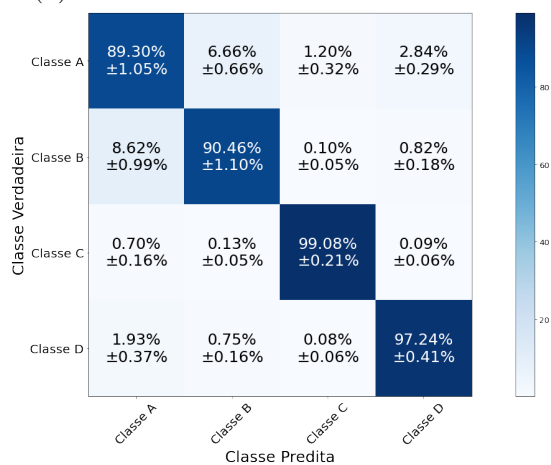
Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
--	86,23 ± 3,94	84,07 ± 3,83	76,90 ± 4,80	86,41 ± 3,84
10	87,61 ± 3,44	85,82 ± 3,17	79,13 ± 4,10	87,77 ± 3,35



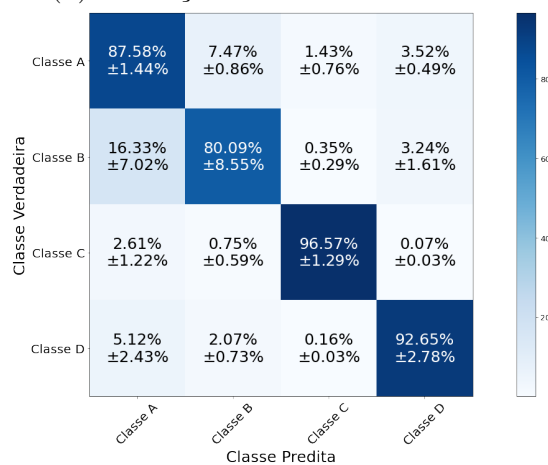
(a) Treinamento: Sem a Rede de fusão.



(b) Validação: Sem a Rede de fusão.

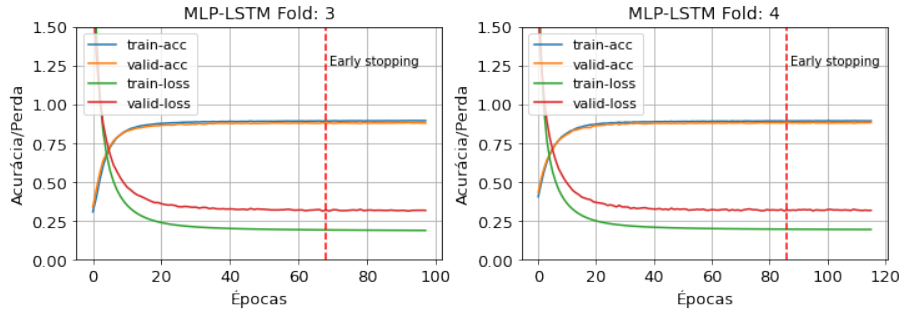


(c) Treinamento: Sem a Rede de fusão-10.

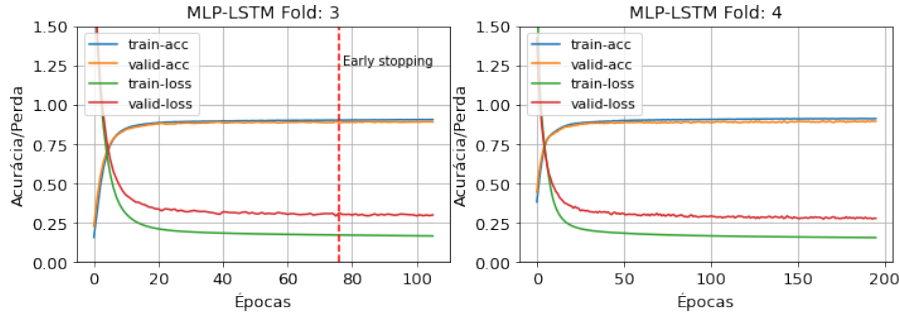


(d) Validação: Rede de fusão-10.

Figura A.11: Matriz de Confusão da fusão dos modelos MLP com LSTM sem e com a rede de fusão com 10 neurônios (águas profundas).



(a) Sem a Rede de fusão.



(b) Rede de fusão com 10 neurônios.

Figura A.12: Curvas de treinamento e validação da fusão dos modelos MLP com LSTM (águas profundas).

A.1.4 Fusão CNN com LSTM

O melhor modelo LOFAR convolucional pré-treinado é o modelo com duas camadas convolucionais de 8 filtros, uma camada *fully connected* com 20 neurônios e *kernel* 3. O modelo temporal é o modelo LSTM com 75 células na camada oculta. A Tabela A.8 apresenta o índice SP do conjunto de validação pela variação do número de neurônios da rede de fusão. Não houve uma melhora significativa do índice SP com a adição da rede de fusão e o aumento do número de neurônios. O gráfico de violino da Figura A.13 mostra os valores do índice SP pela variação de neurônio do da rede. Em todos os casos, temos um *outliers* em torno de 80% e muitos dos resultados se concentram próximos de 88%. O modelo selecionado é o modelo com 10 neurônios na rede, com o SP de $(88, 81 \pm 3, 47)\%$, mas para comparação os resultados do modelo sem rede de fusão serão apresentados.

A Figura A.14 mostra uma matriz de histogramas da saída de um *fold* de treinamento da fusão dos modelos CNN e LSTM sem e com rede de fusão com 10 neurônios. Ambos os modelos apresentam uma grande frequência de valores próximos de “1” para o Verdadeiros Positivos e próximos de “0” para os Verdadeiros Negativos. Apenas para a classe A, a rede de fusão mostra uma pequena melhora. Isso reflete uma boa classificação com valores próximos de 90% de SP. Esse comportamento foi observado em todos os *folds* (vide as Figuras C.69 a C.73).

A Figura A.15 apresenta a matriz de confusão dos conjuntos de treinamento e

Tabela A.8: Resultados experimentais do índice SP do conjunto de validação da fusão dos modelos (águas profundas).

Neurônios da Rede de fusão	SP(%)
–	$88,84 \pm 3,38$
10	$88,81 \pm 3,47$
30	$88,66 \pm 3,27$
50	$88,49 \pm 3,35$
100	$88,57 \pm 3,33$

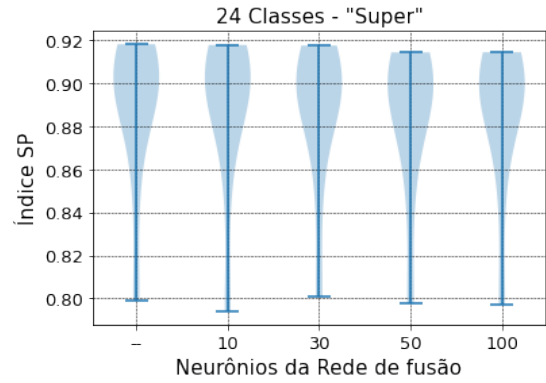


Figura A.13: Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos CNN com LSTM (águas profundas).

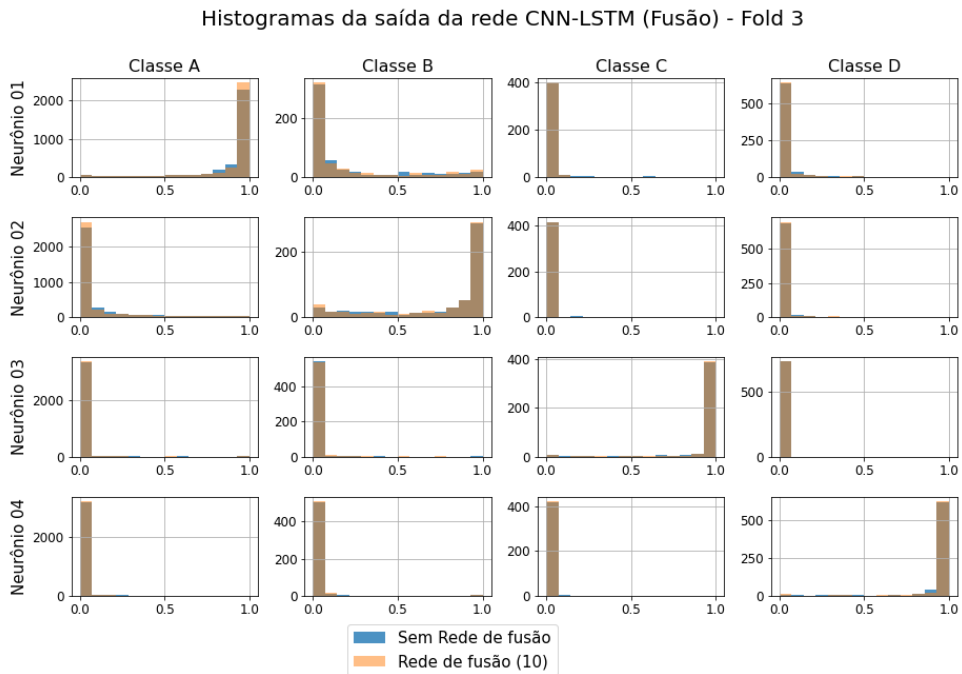
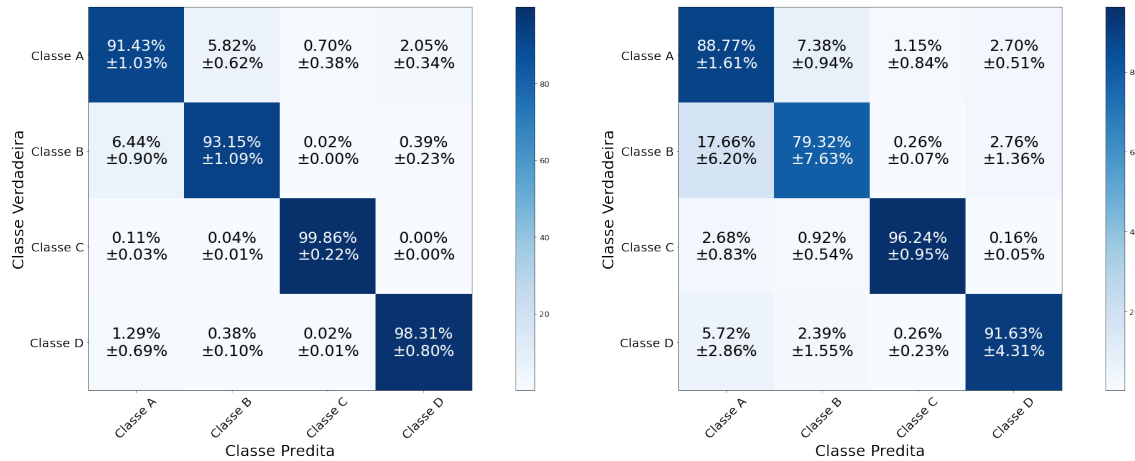


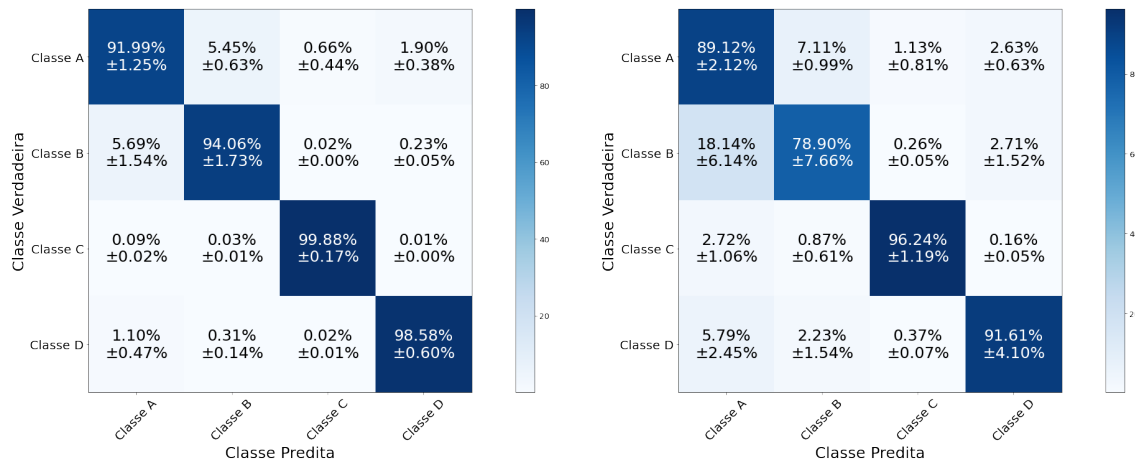
Figura A.14: Histogramas da saída da fusão do modelo CNN com LSTM (*fold* 3).

validação da fusão dos modelos CNN e LSTM sem rede de fusão e com 10 neurônios na camada oculta da rede. Não há diferença estatística entre os modelos. A melhor eficiência durante o processo de validação é da classe C com acurácia de $(96,24 \pm 0,95)\%$ e $(96,24 \pm 1,19)\%$ para os modelos sem e com rede de fusão respectivamente. Os Falsos Positivos da classe A são os maiores erros do modelo.



(a) Treinamento: Sem Rede de fusão.

(b) Validação: Sem Rede de fusão.



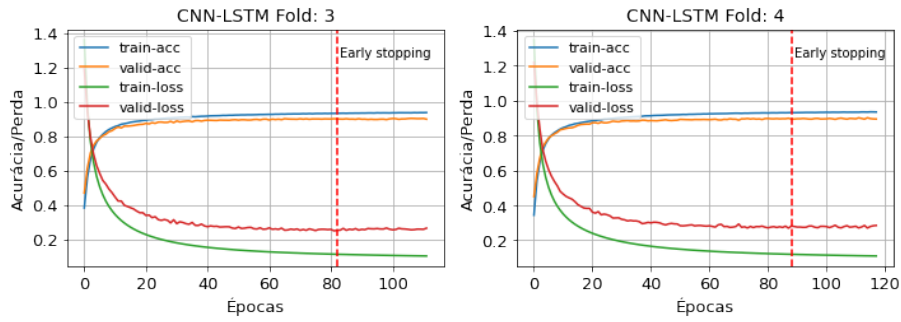
(c) Treinamento: Rede de fusão (10).

(d) Validação: Rede de fusão (10).

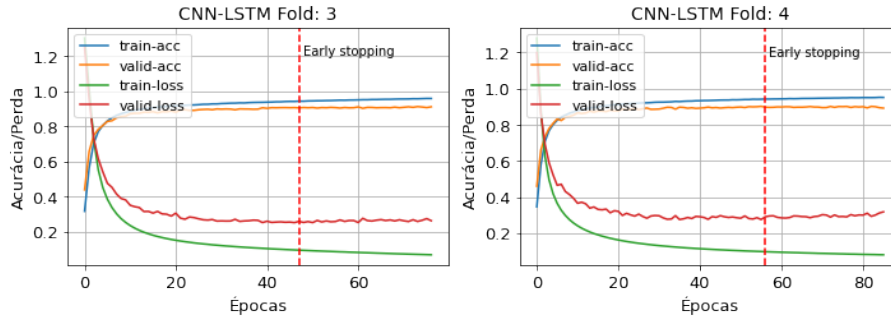
Figura A.15: Matriz de Confusão da fusão dos modelos CNN com LSTM sem e com uma rede de fusão com 10 neurônios (águas profundas).

A Figura A.16 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folds* do *dataset* (as curvas dos outros *folds* estão no Apêndice C.3.1). Com a adição da rede de fusão, o *overfitting* passa de valores em torno de 80 épocas para cerca de 50 épocas. Em todas as técnicas (MLP, CNN e LSTM), o conjunto de treinamento não fornece informações suficientes para entender o problema, em relação ao conjunto de dados de validação. Esse processo foi observado também na fusão dos modelos CNN e LSTM. Devido a lacuna entre as curvas de treinamento e validação.

A Tabela A.7 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall*



(a) Sem Rede de fusão.



(b) Rede de fusão (10).

Figura A.16: Curvas de treinamento e validação da fusão dos modelos CNN com LSTM (águas profundas).

dos modelos com e sem a rede de fusão. Assim como constatado até o momento, os resultados dos dois modelos são semelhantes, mas para dar seguimento a proposta do trabalho, iremos utilizar o modelo com a rede de fusão.

Tabela A.9: Resultados experimentais do conjunto de validação da fusão dos modelos CNN com LSTM (águas profundas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
--	$88,84 \pm 3,38$	$88,78 \pm 2,29$	$83,35 \pm 3,39$	$88,99 \pm 3,24$
10	$88,81 \pm 3,47$	$88,98 \pm 2,60$	$83,62 \pm 3,80$	$88,97 \pm 3,34$

A.2 Águas Rasas

A Tabela A.10 apresenta os resultados dos índices SP do conjunto de validação dos modelos LOFAR e temporal escolhidos (vide o Capítulo 5). Esses modelos são entradas de uma rede MLP (rede de fusão).

Tabela A.10: Resultados experimentais do índice SP dos modelos no conjunto de validação (águas rasas)

Modelos	Índice SP (%)	
	LOFAR-1024	tempo-1024
MLP	$74,60 \pm 2,54$	$59,06 \pm 7,51$
CNN	$77,74 \pm 3,84$	$73,86 \pm 11,55$
LSTM		$70,75 \pm 7,15$

A.2.1 Fusão MLP com MLP

Os melhores modelos LOFAR e temporal pré-treinados possuem 10 neurônios na camada oculta. A Tabela A.11 apresenta o índice SP do conjunto de validação pela variação do número de neurônios da rede de fusão. Considerando os valores médios e a margem de erro do índice SP, os resultados estão próximos. O gráfico de violino da Figura A.17 mostra os valores do índice SP pela variação de neurônios da rede de fusão. Analisando o gráfico de violino, a adição da rede de fusão melhora o modelo. O melhor resultado de SP é o de 10 neurônios na camada oculta da rede de fusão. Não houve uma melhora considerável dos valores médios do índice SP, apesar do aumento do número de neurônios. O índice SP do conjunto de validação do modelo escolhido é $(77,96 \pm 3,65)\%$. Esse resultado é superior aos resultados obtidos no modelo temporal. No modelo LOFAR, o valor médio do índice SP é superior ao modelo MLP, mas considerando a margem de erro, os valores são similares.

A Figura A.18 mostra uma matriz de histogramas da saída da fusão dos modelos MLP com 10 neurônios na camada oculta da rede de fusão. Os histogramas são de apenas um dos *folds*, os demais estão no Apêndice C.4.2. Em relação a classe B, há uma grande frequência de eventos para os valores de alta probabilidade. Para a classe A, essas frequências estão mais distribuídas ao longo de histograma.

A Figura A.19 apresenta a matriz de confusão dos conjuntos de treinamento e validação da fusão dos modelos MLP com 10 neurônios na camada oculta da rede de fusão. Durante o treinamento a classe A teve um bom desempenho, mas esse desempenho teve uma queda acentuada no conjunto de validação. O número pequeno de amostras para essa classe pode ser um dos motivos dessa queda. A classe B manteve valores bem parecidos durante o treinamento e validação.

Tabela A.11: Resultados experimentais do índice SP do conjunto de validação da fusão dos modelos (águas rasas).

Neurônios da Rede de fusão	SP(%)
–	$71,74 \pm 2,28$
10	$77,96 \pm 3,65$
30	$77,97 \pm 3,92$
50	$78,08 \pm 3,79$
60	$77,77 \pm 3,34$

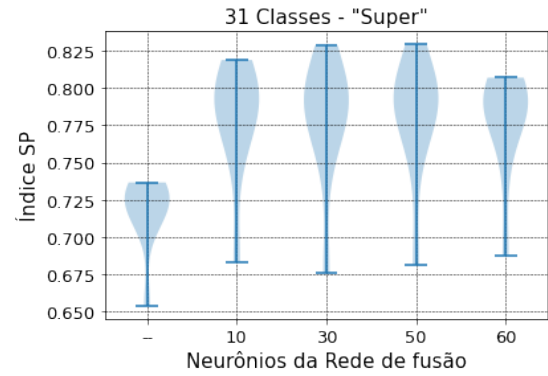


Figura A.17: Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos MLP (águas rasas).

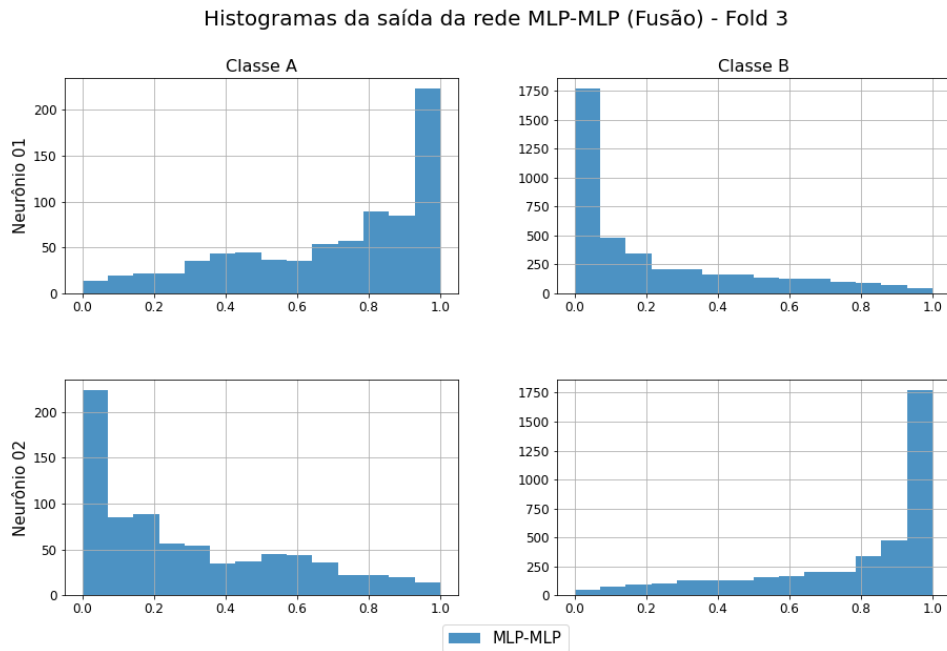


Figura A.18: Histogramas da saída da fusão dos modelos MLP, *fold* 3 (águas rasas).

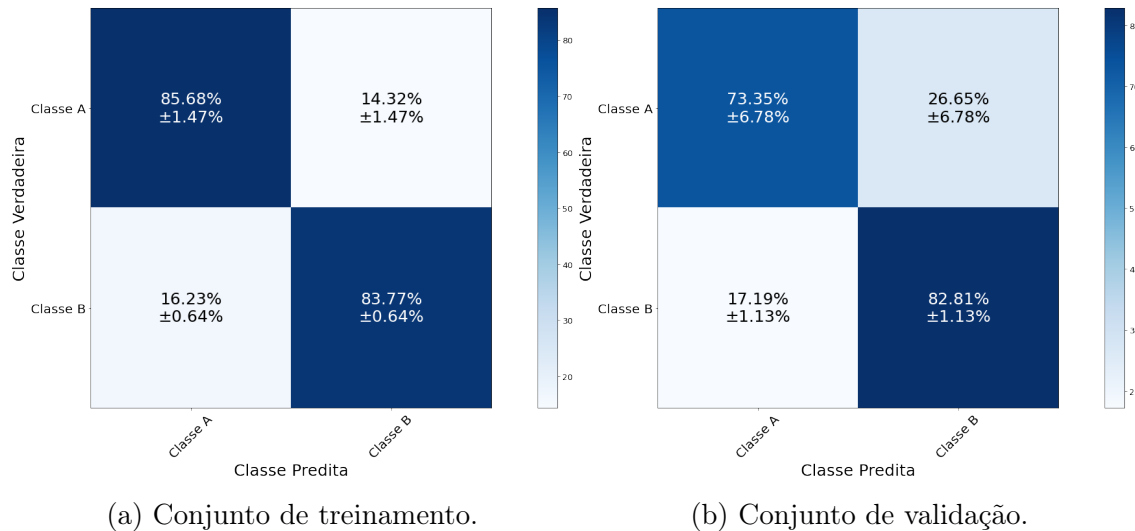


Figura A.19: Matriz de Confusão da fusão dos modelos MLP (águas rasas).

A Figura A.20 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folders* do *dataset* (as curvas dos outros *folders* estão no Apêndice C.3.2). O treinamento foi estável, tendendo para um valor fixo. Apesar da lacuna entre as curvas de treinamento e validação, o treinamento aparenta ter um bom ajuste.

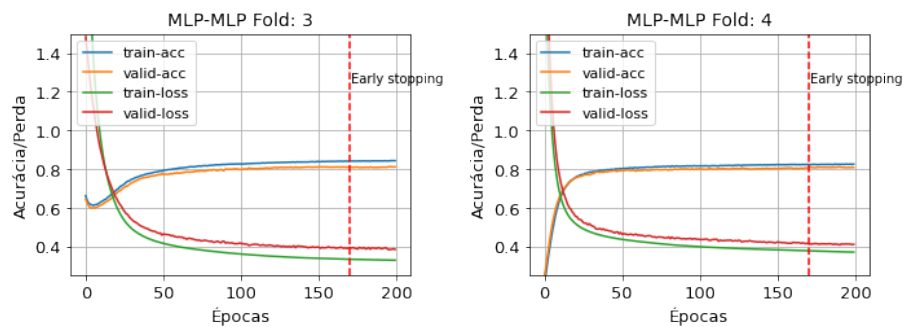


Figura A.20: Curvas de treinamento e validação da fusão dos modelos MLP (águas rasas).

Os resultados de classificação deste modelo são: Acurácia de $(81,26 \pm 1,51)\%$, Precisão de $(69,74 \pm 2,16)\%$ e *Recall* de $(78,08 \pm 3,48)\%$.

A.2.2 Fusão CNN com CNN

Os melhores modelos LOFAR e temporal convolucionais pré-treinados são com duas camadas convolucionais de oito filtros, uma camada *fully connected* de 10 neurônios e *kernel* 3 (8,8,10,3). A Tabela A.12 apresenta o índice SP do conjunto de validação pela variação do número de neurônios da rede de fusão. Em todas as ocasiões, os resultados estão dentro da mesma faixa de valores. O gráfico de violino da Figura A.21 mostra os valores do índice SP pela variação de neurônios da rede de fusão. Ele mostra que as distribuições estão concentradas em valores acima de 75% e *outliers* próximos a 65%. O intuito do trabalho é avaliar o efeito da rede de fusão. Portanto, o modelo com 10 neurônios na camada oculta da rede de fusão é mais apropriado, mas iremos comparar os resultados de classificação com o modelo sem a rede de fusão.

Tabela A.12: Resultados experimentais do índice SP do conjunto de validação da fusão dos modelos (águas rasas).

Neurônios da Rede de fusão	SP(%)
–	80,92 ± 5,38
10	81,33 ± 6,06
30	80,89 ± 5,19
50	80,94 ± 5,71
100	80,64 ± 5,13

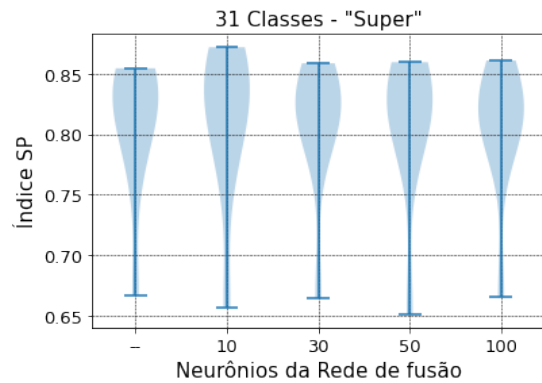


Figura A.21: Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos CNN (águas rasas).

A Figura A.22 mostra uma matriz de histogramas da saída da fusão dos modelos CNN com 10 neurônios na camada oculta da rede de fusão e sem a rede (apenas o *fold* 3, os demais estão no Apêndice C.4.2). Em todos os *folds* é possível observar uma frequência de eventos com valores próximos de “1” e “0”, no modelo com 10 neurônios na rede de fusão.

A Figura A.23 apresenta a matriz de confusão dos conjuntos de treinamento e validação da fusão dos modelos CNN com 10 neurônios na camada oculta da rede de fusão e sem a rede. Comparando os conjuntos de treinamento, a classe A teve um desempenho levemente superior, entretanto, essa melhora não está refletida no conjunto de validação. Os valores de eficiência da classe A são: (76,52 ± 9,70)% e (75,88 ± 10,19)% para os modelos sem e com a rede de fusão respectivamente. O alto número de Falsos Positivos da classe B chama a atenção

A Figura A.24 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folds* do *dataset* da fusão dos modelos convolucionais (as

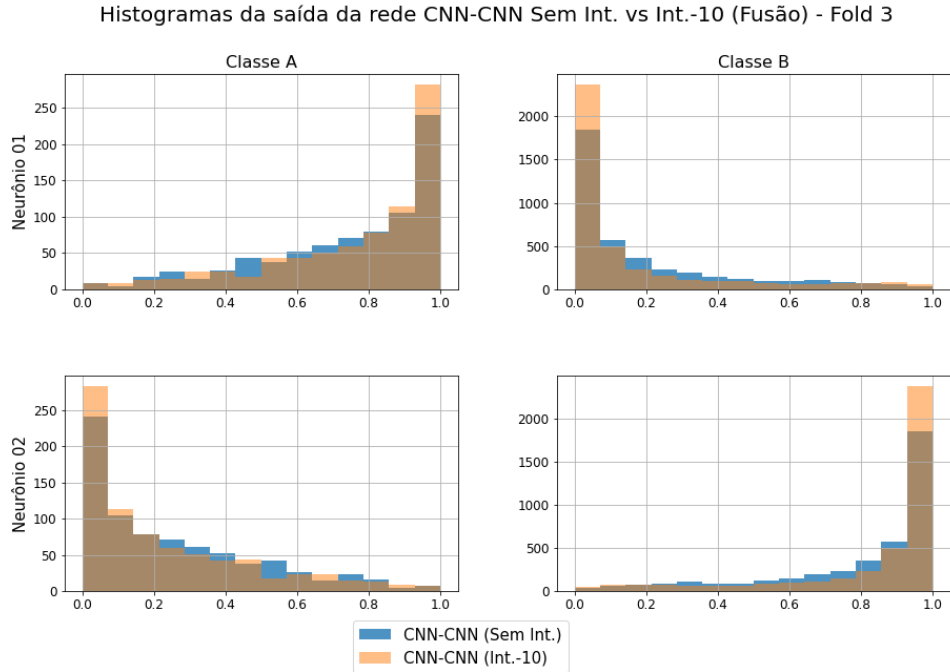


Figura A.22: Histogramas da saída da fusão dos modelos CNN, *fold* 3 (águas rasas).

curvas dos outros *folds* estão no Apêndice C.3.2). As curvas do modelo sem a rede de fusão apresentam um bom ajuste. Isso significa que, a lacuna entre as curvas de perda é mínima entre os dois valores finais e a perda dos conjuntos de treinamento e validação diminuem até um ponto de estabilidade. Com a rede de fusão, o modelo começa a apresentar *overfitting* entre 30 e 60 épocas de treinamento.

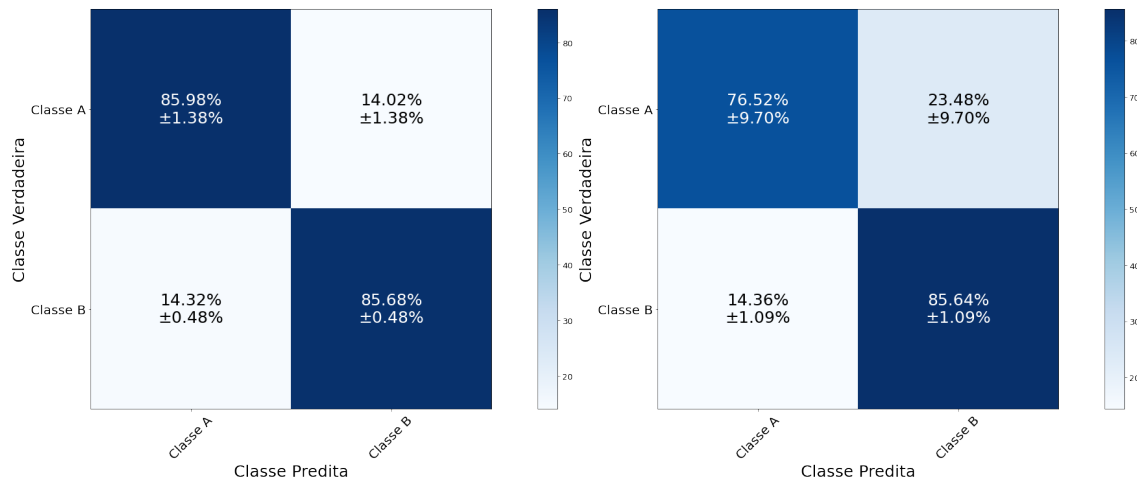
A Tabela A.13 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com e sem a rede de fusão. O modelo com a rede de fusão tem uma melhora pouco significativa nos valores médios e são bem semelhantes se considerarmos a margem de erro. Para esse estudo iremos utilizar o modelo com a rede de fusão.

Tabela A.13: Resultados experimentais do conjunto de validação da fusão dos modelos CNN com e sem a rede de fusão (águas rasas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
--	$80,92 \pm 5,38$	$84,15 \pm 2,14$	$72,87 \pm 3,26$	$81,07 \pm 5,10$
10	$81,33 \pm 6,06$	$85,31 \pm 2,92$	$74,18 \pm 4,18$	$81,51 \pm 5,78$

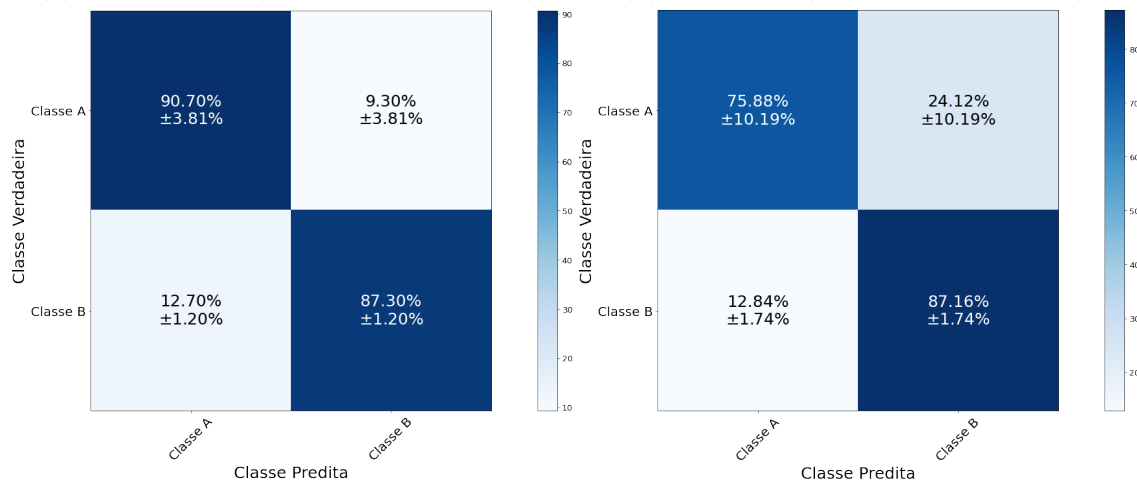
A.2.3 Fusão MLP com LSTM

Os melhores modelos LOFAR e temporal pré-treinados possuem respectivamente, 10 neurônios e 50 células na camada oculta. A Tabela A.14 apresenta o índice SP do conjunto de validação pela variação do número de neurônios da rede de fusão. A adição da rede de fusão e o aumento de neurônios na camada oculta da rede não



(a) Treinamento (Sem a Rede de fusão).

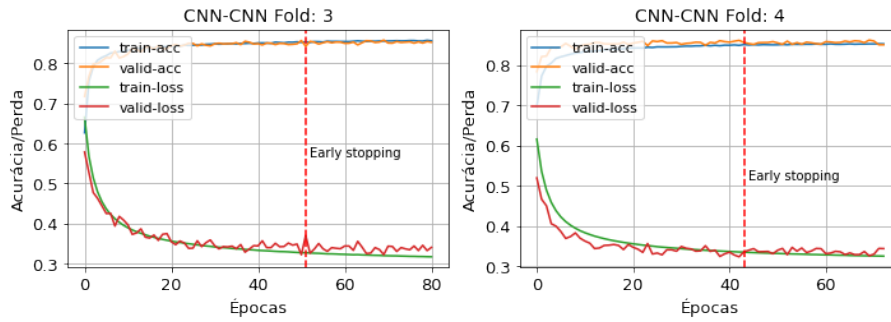
(b) Validação (Sem a Rede de fusão).



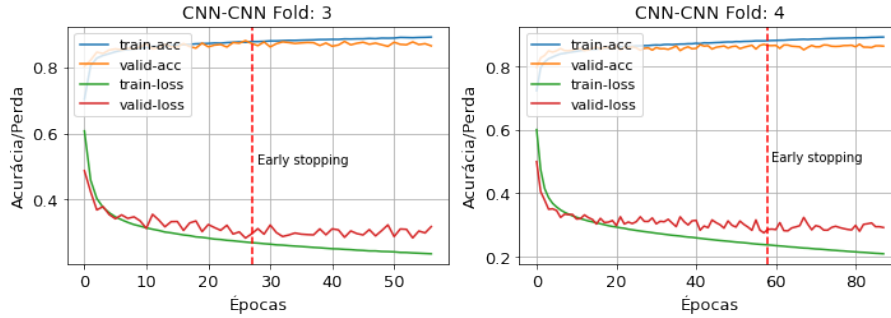
(c) Treinamento (Rede de Fusão-10).

(d) Validação (Rede de Fusão-10).

Figura A.23: Matriz de Confusão da fusão dos modelos CNN sem e com rede de fusão de 10 neurônios (águas rasas).



(a) Modelo (Sem a Rede de fusão).



(b) Modelo (Rede de Fusão-10).

Figura A.24: Curvas de treinamento e validação da fusão dos modelos CNN com e sem a rede de fusão (águas rasas).

melhoram a margem de erro do modelo. Entretanto, é possível observar um aumento do valor médio com a adição da rede de fusão. O gráfico violino da Figura A.25 mostra os valores do índice SP pela variação do número de neurônios da rede de fusão. Os resultados estão concentrados em valores em torno de 81%, com *outliers* em torno de 70%. Sendo assim, o modelo com 10 neurônios na rede de fusão é o mais apropriado neste fusão.

A Figura A.26 mostra uma matriz de histogramas da saída de um *fold* de treinamento da fusão dos modelos MLP e LSTM com 10 neurônios na rede de fusão e sem rede. O modelo com a rede de fusão apresenta uma frequência muito maior de valores próximos de “1” para os Verdadeiros Positivos e próximos de “0” para os Verdadeiros Negativos, isso em relação ao modelo sem a rede de fusão. Esse comportamento foi observado em todos os *folds* (vide as Figuras C.109 a C.113).

A Figura A.27 apresenta a matriz de confusão dos conjuntos de treinamento e validação da fusão dos modelos MLP e LSTM com 10 neurônios na camada oculta da rede de fusão e sem a rede. Durante o processo de treinamento, a classe A teve uma melhora de cerca de 8% e os Falsos Positivos reduziram em torno de 50% com a adição da rede de fusão. No conjunto de validação, a melhora significativa foi na classe B, com a acurácia de acerto de $(82,49 \pm 1,65)\%$.

A Figura A.28 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folds* do *dataset* da fusão dos modelos convolucionais (as

Tabela A.14: Resultados experimentais do índice SP do conjunto de validação da fusão dos modelos (águas rasas).

Neurônios da Rede de fusão	SP(%)
–	$76,35 \pm 3,42$
10	$80,17 \pm 3,82$
30	$79,83 \pm 3,78$
50	$80,14 \pm 3,75$
100	$80,02 \pm 3,94$

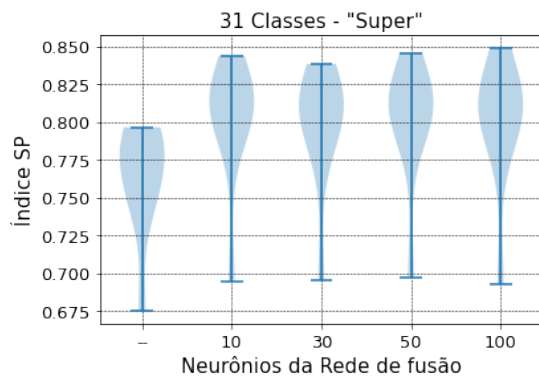


Figura A.25: Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos MLP com LSTM (águas rasas).

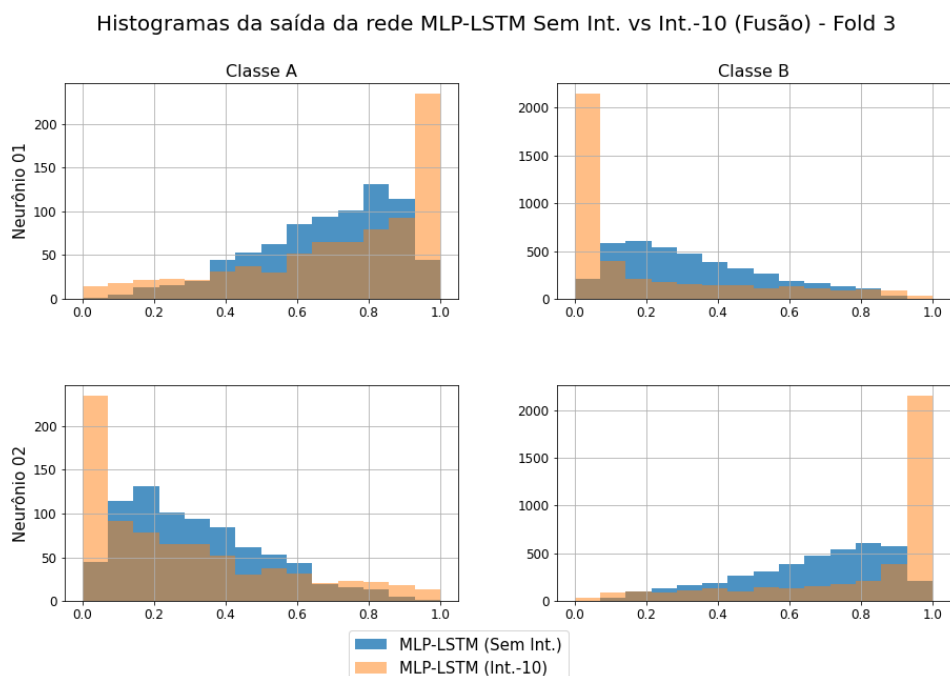
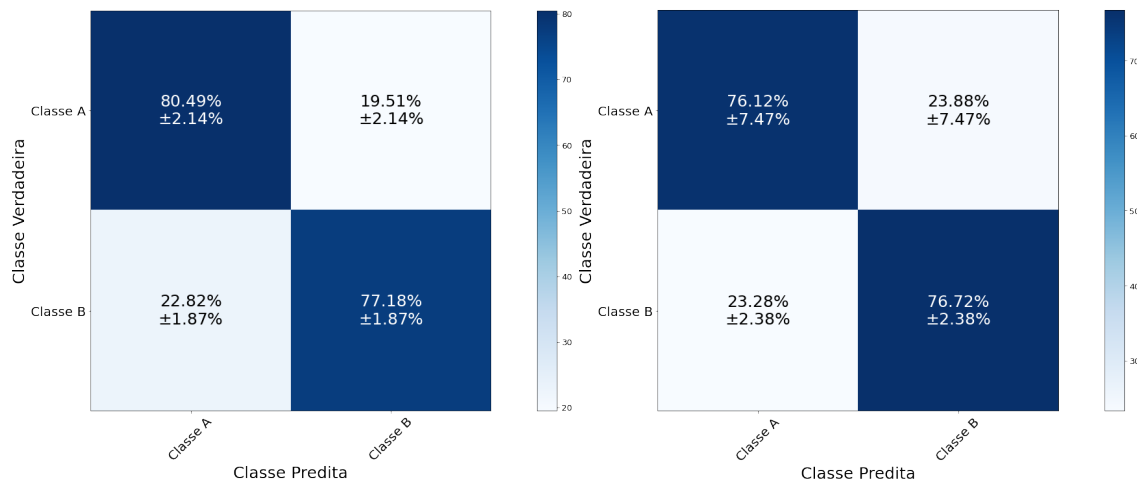
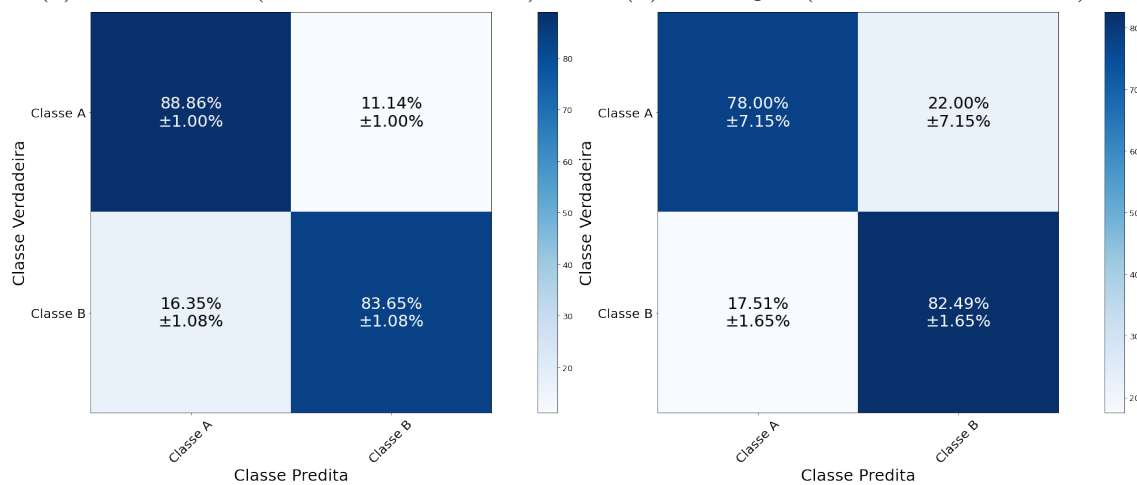


Figura A.26: Histogramas da saída da fusão dos modelos MLP com LSTM, *fold* 3 (águas rasas).



(a) Treinamento (Sem a Rede de fusão).

(b) Validação (Sem a Rede de fusão).

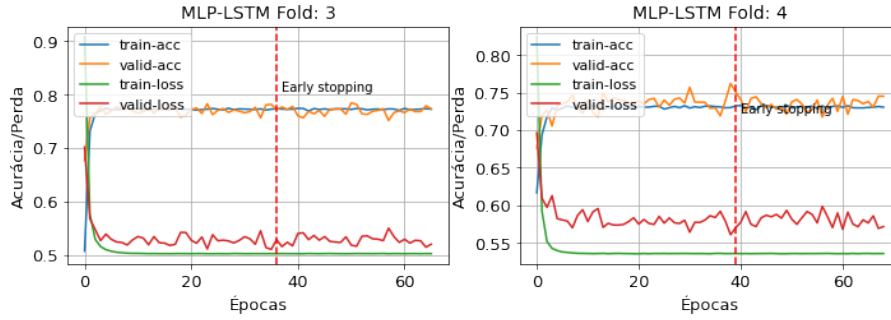


(c) Treinamento (Rede de fusão-10).

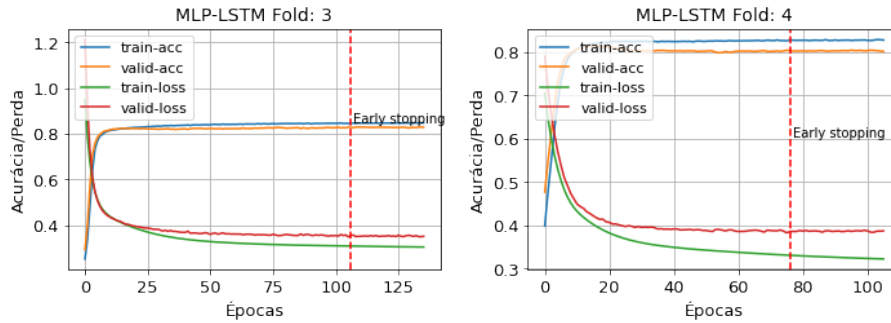
(d) Validação (Rede de fusão-10).

Figura A.27: Matriz de Confusão da fusão dos modelos MLP e LSTM (águas rasas).

curvas dos outros *folders* estão no Apêndice C.3.2). A curva de perda de validação no modelo sem a rede de fusão mostrou-se ruidosa em torno de 0,5, esse valor foi alcançado com poucas épocas. O modelo com a rede de fusão apresentou uma pequena lacuna entre a curva de treinamento e perda. Isso significa que, o conjunto de dados de treinamento não fornece informações suficientes para entender o problema, em relação ao conjunto de dados de validação usado para avaliá-lo.



(a) Modelo (Sem a Rede de fusão).



(b) Modelo (Rede de fusão-10).

Figura A.28: Curvas de treinamento e validação da fusão do modelo MLP e LSTM com e sem a rede de fusão (águas rasas).

A Tabela A.15 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com e sem a rede de fusão. O modelo com a rede de fusão tem uma melhora significativa nos valores médios de todos os índices. Isto posto, o modelo com 10 neurônios na camada oculta da rede de fusão é o mais adequado nesta fusão.

Tabela A.15: Resultados experimentais do conjunto de validação da fusão dos modelos MLP com LSTM (águas rasas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
--	76,35 ± 3,42	76,61 ± 1,79	66,64 ± 1,87	76,41 ± 3,35
10	80,17 ± 3,82	81,75 ± 1,84	70,78 ± 2,36	80,24 ± 3,70

A.2.4 Fusão CNN com LSTM

O melhor modelo LOFAR convolucional pré-treinado é modelo com duas camadas convolucionais de 8 filtros cada, uma camada *fully connected* com 20 neurônios e *kernel* 3. Já o melhor modelo temporal LSTM é o modelo com 50 células na camada oculta. A Tabela A.16 apresenta o índice SP do conjunto de validação pela variação do número de neurônios da rede de fusão. Os resultados são semelhantes ao considerarmos os valores médios do índice SP e a margem de erro. Como mencionado no modelo A.2.2, apesar do uso da rede de fusão e o aumento do número de neurônios não melhorar significativamente o índice SP, o uso da rede de fusão será considerado neste estudo. O gráfico de violino da Figura A.29 mostra os valores do índice SP pela variação de neurônios da rede de fusão. A maior parte dos valores estão concentrados acima de 75% e com *outliers* em torno de 65%.

Tabela A.16: Resultados experimentais do índice SP do conjunto de validação da fusão dos modelos (águas rasas).

Neurônios da Rede de fusão	SP(%)
–	80,76 ± 5,16
10	80,66 ± 5,75
30	81,52 ± 4,87
50	80,57 ± 4,90
100	80,41 ± 4,48

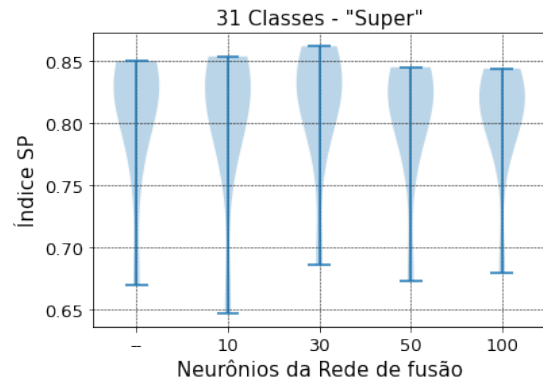


Figura A.29: Gráfico de Violino do desempenho do conjunto de validação da fusão dos modelos CNN com LSTM (águas rasas).

A Figura A.30 mostra uma matriz de histogramas da saída de um *fold* de treinamento da fusão dos modelos CNN e LSTM com 10 neurônios na rede de fusão e sem rede. Apesar dos modelos apresentarem SP semelhantes, o modelo com 10 neurônios na rede de fusão apresenta uma frequência de valores próximos de “1” para o Verdadeiros Positivos e próximos de “0” para os Verdadeiros Negativos, isso em relação ao modelo sem a rede de fusão. Esse comportamento foi observado em todos os *folders* (vide as Figuras C.114 a C.118).

A Figura A.31 apresenta a matriz de confusão dos conjuntos de treinamento e validação da fusão dos modelos CNN e LSTM com 10 neurônios na camada oculta da rede de fusão e sem rede. Os resultados dos conjuntos de treinamento e validação estão bem próximos e dentro da margem de erro. Avaliando o conjunto de validação, chama a atenção o valor de cerca de 23% para os Falsos Positivos da classe B.

A Figura A.32 apresenta as curvas de perda e acurácia dos conjuntos de treina-

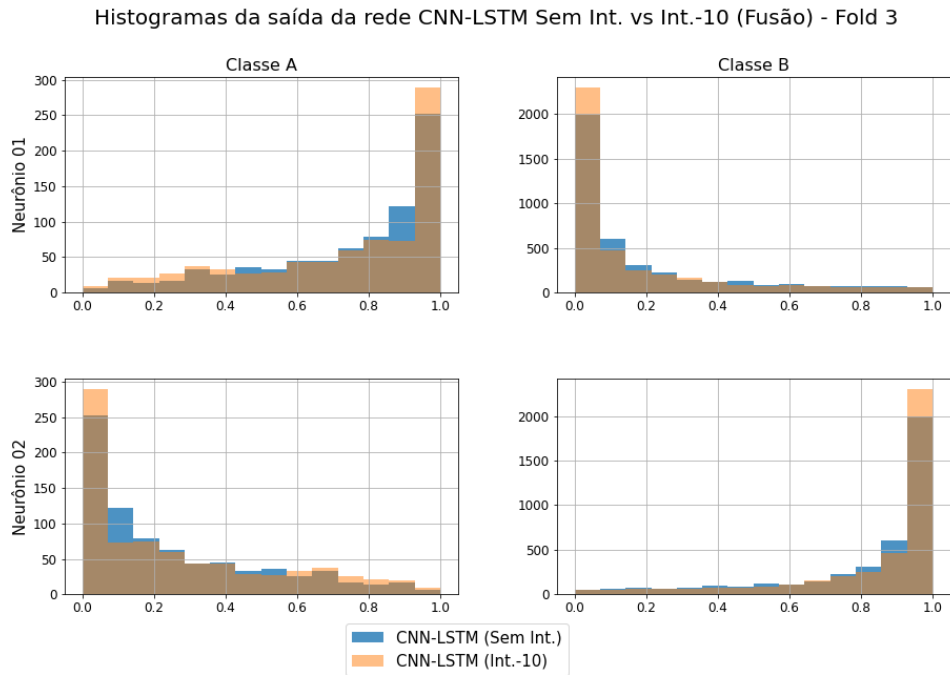


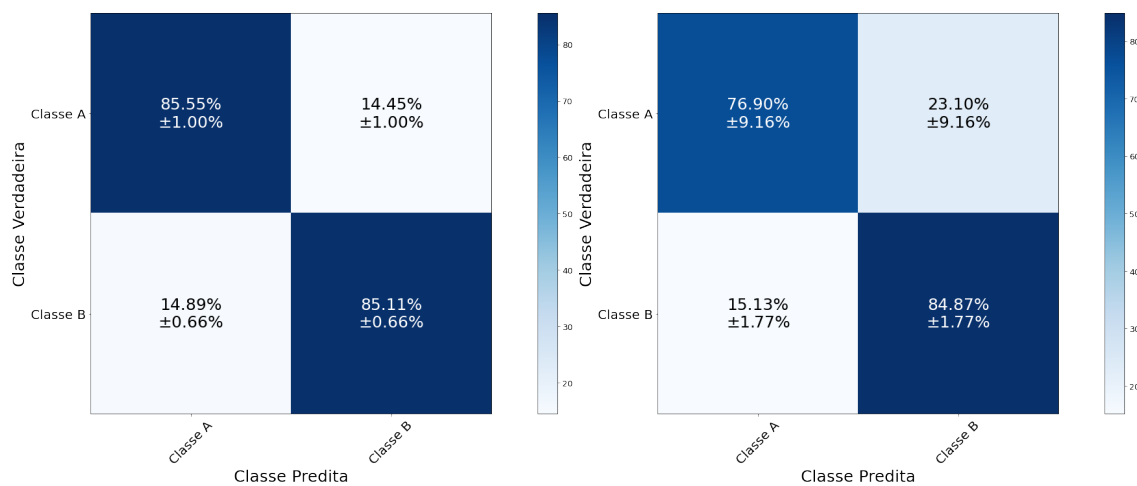
Figura A.30: Histogramas da saída da fusão dos modelos CNN com LSTM , *fold* 3 (águas rasas).

mento e validação de dois *folders* do *dataset* da fusão dos modelos CNN e LSTM com 10 neurônios na camada oculta da rede de fusão e a fusão sem a rede (as curvas dos outros *folders* estão no Apêndice C.3.2). O modelo sem a rede de fusão apresentou um bom ajuste, dado que a perda de treinamento e validação diminuiu até um ponto de estabilidade com uma lacuna mínima entre os dois valores finais. O modelo com a rede de fusão apresentou *overfitting* entre 60 e 100 épocas, esse ponto significa que o modelo aprendeu o conjunto de dados de treinamento muito bem, incluindo o ruído estatístico ou flutuações aleatórias no conjunto de dados de treinamento.

A Tabela A.17 apresenta o resultado do índice SP, Acurácia, Precisão e *Recall* dos modelos com e sem a rede de fusão. O modelo com a rede de fusão tem uma melhora pouco significativa nos valores médios, mas são bem semelhantes se considerarmos a margem de erro. Para esse estudo iremos utilizar o modelo com a rede de fusão.

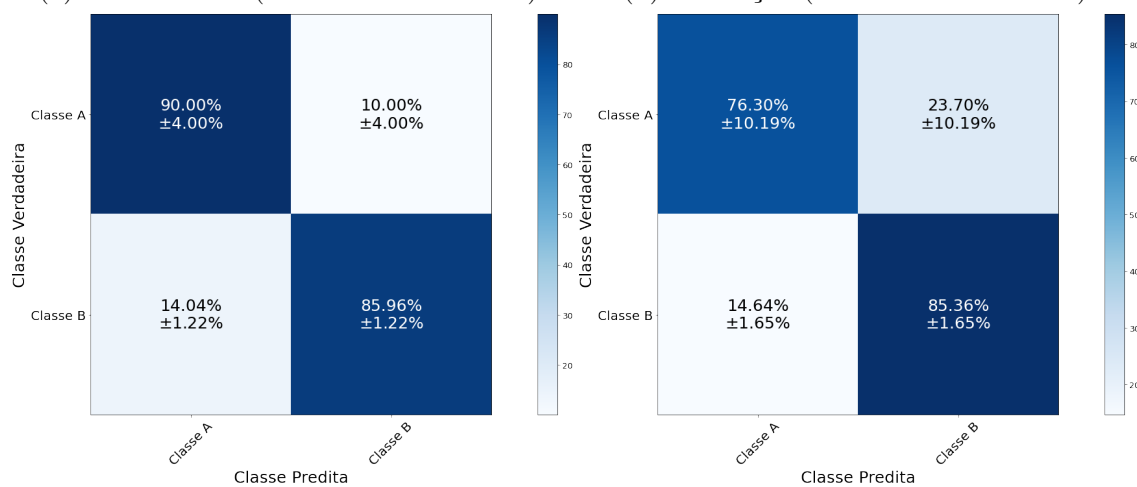
Tabela A.17: Resultados experimentais do conjunto de validação da fusão dos modelos CNN com LSTM (águas rasas)

Neurônios	Índice SP (%)	Acurácia (%)	Precisão (%)	Recall (%)
--	$80,76 \pm 5,16$	$83,56 \pm 2,41$	$72,36 \pm 3,35$	$80,88 \pm 4,93$
10	$80,66 \pm 5,75$	$83,88 \pm 2,49$	$72,60 \pm 3,69$	$80,82 \pm 5,43$



(a) Treinamento (Sem a Rede de fusão).

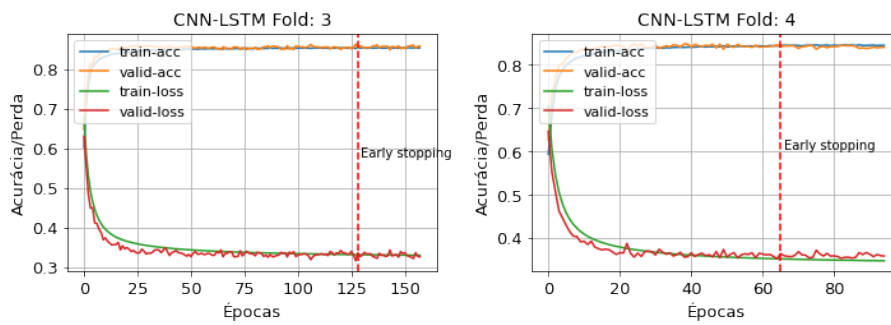
(b) Validação (Sem a Rede de fusão).



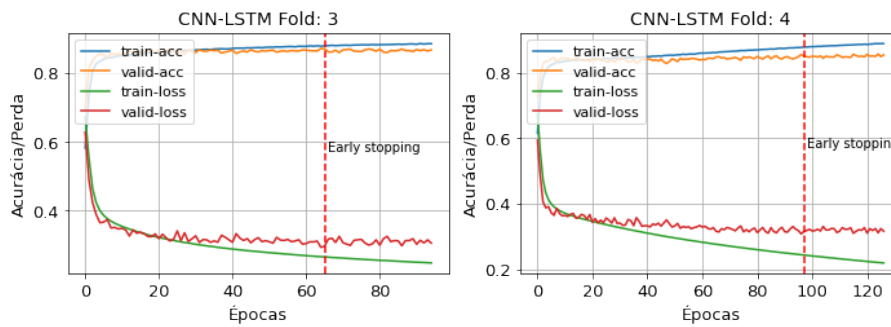
(c) Treinamento (Rede de fusão-10).

(d) Validação (Rede de fusão-10).

Figura A.31: Matriz de Confusão da fusão dos modelos CNN e LSTM (águas rasas).



(a) Modelo (Sem a Rede de fusão).



(b) Modelo (Rede de fusão-10).

Figura A.32: Curvas de treinamento e validação da fusão dos modelos CNN e LSTM com e sem a rede de fusão (águas rasas).

Apêndice B

Aprendizado de Máquina

Este apêndice tem por objetivo apresentar o conceito de Redes Neurais Artificiais, além dos três modelos de redes neurais (*MultiLayer Perceptron*, Convolucionais e Recorrentes) utilizados durante o desenvolvimento deste trabalho.

B.1 Redes Neurais Artificiais

A grande capacidade de processamento de informações do cérebro humano tem motivado pesquisas no sentido de encontrar modelos que reproduzam suas características computacionais, que são distintas das características do computador digital convencional. O cérebro humano é constituído basicamente de unidades estruturais elementares chamadas de neurônios que podem apresentar diversas entradas e uma saída. Esse neurônio pode está conectado com outros neurônios compondo assim uma rede neural [80]. Segundo HAYKIN [47], uma rede neural é um sistema paralelo distribuído e constituído de unidades básicas de processamento que têm a habilidade de armazenar conhecimento experimental e disponibilizá-lo para uso. As redes neurais artificiais (RNA) são uma família de técnicas de aprendizado de máquina baseadas em um modelo computacional inspirado na estrutura de uma rede de neurônios natural [81]. A grande vantagem no uso de RNA para a solução de problemas complexos é oriunda de algumas propriedades e capacidades úteis, tais como: Aprendizagem, Generalização, Não-Linearidade, Adaptabilidade, Tolerância a Falhas e Resposta a Evidências [80]. Dentre as RNA, as técnicas de aprendizado profundo se destacam em modelar abstrações de alto nível de dados[8]. Aprendizado profundo é o nome dado a uma classe de algoritmos de aprendizado de máquinas que utilizam várias camadas de processamento. A ideia por trás da aprendizagem profunda é descobrir múltiplos níveis de representação com a expectativa de que recursos de alto nível representem uma semântica mais abstrata dos dados [82]. Dentre esses algoritmos pode se destacar duas redes que serão abordadas neste trabalho, as redes Convolucionais e as Recorrentes, as quais serão detalhadas nas próximas

Subseções.

B.1.1 Rede Neural MultiLayer Perceptron

O *Perceptron*, idealizado por Rosenblatt (1958), é a forma mais simples de configuração de uma RNA. A Figura B.1 ilustra uma rede *Perceptron* constituída de n sinais de entrada, representações do problema a ser interpretado, e somente uma saída.

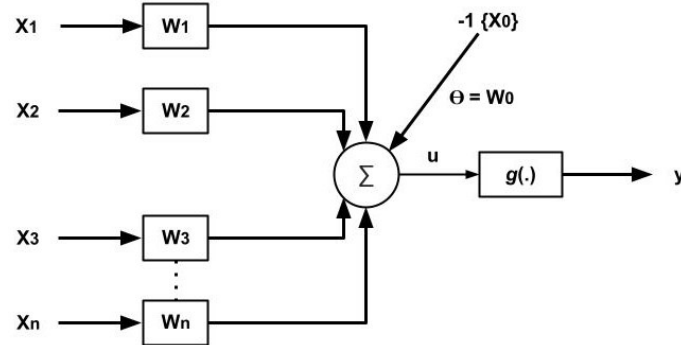


Figura B.1: Rede *Perceptron*.

Fonte: DA SILVA *et al.* [80]

Em DA SILVA *et al.* [80] temos seu princípio de funcionamento descrito: cada uma das entradas x_i , as quais representam informações sobre o comportamento do processo a ser mapeado, será inicialmente ponderada pelos pesos sinápticos x_i com o intuito de quantificar a importância de cada uma frente ao neurônio, cujo propósito será então mapear o comportamento da entrada/saída do referido processo. Em seguida o valor resultante da composição de todas as entradas já devidamente ponderadas por seus respectivos pesos, somado ainda a um limiar de ativação θ são passados como argumento de uma função de ativação, a qual retorna a saída y produzida pelo *Perceptron*. As duas equações do sistema (B.1) representa matematicamente o processamento interno realizado pelo perceptron:

$$\begin{cases} u = \sum_{i=1}^n w_i \cdot x_i - \theta \\ y = g(u) \end{cases} \quad (\text{B.1})$$

onde x_i são as entradas da rede, w_i é o peso (ponderação) associado à i -ésima entrada θ é o limiar de ativação, $g(\cdot)$ é a função de ativação e u é o potencial de ativação.

As redes *Perceptron* Multicamadas (*Multilayer Perceptron*, MLP) são redes que entre as camadas de entrada e saída possuem pelo menos uma camada intermediária de neurônios, amplamente conhecida como camada escondida. Sendo assim, elas

possuem no mínimo duas camadas de neurônios (escondida e de saída) [47]. A rede MLP é caracterizada pelas inúmeras aplicações em diversos tipos de problemas de diferentes áreas, sendo também considerada uma das arquiteturas mais versáteis quanto à aplicabilidade [80]. Seu treinamento é feito de forma supervisionada e consiste na minimização de uma função custo definida pelo usuário. Essa função avalia o erro entre a saída da rede e o alvo desejado. Na minimização deste erro é utilizado um algoritmo de retropropagação, o qual ajusta os pesos sinápticos e limiares de todos os neurônios da rede [8]. Na figura B.2 podemos observar uma rede MLP genérica com duas camadas escondidas e uma camada de saída com três neurônios.

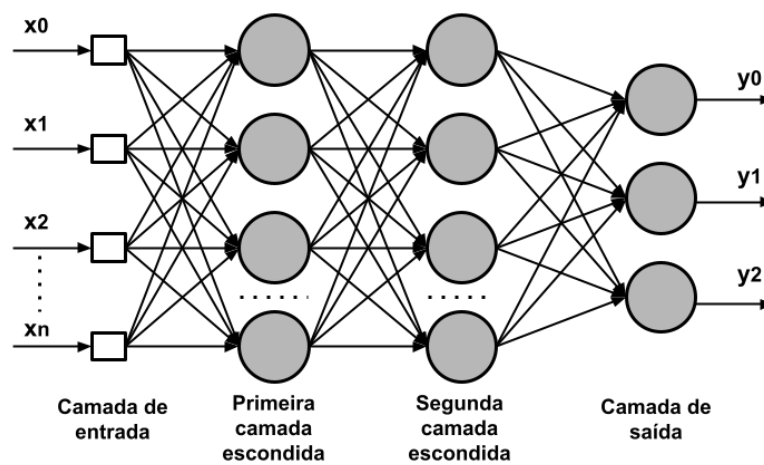


Figura B.2: Redes *Multilayer Perceptron*.

Fonte: DA SILVA *et al.* [80]

B.1.2 Rede Neural Convolucional

A Rede Neural Convolucional (*Convolutional Neural Network* - CNN) é um tipo especializado de rede neural para processamento de dados que possui uma topologia conhecida como grade. Ela tem sido aplicada com sucesso em diversos tipos de dados, tais como: de séries temporais que podem ser considerados uma grade 1D coletando amostras em intervalos de tempo regulares, e dados de imagem que podem ser considerados uma grade 2D de *pixels*[22]. O nome “rede neural convolucional” é dado devido ao fato da rede empregar uma operação matemática chamada de convolução, entre os dados da entrada por filtros (comumente chamados de *kernels*), ao invés de usar multiplicação matricial em pelo menos uma de suas camadas [23].

Na classificação de imagens, as entradas são usualmente matrizes bidimensionais com altura e largura (em que cada posição da matriz representa um *pixel* da imagem), podemos pensar também como entradas tridimensionais se levarmos em

consideração a profundidade da imagem que é determinada pela quantidade de canais de suas cores.

Uma CNN pode ser dividida em duas partes: extração de características e uma rede neural tradicional [22]. Em uma forma mais geral, matematicamente uma convolução é uma operação linear que, a partir de duas funções, gera uma terceira (chamada de *feature map*). Podemos entender esse processo como um *kernel* que transforma a imagem de entrada [83]. O *kernel* nada mais é do que uma matriz utilizada para uma operação de multiplicação de matrizes. A Figura B.3 exemplifica a operação do *kernel* sobre uma matriz de entrada. Nela é apresentada a convolução de uma matriz 5x5, na qual representa uma imagem de entrada e um filtro que cobre uma área de 3x3 da imagem. A operação entre as matrizes é aplicada diversas vezes em diferentes regiões da imagem. A cada aplicação, a região é alterada por um parâmetro chamada de chamados de *stride*. Normalmente o *stride* possui o valor 1, o que significa que a transformação será aplicada em todos os *pixels* da imagem e por cada camada para imagens com vários canais, formando no final um *feature map* (mapas de parâmetros) ou *activation map* de 3x3 [83]. O *feature map* apresentado na imagem apresenta apenas o resultado de uma convolução sem salto.

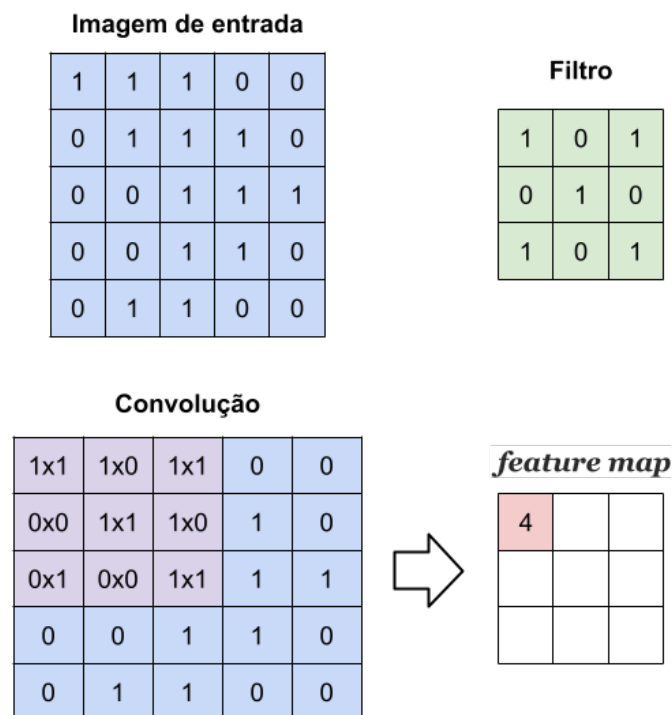


Figura B.3: Operação de convolução entre a entrada e o filtro.

Podemos observar que a matriz resultante da Figura B.3 possui uma dimensionalidade menor que a imagem de origem. Um grande número de convoluções podem impactar no decisão da CNN se o tamanho da imagem for muito reduzido. Uma maneira de contornar esse problema é utilizar o conceito de *Padding*. *Padding* nada

mais é do que um processo em que alguns *pixels* são adicionados ao redor da imagem antes da operação de convolução, de forma a manter a dimensionalidade da imagem resultante durante a operação [84]. Este processo de adição pode ser observado na Figura B.4.

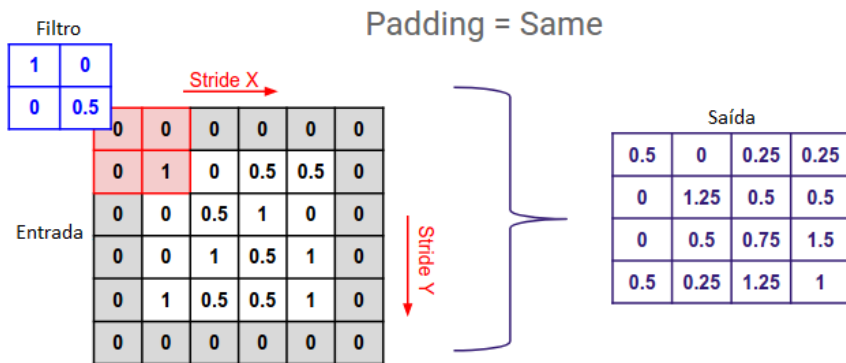


Figura B.4: Processo de *Padding* em uma imagem.

Fonte: AYESHMANTHA PERERA [84].

Existe também um processo de redução de dimensionalidade chamado de *Pooling*, no qual consiste em uma transformação com redução do tamanho da imagem. A redução tem o intuito de diminuir sua variância a pequenas alterações e também de reduzir a quantidade de parâmetros treinados pela rede. Além disso, é preciso escolher como será feita a sumarização. Essa sumarização serve para diminuir a quantidade de pesos a serem aprendidos e também para evitar *overfitting*. As três técnicas existentes são *MaxPooling*, *SumPooling* e *AveragePooling*. Todas têm o mesmo princípio, porém diferem na forma de calcular o valor final. Dentre elas, a mais utilizada é a *MaxPooling*. Nela é feito um deslizamento considerando um parâmetro de *stride*, retirando o maior elemento de determinada região da matriz. Outro elemento importante na CNN é a camada *FLatten*. Essa camada redimensiona a matriz resultante das camadas anteriores de convoluções e *poolings* para um *array* linear de uma única dimensão. Esta etapa é um preparo para se entrar na camada principal de uma rede neural totalmente conectada, camada de *fully connected*. Por fim, é implementada a rede neural totalmente conectada ou *fully connected*, como mencionado anteriormente, na qual também possui as informações da saída da rede [22, 83]. A Figura B.5 apresenta a topologia completa da rede CNN com a representação dos elementos citados anteriormente.

B.1.3 Rede Neural Recorrente

As Redes Neurais Recorrentes (*Recurrente Neural Networks* - RNNs) são classes de redes neurais artificiais bastante úteis para o processamento de dados sequenciais,

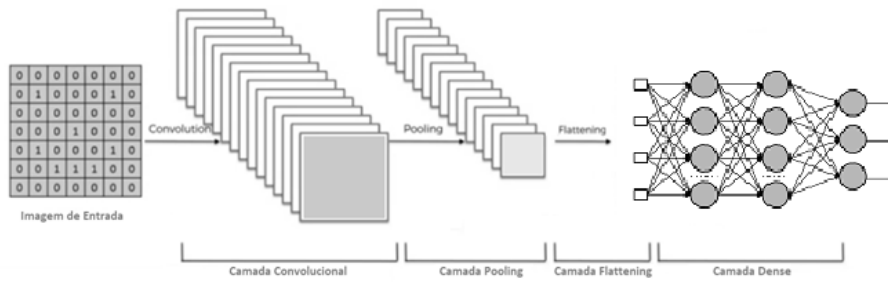


Figura B.5: Topologia CNN.

Fonte: <https://www.medium.com>

tais como, séries temporais ou linguagem natural. São modelos bastantes populares que vem mostrando bastante potencial em diversas áreas [85].

Na rede neural padrão, a predição é realizada considerando cada entrada de forma independente. Esse processo funciona em muitos casos de classificação. Entretanto, quando a ideia é prever dados que tenham uma tendência relacionada ao tempo, elas não são tão eficientes. Para esse problema, surgiu as RNNs. Elas são uma família de redes neurais especializadas para processar dados sequenciais $x^{(1)}, \dots, x^{(\tau)}$. Elas podem escalar sequências mais longas, o que seria prático para redes sem especialização baseadas em sequência. As RNNs também podem processar sequências de comprimentos variáveis [22]. Na figura B.6 podemos observar uma breve comparação entre a arquitetura de uma rede *feedforward* e uma rede recorrente. Na recorrente, podemos observar que ela possui conexões ponderadas dentro de uma mesma camada, mantendo assim seu estado interno através dos nós de contexto, diferentemente de uma rede *feedforward*. Esse comportamento será descrito a seguir.

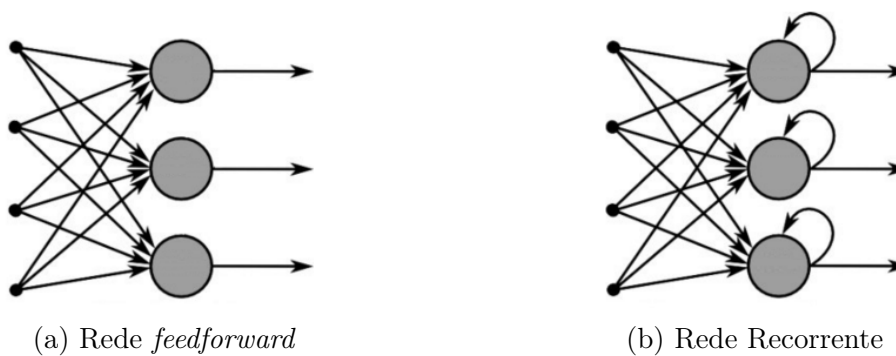


Figura B.6: Comparação da arquitetura de uma rede *feedforward* e uma recorrente.

Fonte: <https://www.deeplearningbook.com.br>

Para passar de redes multicamadas para redes recorrentes é preciso aproveitar a ideia de compartilhamento de parâmetros em diferentes partes de um modelo. Esse compartilhamento torna possível estender e aplicar o modelo a diferentes formas de

tamanho. Se tivéssemos parâmetros separados para cada valor do índice de tempo, não seria possível generalizar o modelo para sequências não vistas durante o treinamento e nem compartilhar força estatística. No compartilhamento de parâmetros, cada membro da saída é uma função dos membros anteriores da saída. Sendo que cada membro da saída é produzido usando a mesma regra de atualização aplicada às saídas anteriores. Essa formulação recorrente resulta no compartilhamento de parâmetros por meio de um grafo computacional muito profundo [22]. Em suma, podemos pensar nas RNNs como redes com “memória” que mantém informações sobre o que foi calculado até o momento. A Figura B.7 apresenta graficamente o funcionamento de uma RNNs e a ideia de compartilhamento de parâmetros. GUILHERME DUARTE [85] descreve de forma clara e objetiva o processo de funcionamento da rede: A camada da rede neural recebe uma entrada x e gera uma saída o , onde U é o peso na camada de entrada e W o peso na camada de saída. A célula transmite a informação para as outras camadas e para si mesma por meio de um *loop* com peso “ V ”. Cada célula possui um estado, reapresentado por “ h ”, que varia ao longo do tempo, ou seja, uma célula no estado $h(t - 1)$ recebe uma entrada $x(t - 1)$, e além de transmitir a saída $o(t - 1)$ para a camada seguinte, ela utiliza a entrada para atualizar seu estado para $h(t)$. Portanto, quando a célula recebe a entrada $x(t)$, ela “se lembra” da entrada $x(t - 1)$. Esse processo continua sucessivamente. Devemos atentar para dois pontos importantes:

1. $h(t)$ captura as informações sobre o que aconteceu em todas as etapas de tempo anterior, sendo assim podemos pensar no estado $h(t)$ como a memória da rede..
2. A saída na etapa $o(t)$ é calculada com base apenas na memória no tempo t .

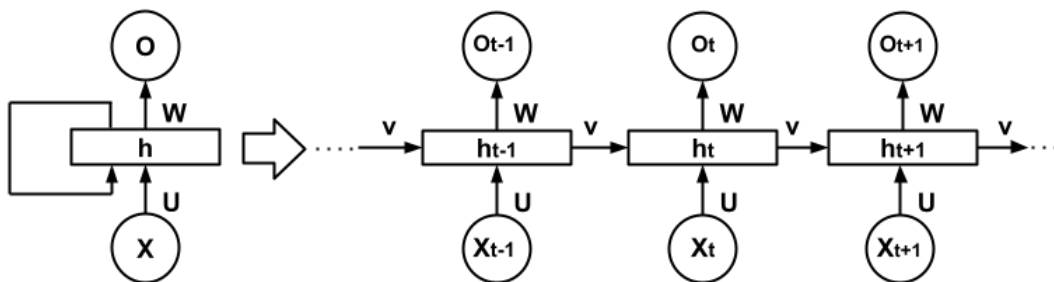


Figura B.7: Rede Neural Recorrente e o desdobramento de seu processo temporal.

Fonte: GUILHERME DUARTE [85]

Um dos desafios matemáticos que temos neste tipo de rede é aprender as dependências de longo prazo. O problema é que gradientes propagados por muitos estágios tendem a desaparecer (na maioria das vezes) ou explodir (raramente, mas

com prejuízo para a otimização). Mesmo se assumirmos que os parâmetros são tais que a rede recorrente é estável (pode armazenar memórias, com gradientes não explodindo), a dificuldade com dependências de longo prazo surge dos pesos exponencialmente menores dadas às interações de longo prazo (envolvendo a multiplicação de muitos jacobianos) em comparação com as de curto prazo [22].

Redes Long Short-Term Memory

Um tipo de RNNs que pode ser utilizada para contornar o problema descrito na Seção B.1.3 são as redes de Memória de Curto e Longo Prazo, conhecidas como (*Long Short-Term Memory* - LSTM), pois elas são capazes de aprender conexões de longo prazo, introduzindo *auto-loops* para produzir caminhos onde o gradiente pode fluir por longas durações [22]. Sua arquitetura pode ser vista na Figura B.8.

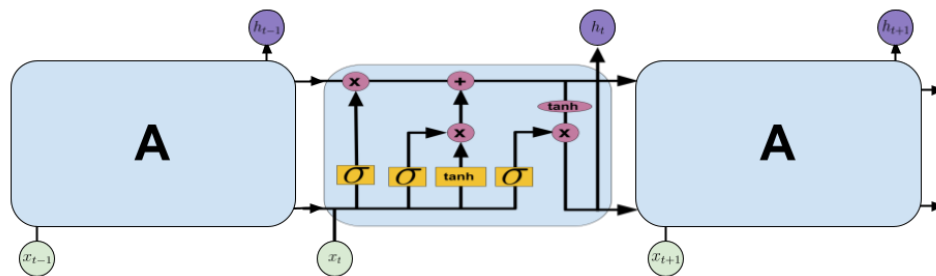


Figura B.8: Módulo de repetição em um LSTM.

Fonte: FERNANDO MATSUMOTO [86]

O principal objetivo é entender como a rede decide quais partes da memória de curto prazo devem ser lembradas e como a memória de longo prazo deve afetar o entendimento atual das informações de entrada. Para isso é importante entender o estado da célula LSTM que funciona como uma correia transportadora (linha horizontal na parte superior da Figura B.8), na qual ocorrem algumas operações lineares e as informações fluem sem alterações e esse estado é chamado também de memória de longo prazo. A linha inferior da célula corresponde a memória de curto prazo que contém a entrada e a saída no instante anterior [87]. Três estruturas chamadas *gates* (portas) realizam o controle das informações que fluem para o estado da célula. Na Figura B.9 podemos observar que elas são basicamente compostas de uma função sigmóide aplicada a entrada b (controla qual a porcentagem com que cada valor deve passar pela porta) e o resultado é multiplicado pelo sinal a , o qual representa o estado da célula mencionado anteriormente [85].

Na célula LSTM (Figura B.8) temos três portas, que são respectivamente [87]:

1. *Forget Gate*: As informações que não são mais úteis no estado da célula são removidas com o *forget gate*;

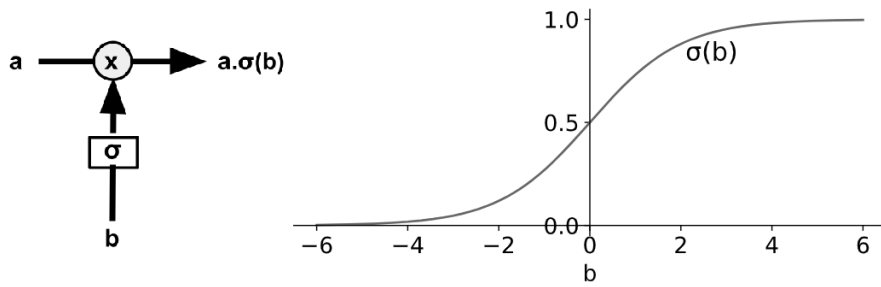


Figura B.9: Representação de um *gate* da célula LSTM e ao lado o gráfico de uma função sigmóide.

Fonte: FERNANDO MATSUMOTO [86]

2. *Input Gate*: A adição de informações úteis ao estado da célula é feita pelo *input gate*; e
3. *Output Gate*: A tarefa de extrair informações úteis do estado da célula atual para serem apresentadas como uma saída é feita pelo *output gate*.

Após compreender os elementos da estrutura podemos em fim entender o seu funcionamento. Primeiro a *forget gate* determina quais partes do estado da célula são importantes e quais devem ser esquecidas, em seguida, por meio do *input gate* são determinadas quais novas informações da memória de curto prazo devem ser colocadas no estado da célula. Por fim, o estado da célula é utilizado para se calcular a saída e o estado da rede. No *output gate* a memória de curto prazo também é utilizada para determinar a relevância das partes da saída [86].

Apêndice C

Resultados Complementares

Este apêndice apresenta resultados complementares aos discutidos na Seção 5.

C.1 Águas Profundas

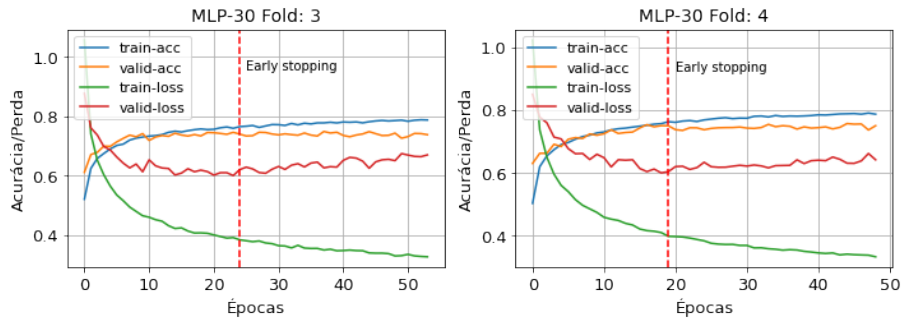
C.1.1 Rede Neural MLP

A Figura C.1 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folders* do *dataset*. Essas curvas mostram mudanças no desempenho da aprendizagem ao longo do tempo em termos de experiência. As curvas dos outros *folders* estão no Apêndice C.3.1. Nestes *folders*, o modelo apresenta a menor perda do conjunto de validação em torno de 20 a 30 épocas. A partir desse valor o modelo sofre um sobre-ajuste (do inglês: *overfitting*), ou seja, o modelo estatístico ajusta-se muito bem ao conjunto de treinamento se tornando ineficaz na previsão de novos dados. A grande lacuna entre as curvas de treinamento e validação significa que o conjunto de dados de treinamento não fornece informações suficientes para entender o problema, em relação ao conjunto de validação usado para avaliá-lo.

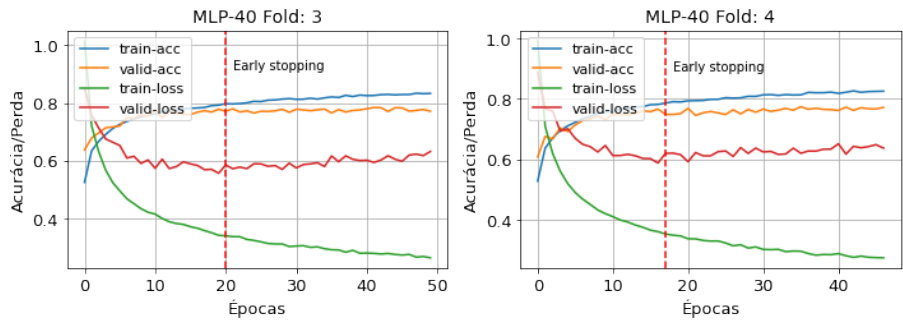
A Figura C.2 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folders* do *dataset* do modelo temporal com 30 neurônios. A lacuna entre as curvas de treinamento e validação não é pequena, isso representa que o conjunto de dados de treinamento não fornece boas informações para o modelo compreender o problema, em relação ao conjunto de validação usado para avaliá-lo.

C.1.2 Rede Neural CNN

A Figura C.3 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folders* do *dataset* dos modelos convolucionais abordados (as curvas dos outros *folders* estão no Apêndice C.3.1). A lacuna entre as curvas de treinamento e validação é menor que a apresentada no modelo MLP, isso significa que o conjunto de

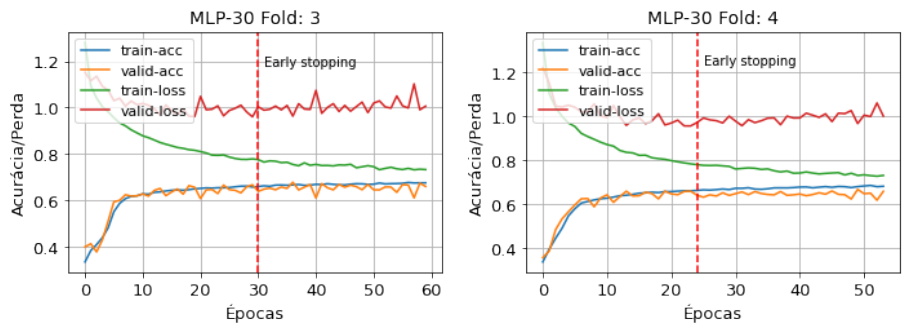


(a) Modelo com 30 neurônios.



(b) Modelo com 40 neurônios.

Figura C.1: Curvas de treinamento e validação dos modelos LOFAR MLP, *folds* 3 e 4 (águas profundas).



(a) Modelo com 30 neurônios.

Figura C.2: Curvas de treinamento e validação do modelo temporal MLP com 30 neurônios, *folds* 3 e 4 (águas profundas).

treinamento teve uma boa representação em relação ao conjunto de validação. Como mencionado anteriormente, o modelo com 20 neurônios na camada *fully connected* teve uma melhor compreensão das informações extraídas das camadas convolucionais em relação ao modelo com 10 neurônios, uma vez que a curva de acurácia do conjunto de validação está acima da curva de treinamento.

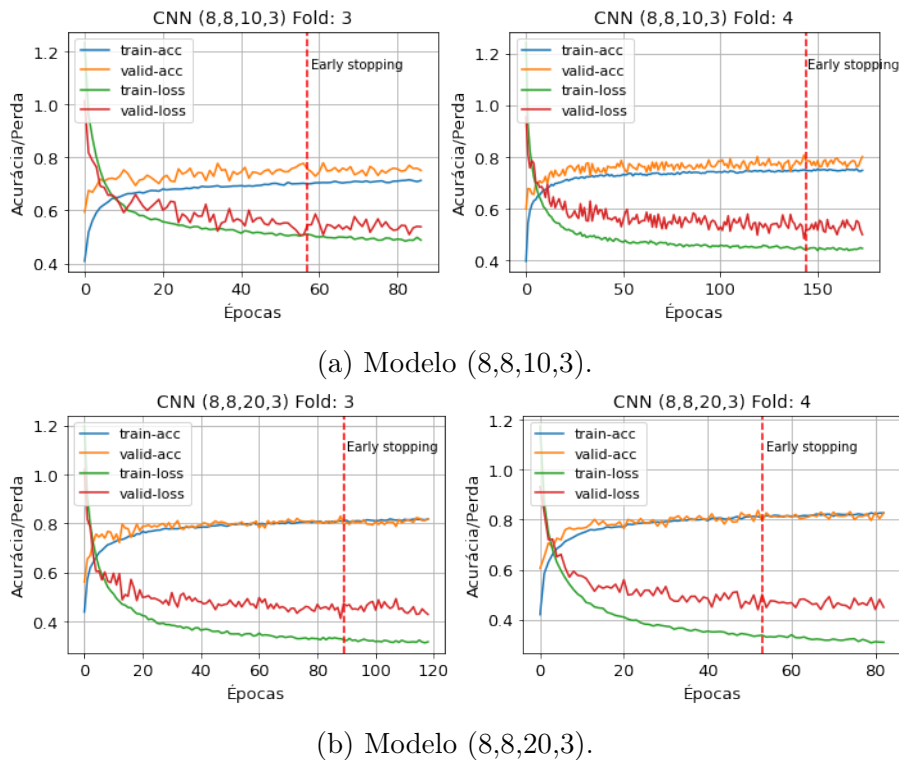


Figura C.3: Curvas de treinamento e validação dos modelos LOFAR (8,8,10,3) e (8,8,20,3), *folds* 3 e 4 (águas profundas).

A Figura C.4 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folds* do *dataset* dos modelos convolucionais abordados (as curvas dos outros *folds* estão no Apêndice C.3.1). A lacuna entre as curvas de treinamento e validação é menor do que a apresentada no modelo MLP, isso significa que, o conjunto de treinamento teve uma boa representação em relação ao conjunto de validação. Essa lacuna é menor no modelo com 10 neurônios. O desempenho do treinamento foi superior no modelo com 20 neurônios e a curva de validação é um pouco menos ruidosa.

A Tabela C.1 apresenta o resultado do índice SP do conjunto de validação dos modelos CNN construídos pela busca paramétrica.

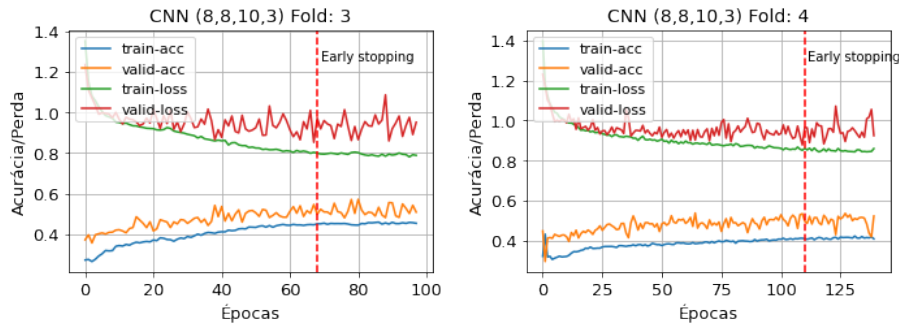
A Tabela C.2 apresenta a quantidade de parâmetros dos modelos CNN construídos pela busca paramétrica.

Tabela C.1: Resultados experimentais do índice SP para o modelo CNN (conjunto de validação - águas profundas).

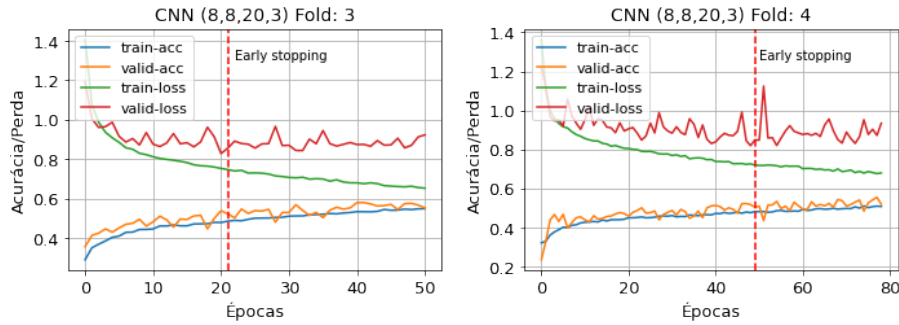
Modelos	Índice SP (%)			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
(16,16,10,3)	78,93 ± 3,47	83,73 ± 3,49	63,65 ± 2,05	63,55 ± 2,60
(16,16,10,6)	78,60 ± 3,18	83,52 ± 3,83	65,67 ± 2,48	63,57 ± 2,58
(16,16,20,3)	79,88 ± 2,86	84,16 ± 4,20	66,18 ± 1,59	61,90 ± 3,63
(16,16,20,6)	80,68 ± 3,14	84,24 ± 4,39	68,22 ± 2,60	66,32 ± 2,04
(16,16,30,3)	81,15 ± 3,71	84,83 ± 4,33	67,95 ± 1,76	65,25 ± 2,18
(16,16,30,6)	80,91 ± 3,26	84,52 ± 3,59	69,42 ± 2,27	67,54 ± 1,84
(16,8,10,3)	79,56 ± 3,48	84,28 ± 3,90	61,59 ± 2,13	62,46 ± 3,25
(16,8,10,6)	79,00 ± 2,67	83,70 ± 3,69	64,56 ± 2,67	62,56 ± 2,22
(16,8,20,3)	80,73 ± 3,24	84,36 ± 4,05	65,20 ± 1,59	62,63 ± 3,03
(16,8,20,6)	80,97 ± 3,05	84,49 ± 4,09	66,55 ± 2,35	65,17 ± 2,81
(16,8,30,3)	81,97 ± 3,15	85,36 ± 4,67	66,82 ± 2,34	63,29 ± 2,39
(16,8,30,6)	81,01 ± 2,75	84,88 ± 3,91	67,46 ± 2,63	65,52 ± 1,88
(16,-,10,3)	75,60 ± 2,65	80,37 ± 3,76	58,40 ± 2,18	56,30 ± 3,42
(16,-,10,6)	75,09 ± 2,89	80,04 ± 3,68	60,41 ± 4,04	58,29 ± 1,84
(16,-,20,3)	77,29 ± 2,94	81,56 ± 4,05	63,40 ± 2,24	61,37 ± 1,07
(16,-,20,6)	77,55 ± 3,34	80,56 ± 3,69	65,65 ± 1,57	63,43 ± 2,48
(16,-,30,3)	78,08 ± 2,98	82,12 ± 4,82	65,41 ± 1,82	64,21 ± 1,91
(16,-,30,6)	78,12 ± 3,19	81,93 ± 3,43	66,41 ± 1,79	64,16 ± 2,14
(8,8,10,3)	79,17 ± 3,50	83,67 ± 4,11	62,89 ± 1,43	61,14 ± 1,96
(8,8,10,6)	79,25 ± 2,34	82,82 ± 4,02	64,53 ± 3,11	62,33 ± 2,37
(8,8,20,3)	80,86 ± 2,90	84,03 ± 4,30	64,70 ± 1,68	62,94 ± 2,25
(8,8,20,6)	80,39 ± 3,18	84,36 ± 3,92	66,15 ± 2,06	64,96 ± 2,08
(8,8,30,3)	81,16 ± 2,59	84,81 ± 4,08	66,34 ± 2,60	64,50 ± 2,08
(8,8,30,6)	80,62 ± 3,83	84,86 ± 4,11	68,32 ± 1,79	65,90 ± 1,49
(8,-,10,3)	76,04 ± 2,48	79,40 ± 3,72	58,85 ± 3,76	58,18 ± 1,76
(8,-,10,6)	75,71 ± 2,91	79,56 ± 3,96	58,63 ± 1,80	57,95 ± 2,83
(8,-,20,3)	77,17 ± 2,59	81,69 ± 3,75	63,52 ± 1,63	62,30 ± 1,88
(8,-,20,6)	77,18 ± 3,10	81,77 ± 2,94	64,91 ± 1,71	64,16 ± 2,35
(8,-,30,3)	78,34 ± 3,11	82,29 ± 4,29	64,86 ± 1,75	62,94 ± 1,74
(8,-,30,6)	77,68 ± 3,31	81,31 ± 4,66	66,10 ± 2,06	64,35 ± 1,59

Tabela C.2: Quantidade de parâmetros para cada modelo CNN

Modelos	Parâmetros			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
(16,16,10,3)	32.902	64.902	41.862	82.822
(16,16,10,6)	33.718	66.718	42.678	83.638
(16,16,20,3)	64.952	128.952	82.872	164.792
(16,16,20,6)	65.768	130.768	83.688	165.608
(16,16,30,3)	97.002	194.002	123.882	246.762
(16,16,30,6)	97.818	194.818	124.698	247.578
(16,8,10,3)	16.510	32.510	20.990	41.470
(16,8,10,6)	16.942	32.942	21.422	41.902
(16,8,20,3)	32.560	64.560	41.520	82.480
(16,8,20,6)	32.992	64.992	41.952	82.912
(16,8,30,3)	48.610	96.610	62.050	123.490
(16,8,30,6)	49.042	98.042	62.482	123.922
(16,-,10,3)	32.118	64.118	82.038	163.958
(16,-,10,6)	32.166	64.166	82.086	164.006
(16,-,20,3)	64.168	128.168	164.008	327.848
(16,-,20,6)	64.216	128.216	164.056	327.896
(16,-,30,3)	96.218	192.218	245.978	491.738
(16,-,30,6)	96.266	192.266	246.026	491.786
(8,8,10,3)	16.286	32.286	20.766	41.246
(8,8,10,6)	16.502	32.502	20.982	41.462
(8,8,20,3)	32.336	64.336	41.296	82.256
(8,8,20,6)	32.552	64.552	41.512	82.472
(8,8,30,3)	48.386	96.386	61.826	123.266
(8,8,30,6)	48.602	96.602	62.042	123.482
(8,-,10,3)	16.086	32.086	41.046	82.006
(8,-,10,6)	16.110	32.110	41.070	82.030
(8,-,20,3)	32.136	64.136	82.056	163.976
(8,-,20,6)	32.160	64.160	82.080	164.000
(8,-,30,3)	48.186	96.186	123.066	245.946
(8,-,30,6)	48.210	96.210	123.090	245.970



(a) Modelo (8,8,10,3).

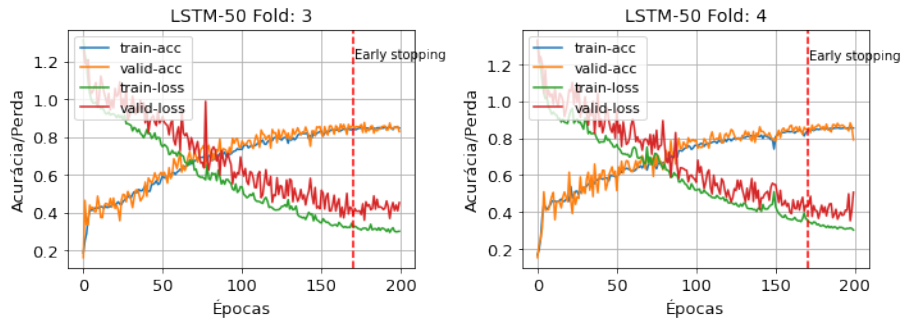


(b) Modelo (8,8,20,3).

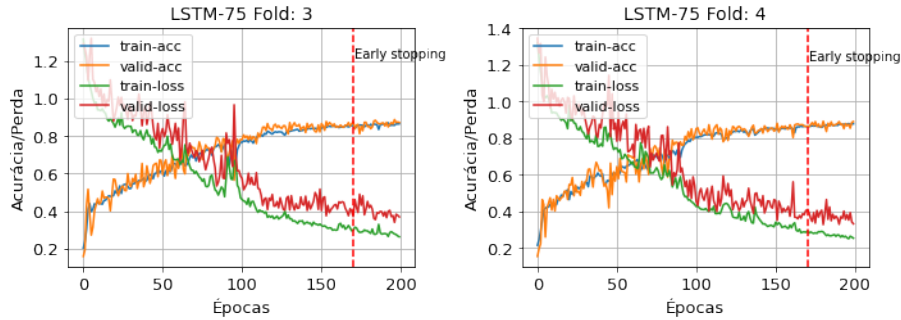
Figura C.4: Curvas de treinamento e validação dos modelos temporais (8,8,10,3) e (8,8,20,3), *folders* 3 e 4 (águas profundas).

C.1.3 Rede Neural LSTM

A Figura C.5 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folders* do *dataset* (as curvas dos outros *folders* estão no Apêndice C.3.1). Nestes *folders*, o modelo apresentou uma menor perda no conjunto de validação em torno de 165 épocas. Portanto, o modelo precisou de muitas amostras para entender o comportamento dos sinais. A curva de perda do treinamento e validação apresentou micro-variações, mas o algoritmo não interrompeu o processo de treinamento. As curvas apresentam uma pequena lacuna entre as perdas de treinamento e validação e apesar de não observarmos um ponto de estabilidade, pode-se considerar que os modelos tiveram um bom ajuste.



(a) Modelo com 50 células.



(b) Modelo com 75 células.

Figura C.5: Curvas de treinamento e validação dos modelos temporais LSTM, *folds* 3 e 4 (águas profundas).

C.2 Águas Rasas

C.2.1 Rede Neural MLP

A Figura C.6 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação do modelo LOFAR de dois *folds* do *dataset*. As curvas dos outros *folds* estão no Apêndice C.3.2. O *overfitting* ocorreu em torno de 30 épocas. O modelo apresentou uma lacuna grande entre a curva de perda e validação, o que indica que o conjunto de treinamento não fornece informações suficientes para entender o problema, em relação ao conjunto de dados de validação usado para avaliá-lo.

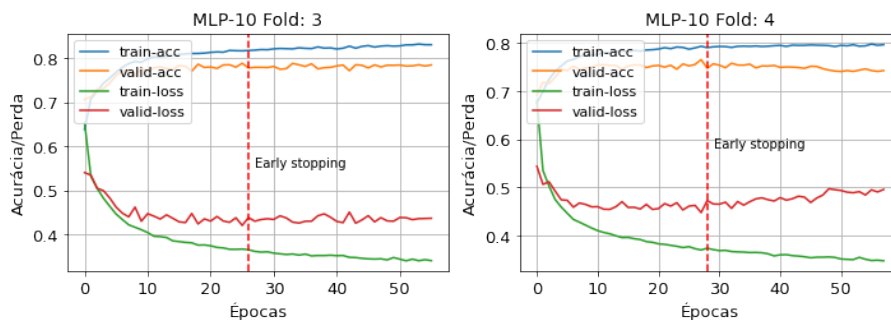


Figura C.6: Curvas de treinamento e validação do modelo LOFAR MLP com 10 neurônios, *folds* 3 e 4 (águas rasas).

A Figura C.7 apresenta as curvas de perda e acurária dos conjuntos de treinamento e validação de dois *folders* do *dataset* dos modelos temporais com 10 e 20 neurônios. A curva de validação é ruidosa para ambos os parâmetros, isso ocorre devido ao conjunto de dados de validação ter poucos exemplos em comparação com o conjunto de dados de treinamento.

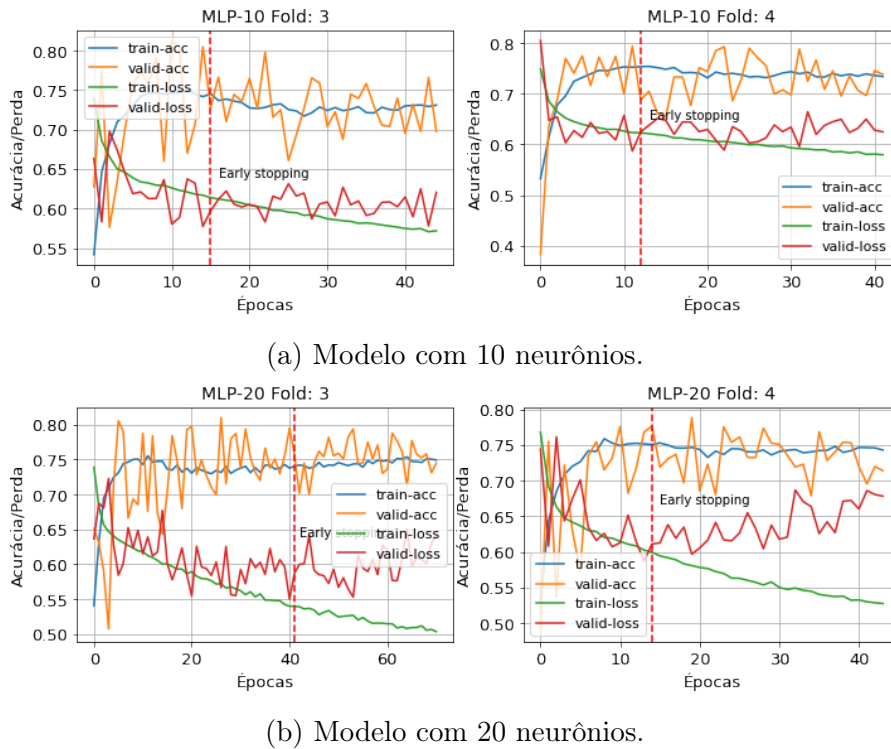
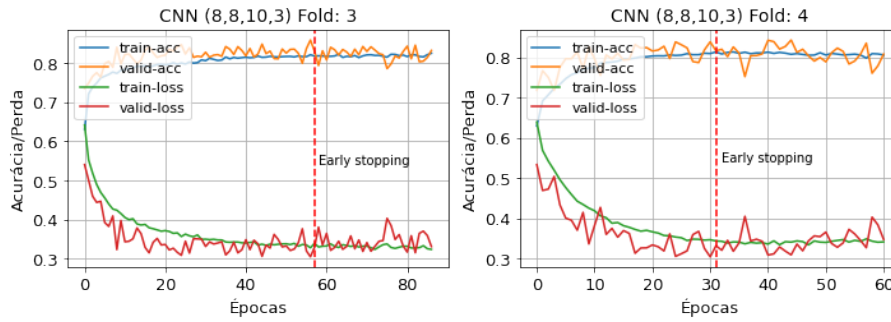


Figura C.7: Curvas de treinamento e validação dos modelos temporais MLP com 10 e 20 neurônios, *folders* 3 e 4 (águas rasas).

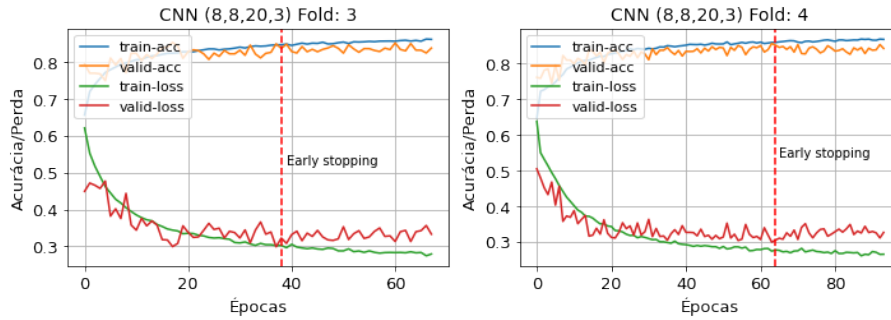
C.2.2 Rede Neural CNN

A Figura C.8 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folders* do *dataset* dos modelos convolucionais abordados (as curvas dos outros *folders* estão no Apêndice C.3.2). As curvas de validação na maioria dos *folders* mostram um conjunto de validação não representativo. Dado que a curva de perda de validação fica oscilando em torno da perda do conjunto de treinamento de forma ruidosa. Significa que, o conjunto de dados de validação tem poucos exemplos em comparação com o conjunto de dados de treinamento.

A Figura C.9 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *folders* do *dataset* dos modelos convolucionais abordados (as curvas dos outros *folders* estão no Apêndice C.3.2). Assim como mencionado no modelo LOFAR, as curvas de validação mostram um conjunto de validação não representativo. Uma vez que a curva de perda de validação fica oscilando em torno



(a) Modelo (8,8,10,3).



(b) Modelo (8,8,20,3).

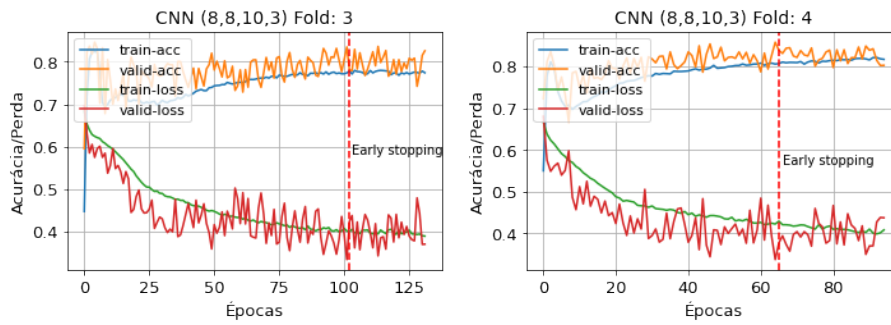
Figura C.8: Curvas de treinamento e validação dos modelos LOFAR (8,8,10,3) e (8,8,20,3), *fold*s 3 e 4 (águas rasas).

da perda do conjunto de treinamento de forma ruidosa.

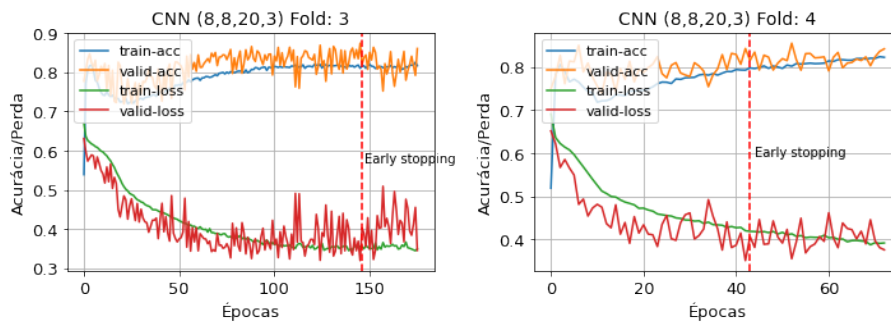
A Tabela C.3 apresenta o resultado do índice SP do conjunto de validação dos modelos CNN construídos pela busca paramétrica.

C.2.3 Rede Neural LSTM

A Figura C.10 apresenta as curvas de perda e acurácia dos conjuntos de treinamento e validação de dois *fold*s do *dataset* (as curvas dos outros *fold*s estão no Apêndice C.3.2). Nestes *fold*s, podemos considerar que o modelo apresentou um bom ajuste, mesmo que 200 épocas não tenham sido suficientes para encontrar o ponto de estabilidade das curvas.



(a) Modelo (8,8,10,3).



(b) Modelo (8,8,20,3).

Figura C.9: Curvas de treinamento e validação dos modelos temporais (8,8,10,3) e (8,8,20,3), *folders* 3 e 4 (águas rasas).

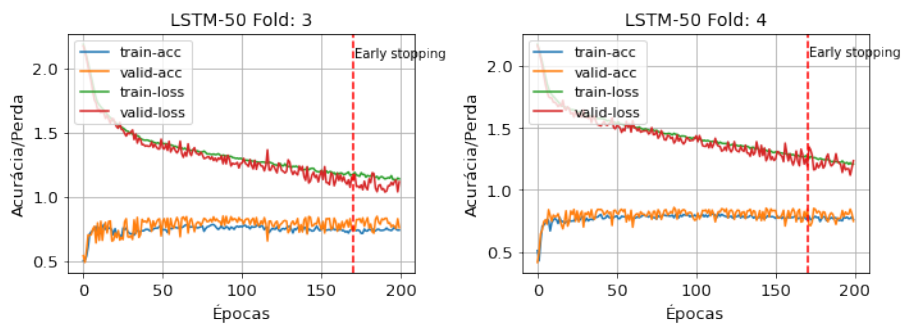


Figura C.10: Curvas de treinamento e validação do modelo temporal LSTM, *folders* 3 e 4 (águas rasas).

Tabela C.3: Resultados experimentais do índice SP para o modelo CNN (conjunto de validação - águas rasas).

Modelos	Índice SP (%)			
	LOFAR-1024	LOFAR-2048	tempo-1024	tempo-2048
(16,16,10,3)	78,51 ± 3,50	79,64 ± 4,82	73,25 ± 9,44	70,88 ± 10,41
(16,16,10,6)	78,90 ± 5,07	80,84 ± 3,16	72,26 ± 16,80	74,30 ± 7,77
(16,16,20,3)	78,50 ± 3,51	81,02 ± 4,94	73,88 ± 9,73	71,48 ± 12,68
(16,16,20,6)	78,66 ± 3,65	80,06 ± 6,10	76,30 ± 7,92	72,26 ± 14,39
(16,16,30,3)	78,56 ± 4,21	81,33 ± 3,17	74,67 ± 7,75	71,70 ± 10,68
(16,16,30,6)	79,21 ± 3,84	81,21 ± 5,15	74,95 ± 8,15	73,47 ± 10,34
(16,8,10,3)	78,84 ± 4,10	80,97 ± 4,60	75,61 ± 7,32	73,28 ± 10,31
(16,8,10,6)	78,60 ± 3,22	79,10 ± 4,85	76,52 ± 7,55	74,09 ± 9,65
(16,8,20,3)	77,70 ± 3,93	81,74 ± 3,89	72,33 ± 13,93	71,48 ± 15,50
(16,8,20,6)	77,88 ± 3,44	80,86 ± 5,48	75,03 ± 10,80	73,70 ± 13,02
(16,8,30,3)	77,93 ± 3,54	81,60 ± 4,40	71,37 ± 10,11	70,30 ± 13,03
(16,8,30,6)	78,50 ± 3,06	81,55 ± 3,38	73,82 ± 8,03	75,64 ± 8,44
(16,-,10,3)	75,78 ± 3,72	77,17 ± 6,55	71,46 ± 14,31	61,80 ± 11,82
(16,-,10,6)	75,31 ± 3,83	78,14 ± 4,31	74,66 ± 13,21	70,33 ± 12,33
(16,-,20,3)	76,08 ± 3,84	78,79 ± 4,83	70,23 ± 9,41	61,72 ± 10,57
(16,-,20,6)	76,87 ± 2,70	78,23 ± 5,81	75,46 ± 9,70	68,14 ± 14,19
(16,-,30,3)	76,74 ± 3,55	79,29 ± 4,45	69,79 ± 10,72	63,50 ± 7,27
(16,-,30,6)	76,99 ± 3,81	78,43 ± 3,50	74,45 ± 12,67	74,37 ± 11,98
(8,8,10,3)	77,74 ± 3,84	81,17 ± 2,48	73,86 ± 11,55	72,03 ± 12,57
(8,8,10,6)	77,92 ± 3,86	79,79 ± 4,45	74,74 ± 9,17	75,78 ± 7,41
(8,8,20,3)	78,13 ± 4,84	81,12 ± 3,80	73,49 ± 11,26	69,66 ± 11,84
(8,8,20,6)	78,93 ± 3,03	81,25 ± 4,44	75,91 ± 7,19	74,43 ± 10,71
(8,8,30,3)	78,52 ± 3,35	80,24 ± 4,45	73,64 ± 6,53	67,78 ± 12,13
(8,8,30,6)	77,84 ± 3,67	81,58 ± 3,92	74,94 ± 8,43	75,39 ± 10,09
(8,-,10,3)	76,35 ± 3,16	78,76 ± 4,86	69,07 ± 10,04	62,94 ± 11,55
(8,-,10,6)	75,62 ± 3,24	78,29 ± 3,27	73,61 ± 13,35	69,20 ± 9,74
(8,-,20,3)	75,96 ± 3,24	77,78 ± 4,83	67,40 ± 14,61	66,33 ± 9,47
(8,-,20,6)	76,05 ± 3,94	78,36 ± 4,70	71,93 ± 10,44	68,38 ± 14,29
(8,-,30,3)	75,46 ± 3,03	77,90 ± 4,27	68,29 ± 10,39	64,64 ± 6,98
(8,-,30,6)	75,90 ± 3,52	78,24 ± 5,19	74,41 ± 11,75	67,65 ± 14,55

C.3 Curvas de Treinamento

C.3.1 Águas Profundas

A Figura C.11 mostra as curvas de treinamento dos 10-*folds* do modelo LOFAR MLP (30 neurônios). Cada sub-imagem possui 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

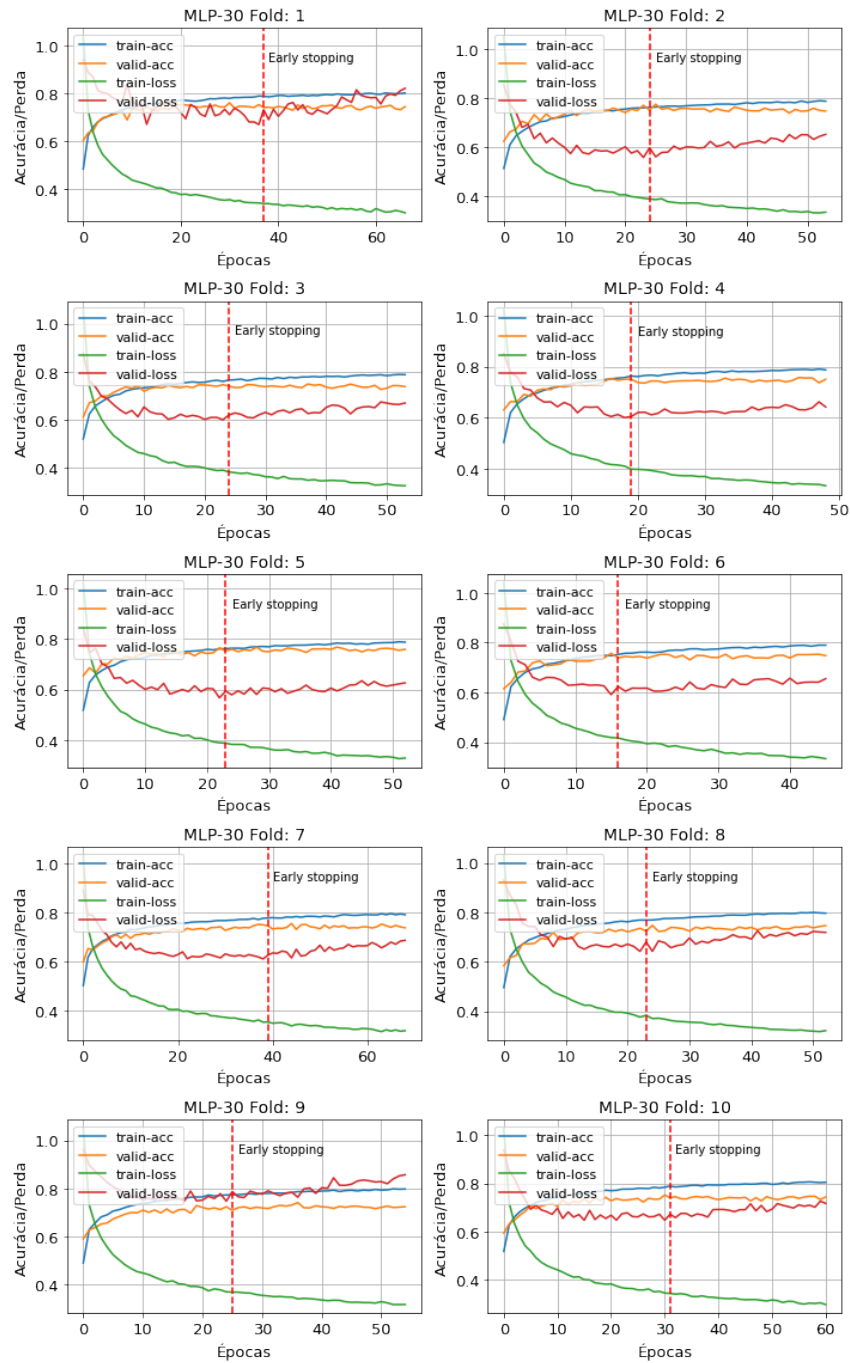


Figura C.11: Curvas de Treinamento do modelo LOFAR MLP (águas profundas).

A Figura C.12 mostra as curvas de treinamento dos 10-*folds* do modelo temporal MLP (30 neurônios). Cada sub-imagem possui 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

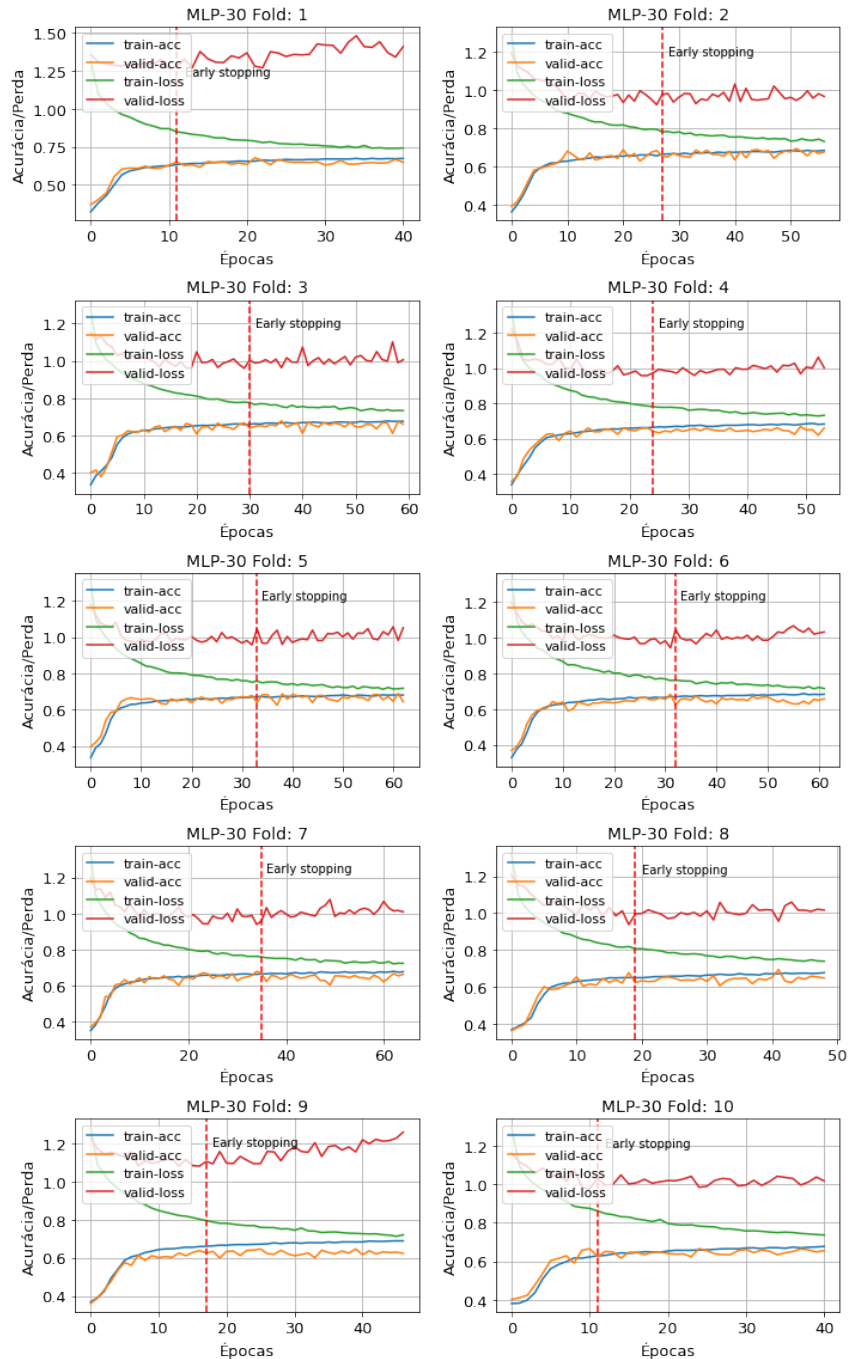


Figura C.12: Curvas de Treinamento do modelo temporal MLP (águas profundas).

A Figura C.13 mostra as curvas de treinamento dos 10-*fold*s do modelo LOFAR CNN (8,8,20,3). Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Stopping*.

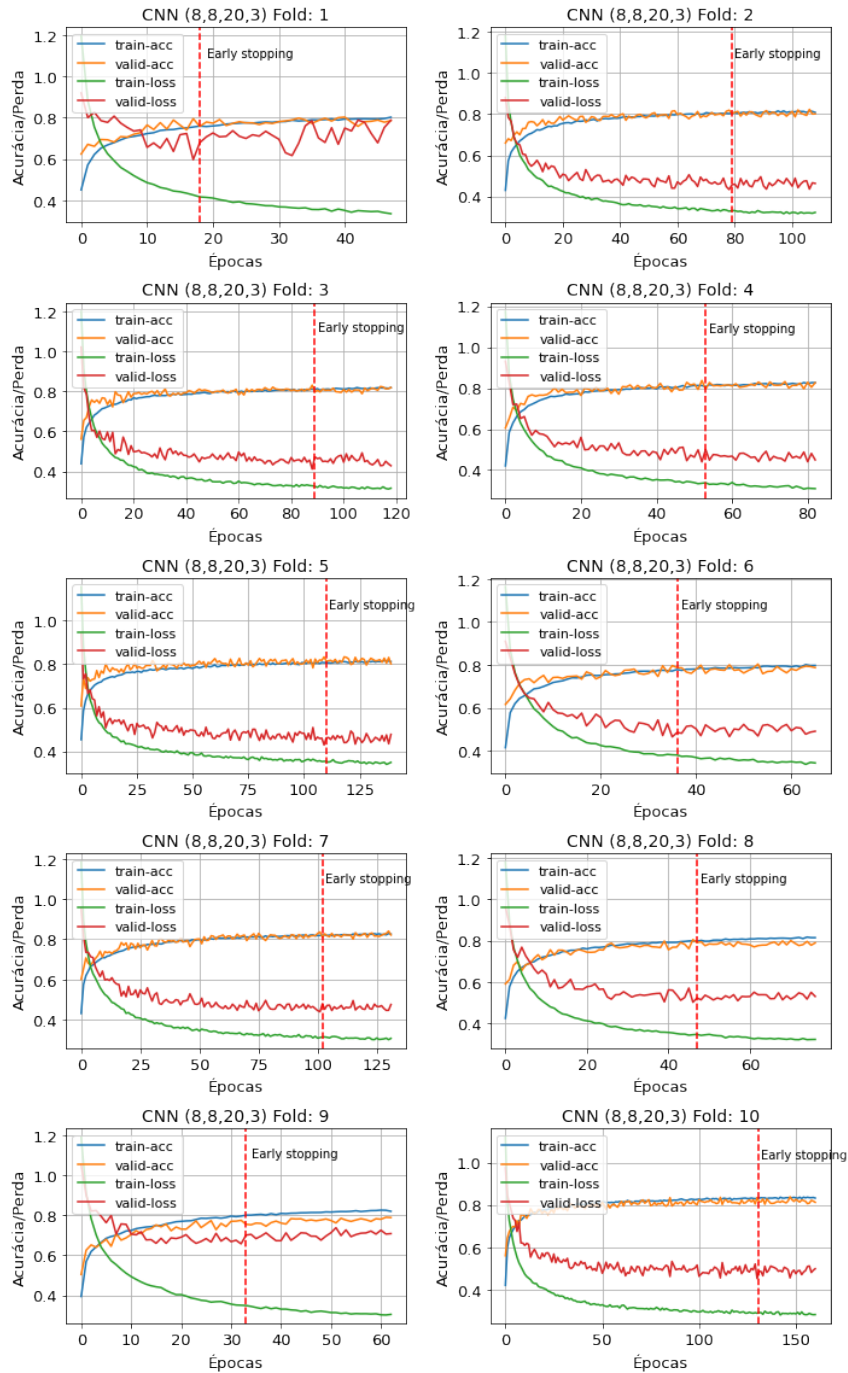


Figura C.13: Curvas de Treinamento do modelo LOFAR CNN (águas profundas).

A Figura C.14 mostra as curvas de treinamento dos 10-*fold*s do modelo temporal CNN (8,8,20,3). Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

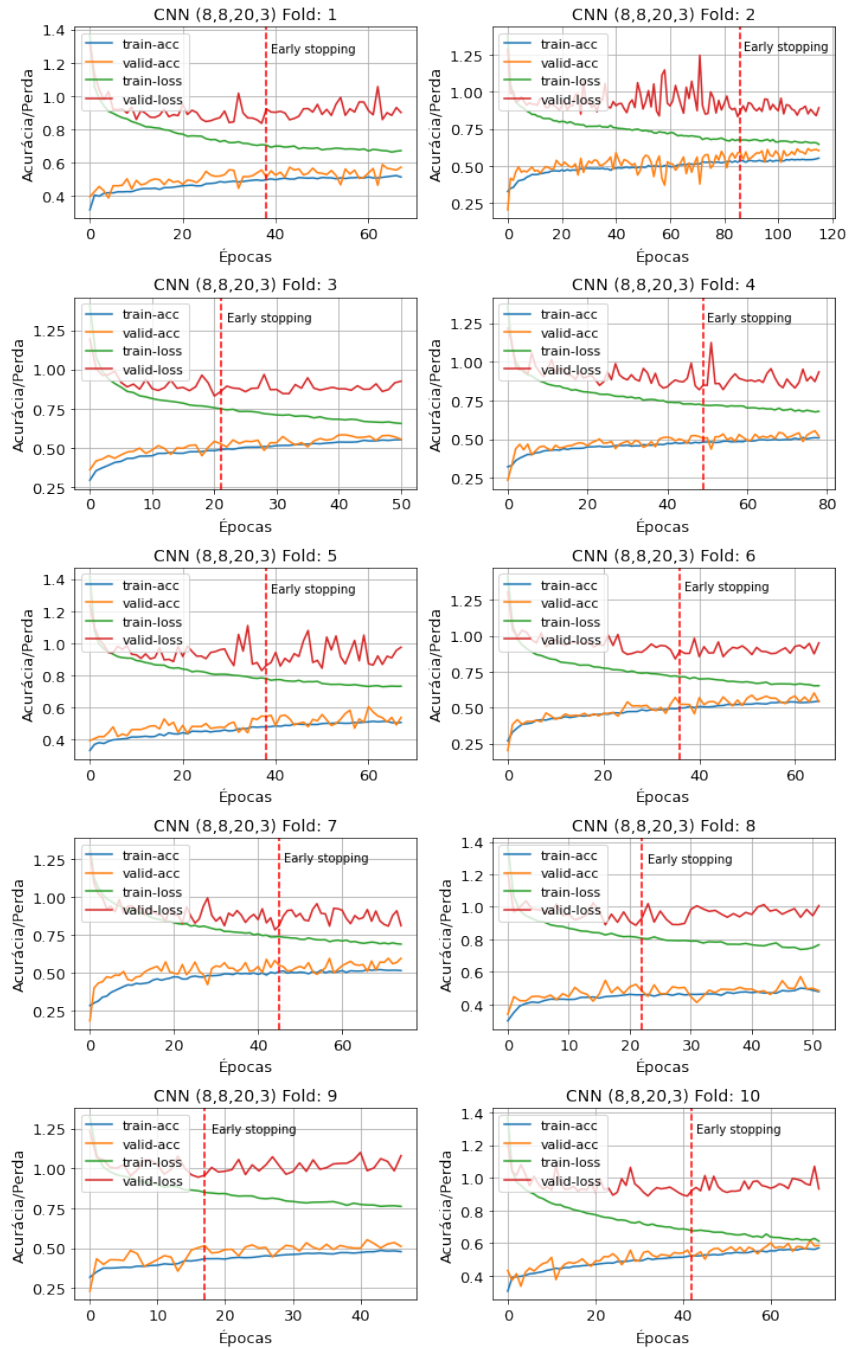


Figura C.14: Curvas de Treinamento do modelo temporal CNN (águas profundas).

A Figura C.15 mostra as curvas de treinamento dos 10-*folds* do modelo temporal LSTM (75 células). Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

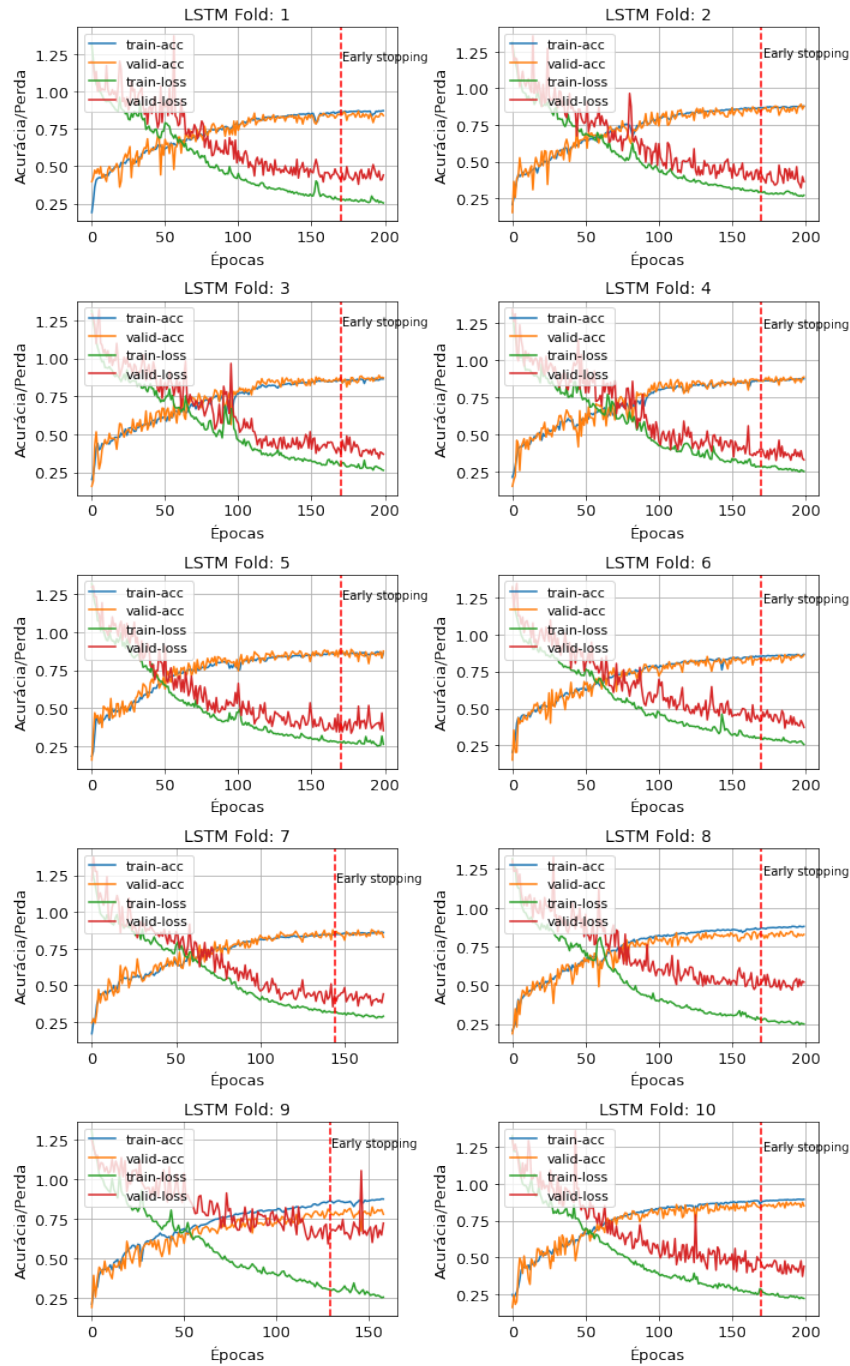


Figura C.15: Curvas de Treinamento do modelo temporal LSTM (águas profundas).

Fusão dos modelos

A Figura C.16 mostra as curvas de treinamento dos 10-*fold*s da fusão dos modelos MLP. Cada sub-imagem possui 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

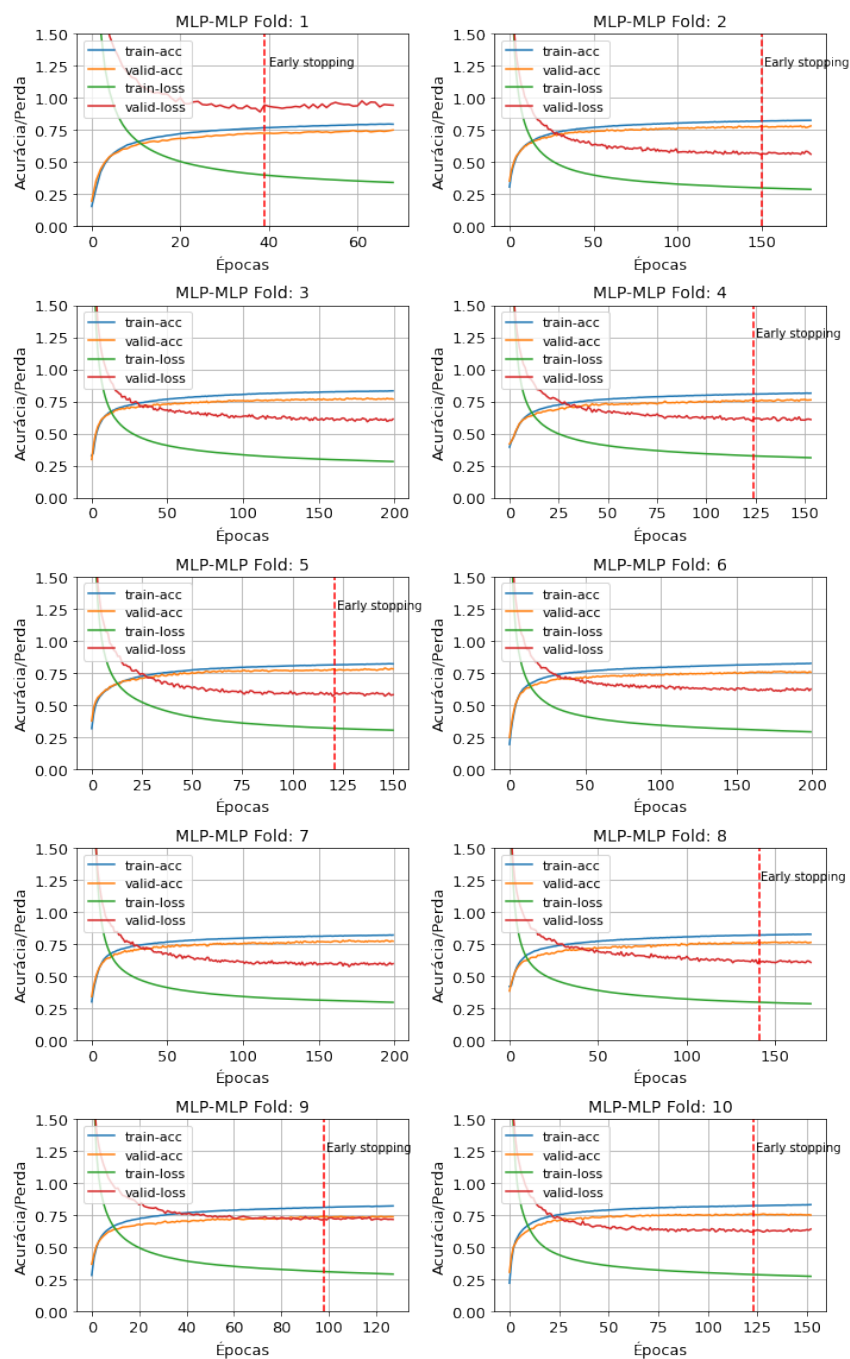


Figura C.16: Curvas de Treinamento da fusão dos modelos MLP (águas profundas).

A Figura C.17 mostra as curvas de treinamento dos 10-*fold*s da fusão dos modelos CNN. Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

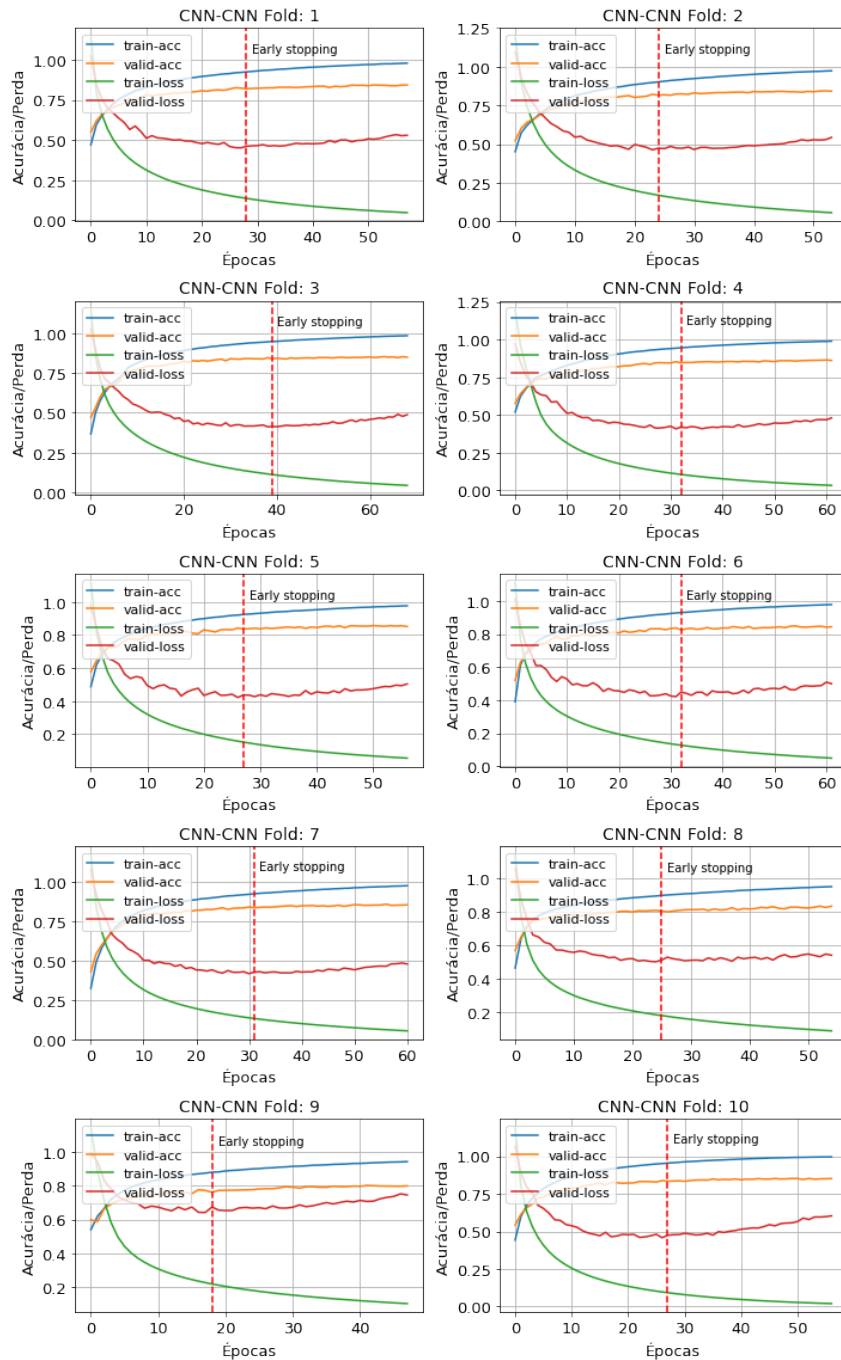


Figura C.17: Curvas de Treinamento da fusão dos modelos CNN (águas profundas).

A Figura C.18 mostra as curvas de treinamento dos 10-*folds* da fusão dos modelos MLP e LSTM. Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

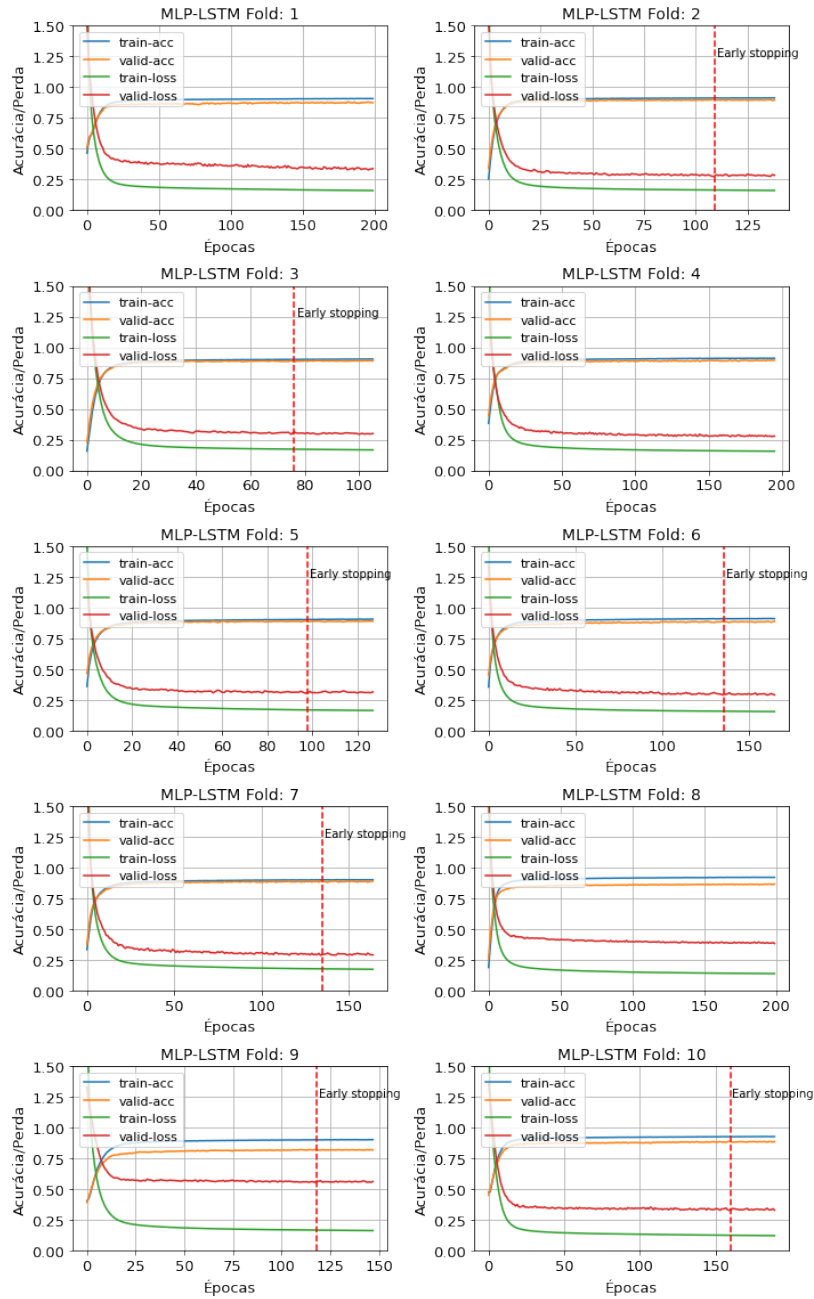


Figura C.18: Curvas de Treinamento da fusão dos modelos MLP e LSTM (águas profundas).

A Figura C.19 mostra as curvas de treinamento dos 10- *folds* da fusão dos modelos CNN e LSTM. Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

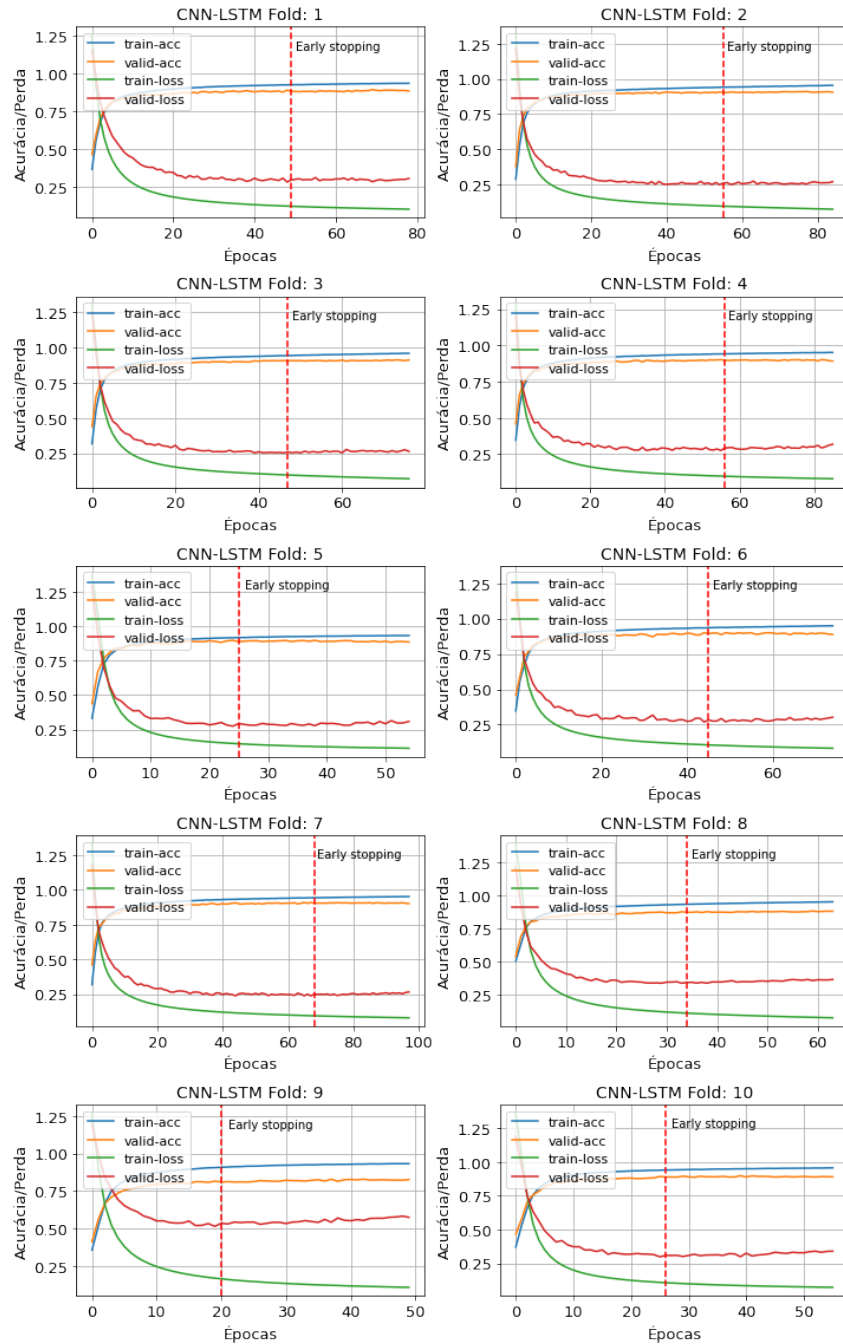


Figura C.19: Curvas de Treinamento da fusão dos modelos CNN e LSTM (águas profundas).

C.3.2 Águas Rasas

A Figura C.20 mostra as curvas de treinamento dos 10-*folds* do modelo LOFAR MLP (10 neurônios). Cada sub-imagem possui 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

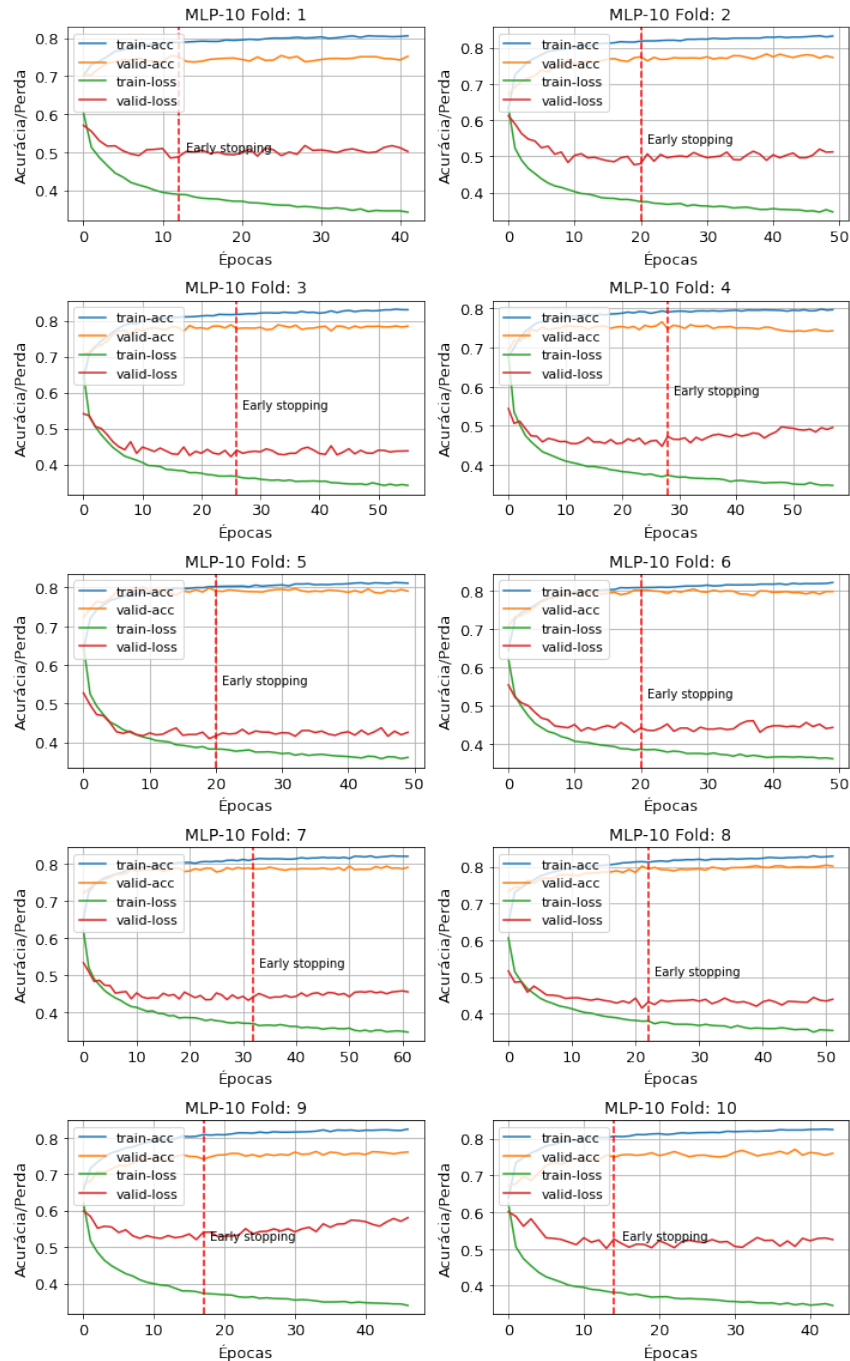


Figura C.20: Curvas de Treinamento do modelo LOFAR MLP (águas rasas).

A Figura C.21 mostra as curvas de treinamento dos 10-*folds* do modelo temporal MLP (10 neurônios). Cada sub-imagem possui 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

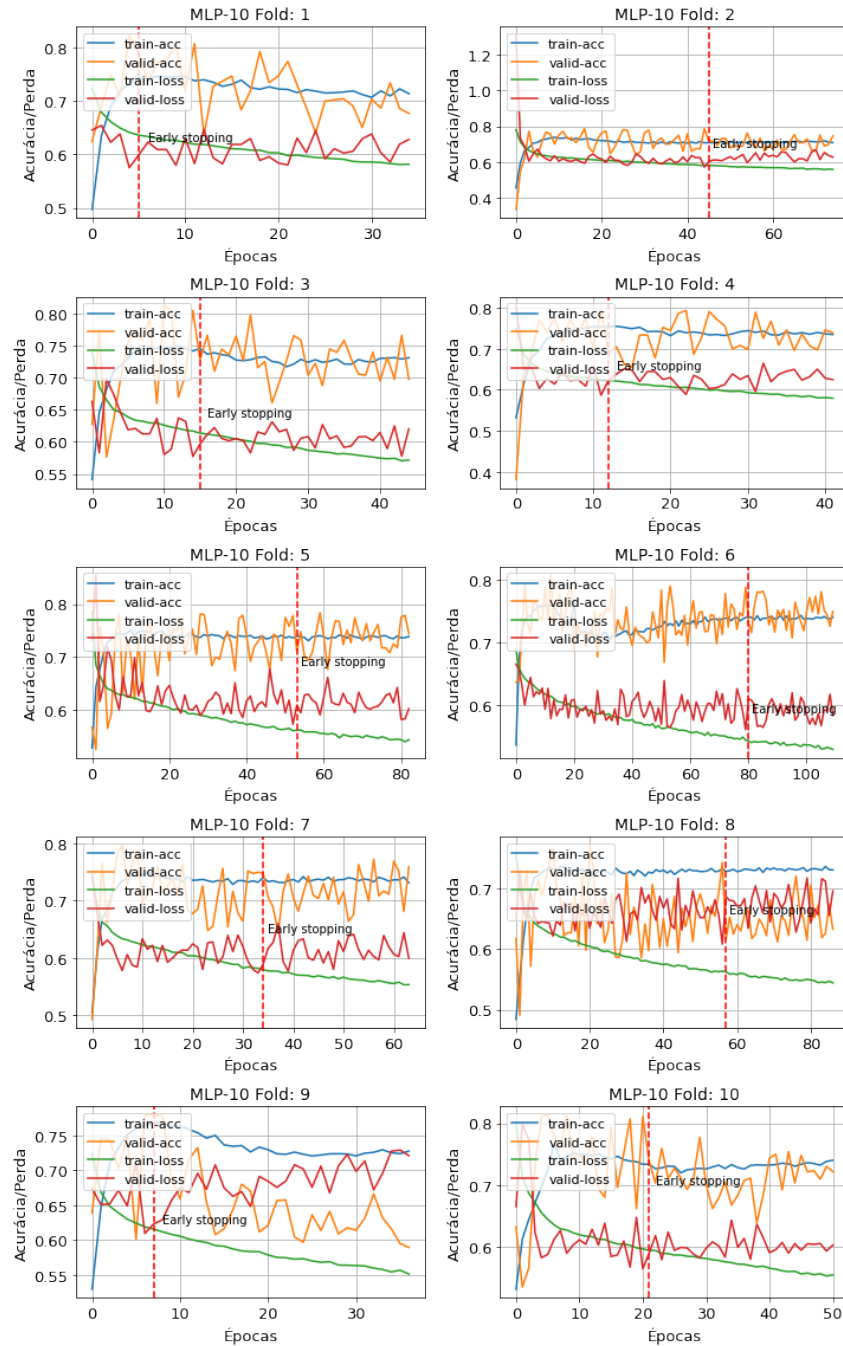


Figura C.21: Curvas de Treinamento do modelo temporal MLP (águas rasas).

A Figura C.22 mostra as curvas de treinamento dos 10-*fold*s do modelo LOFAR CNN (8,8,10,3). Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Stopping*.

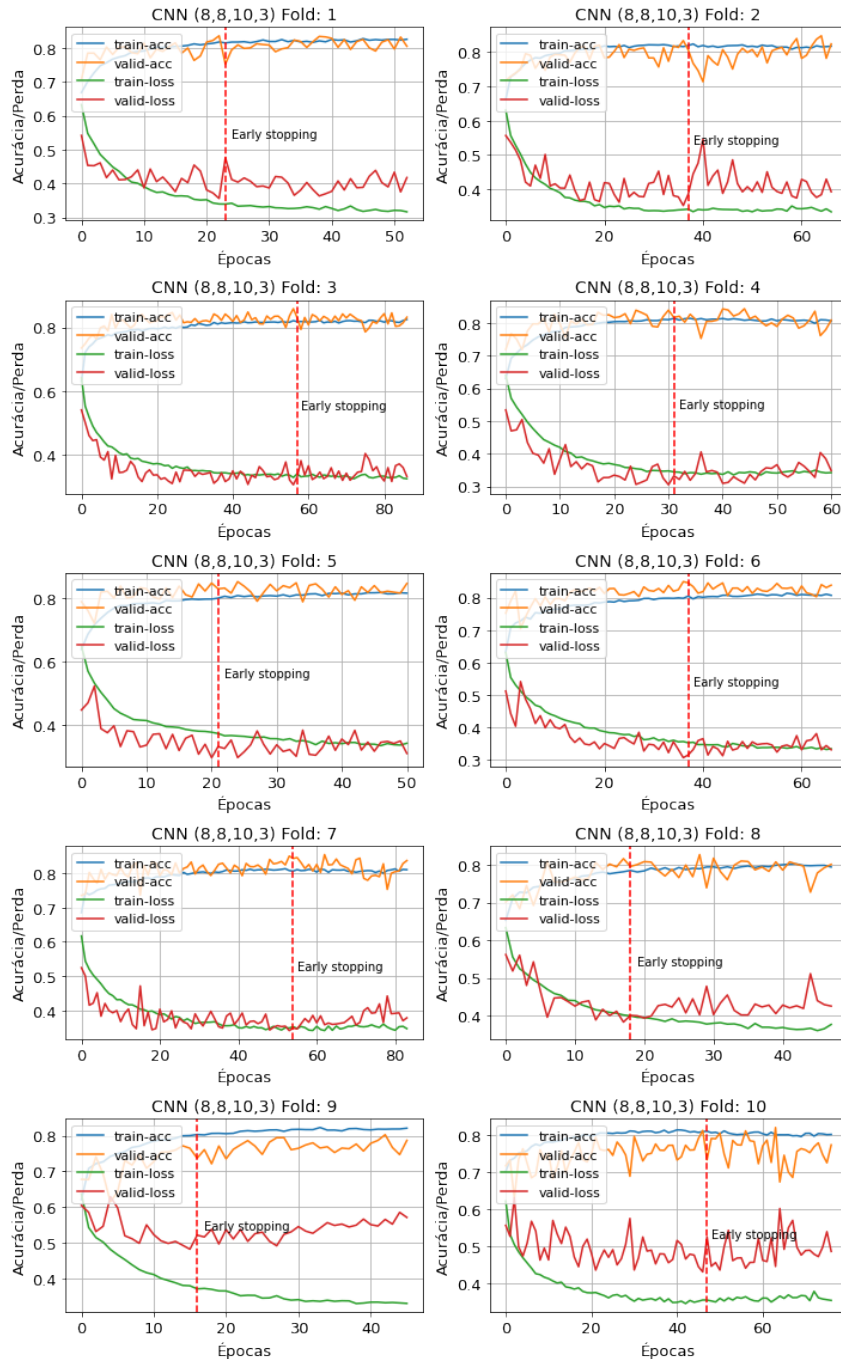


Figura C.22: Curvas de Treinamento do modelo LOFAR CNN (águas rasas).

A Figura C.23 mostra as curvas de treinamento dos 10-*folds* do modelo temporal CNN (8,8,10,3). Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Stopping*.

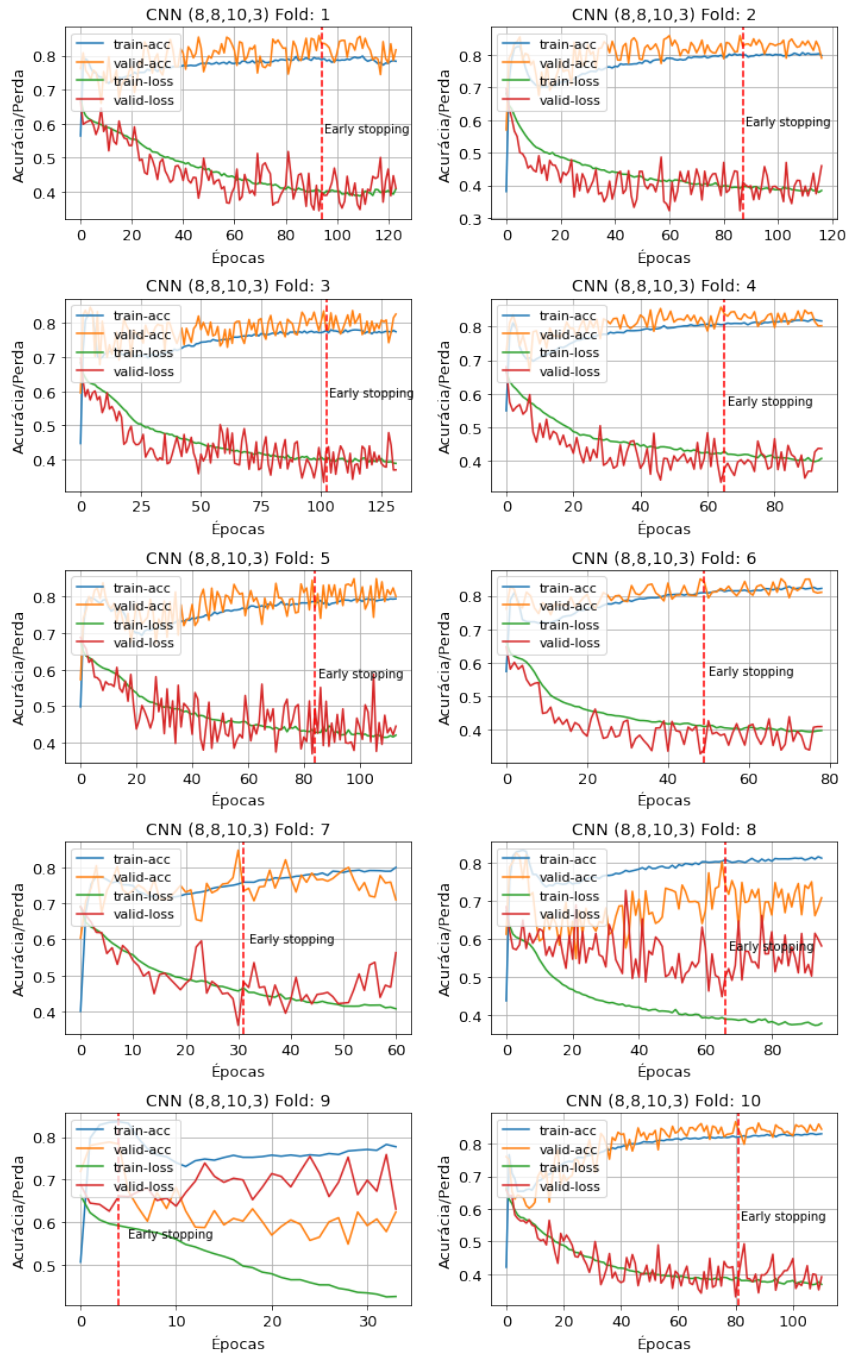


Figura C.23: Curvas de Treinamento do modelo temporal CNN (águas rasas).

A Figura C.24 mostra as curvas de treinamento dos 10-*folds* do modelo temporal LSTM (50 células). Cada sub-imagem possui 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

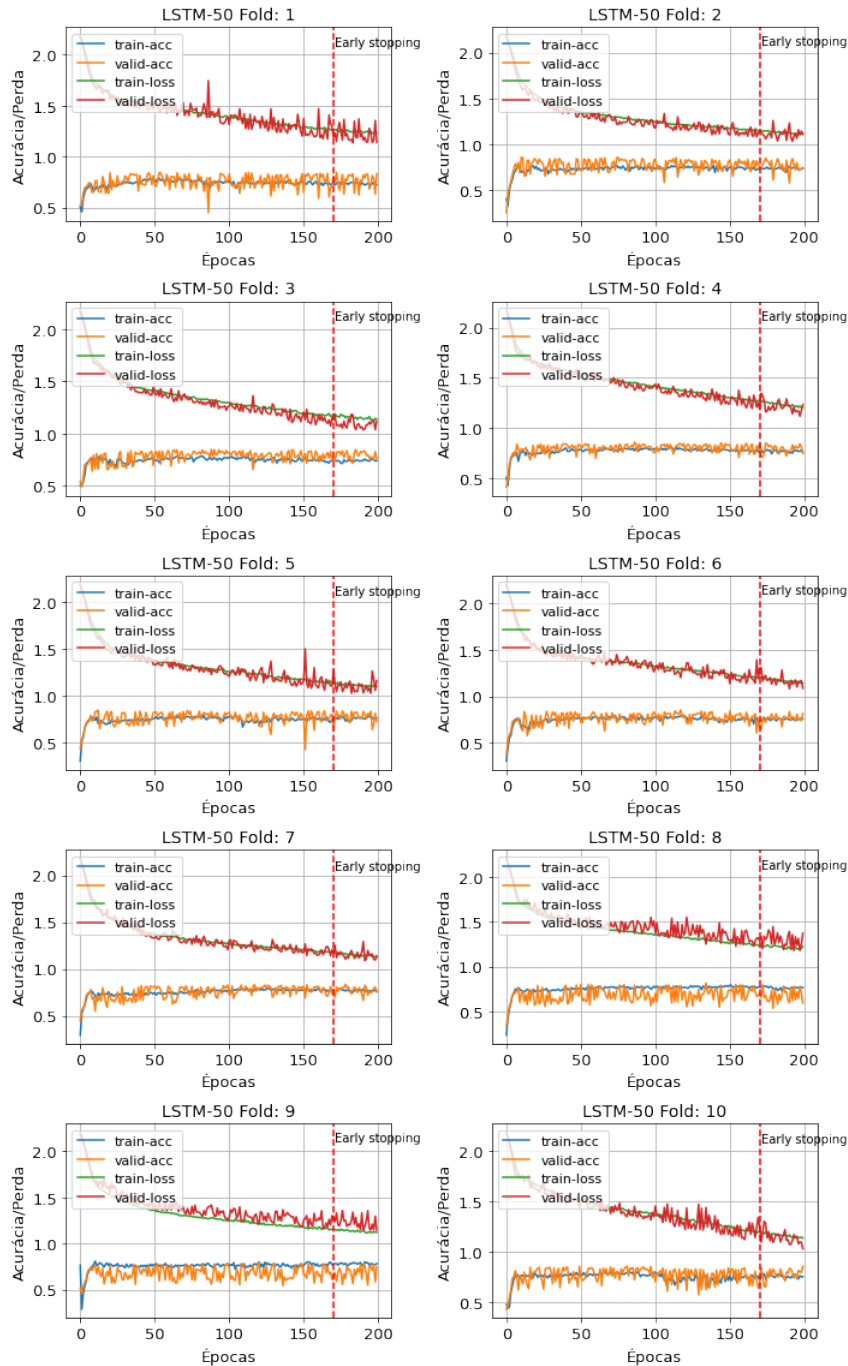


Figura C.24: Curvas de Treinamento do modelo temporal LSTM (águas rasas).

Fusão dos modelos

A Figura C.25 mostra as curvas de treinamento dos 10-*fold*s da fusão dos modelos MLP. Cada sub-imagem possui 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

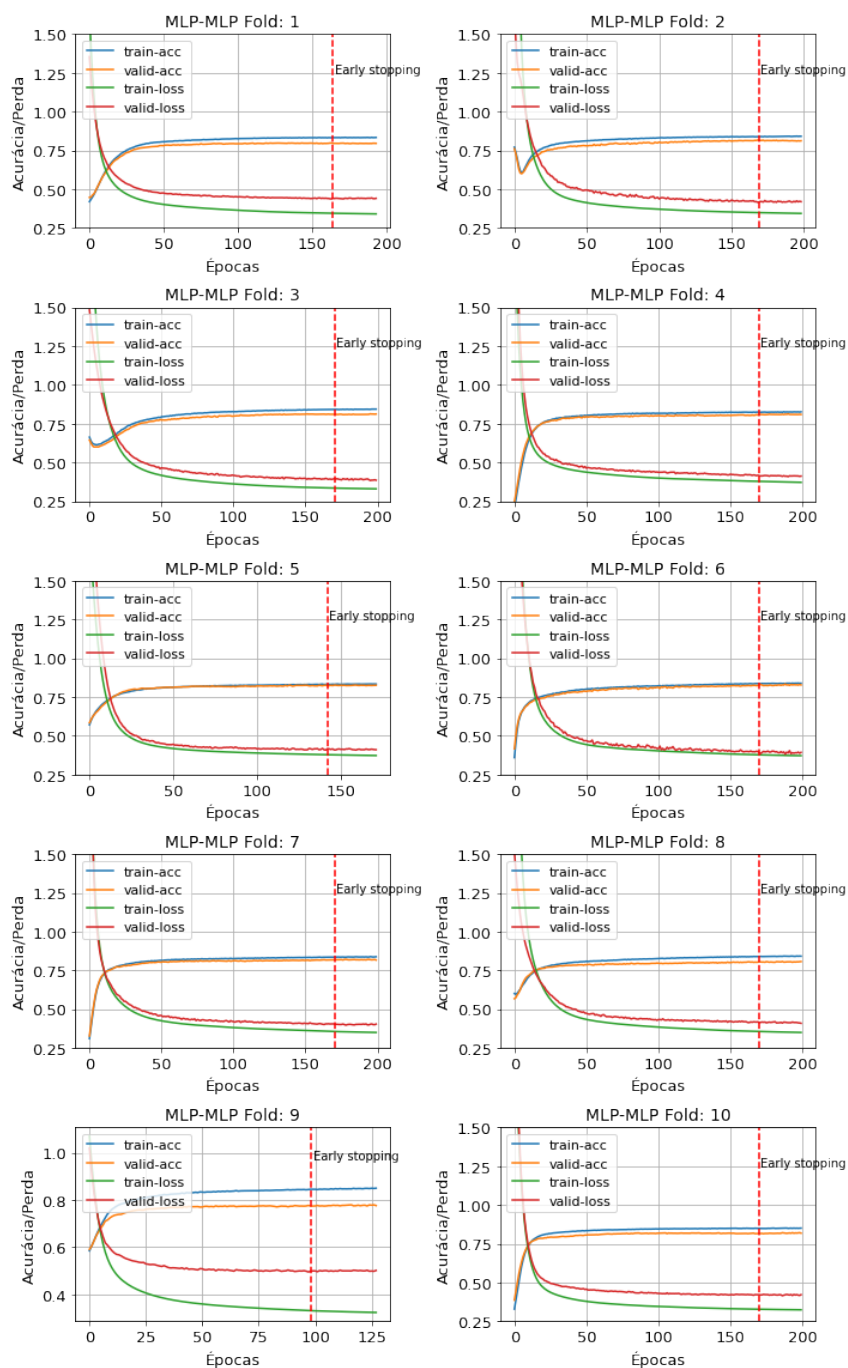


Figura C.25: Curvas de Treinamento da fusão dos modelos MLP (águas rasas).

A Figura C.26 mostra as curvas de treinamento dos 10-*fold*s da fusão dos modelos CNN. Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spotting*.

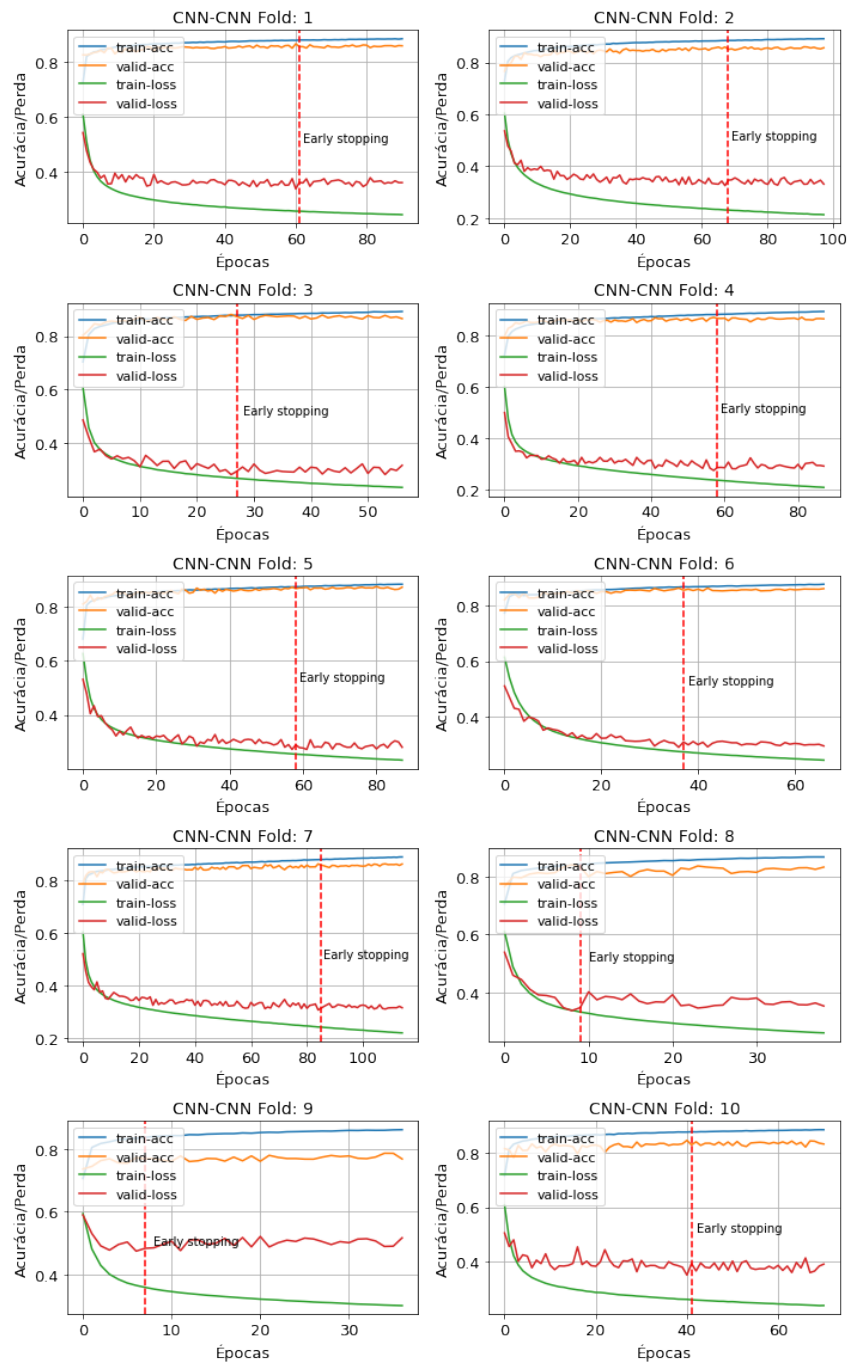


Figura C.26: Curvas de Treinamento da fusão dos modelos CNN (águas rasas).

A Figura C.27 mostra as curvas de treinamento dos 10-*folds* da fusão dos modelos MLP e LSTM. Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

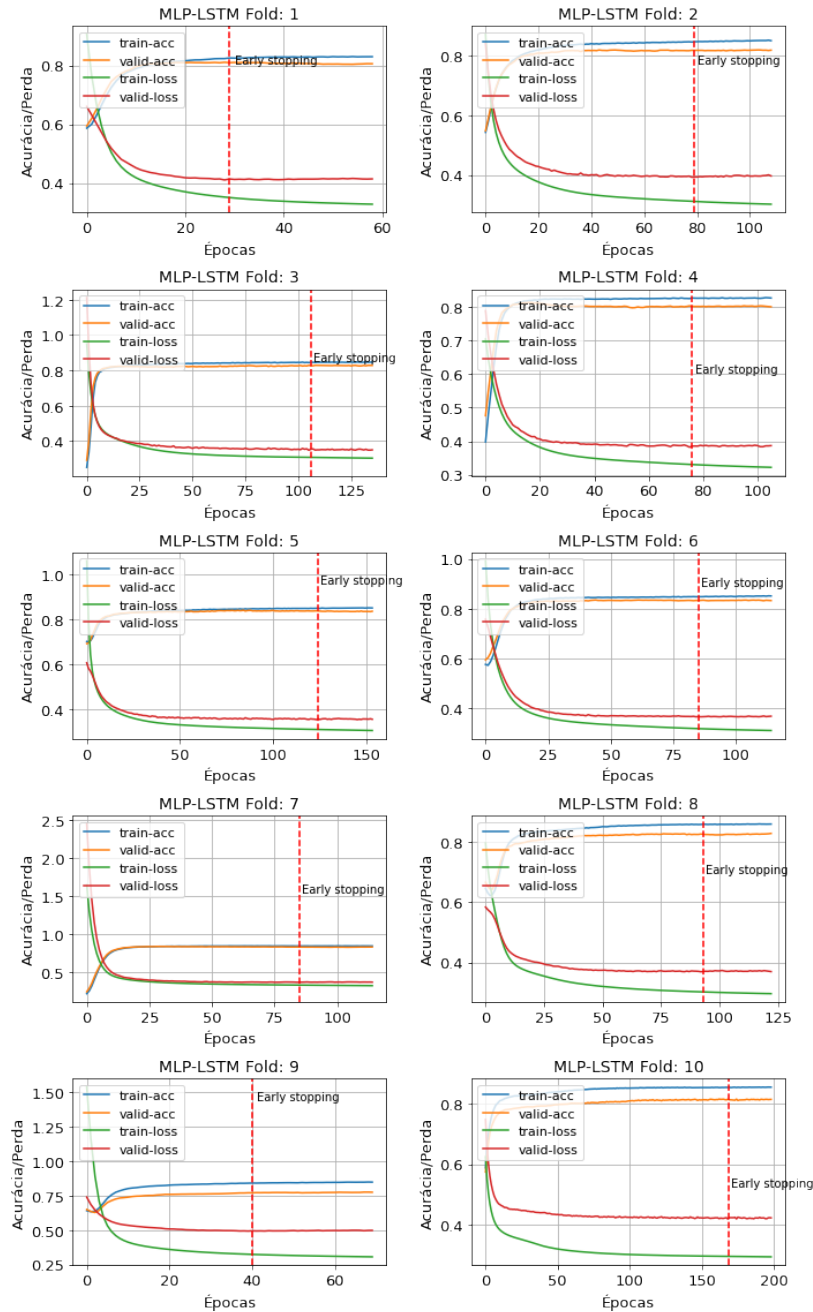


Figura C.27: Curvas de Treinamento da fusão dos modelos MLP e LSTM (águas rasas).

A Figura C.28 mostra as curvas de treinamento dos 10-*folds* da fusão dos modelos CNN e LSTM. Cada sub-imagem possuiu 4 curvas, de acurácia e perdas dos conjuntos de treinamento e validação. A linha tracejada vermelha representa do *Early Spoting*.

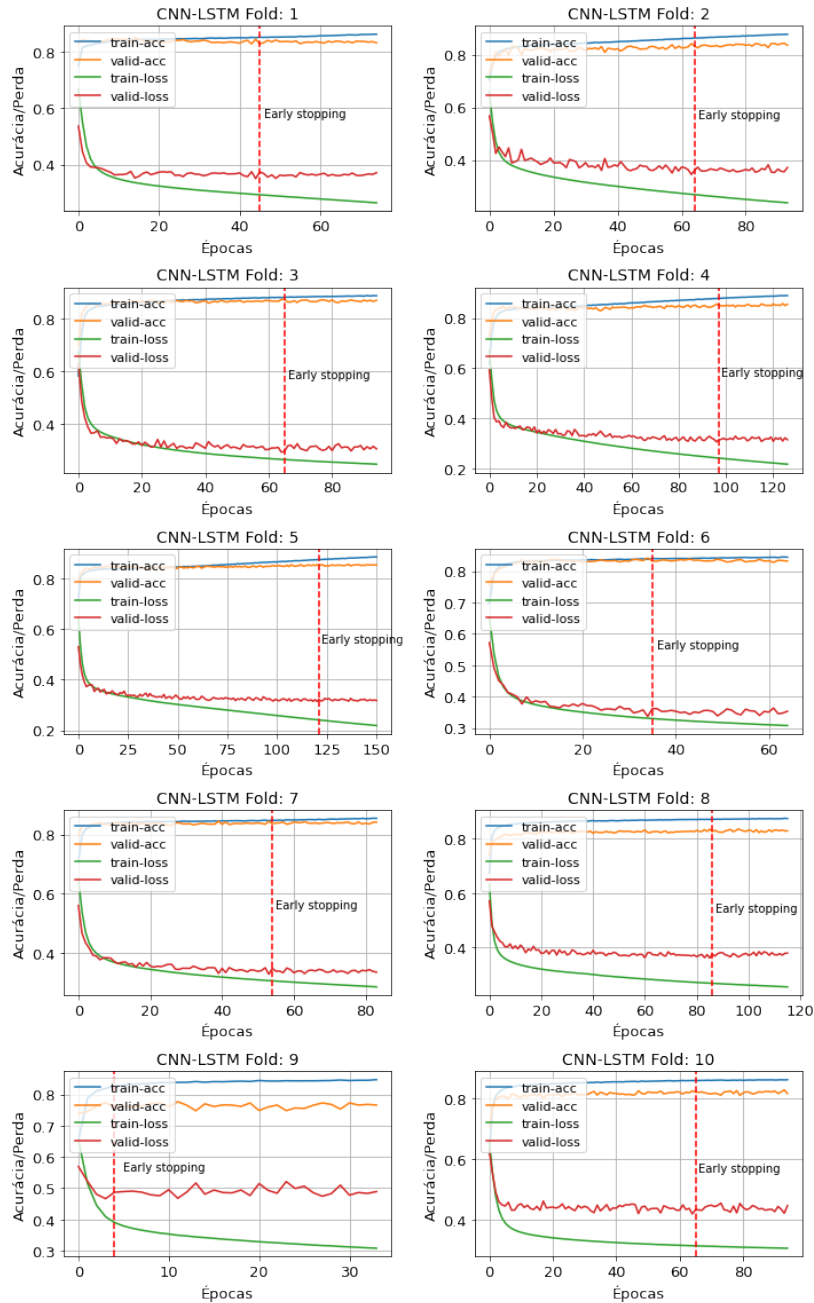
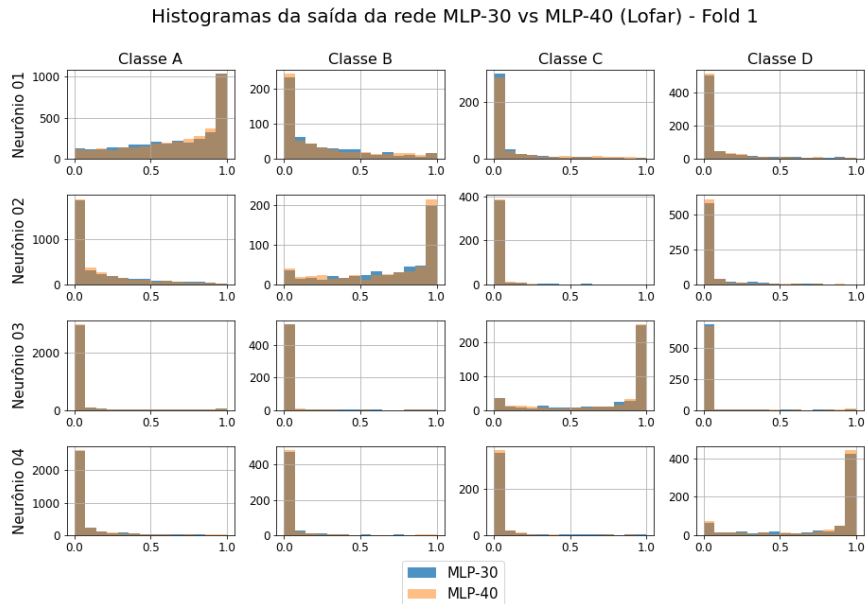


Figura C.28: Curvas de Treinamento da fusão dos modelos CNN e LSTM (águas rasas).

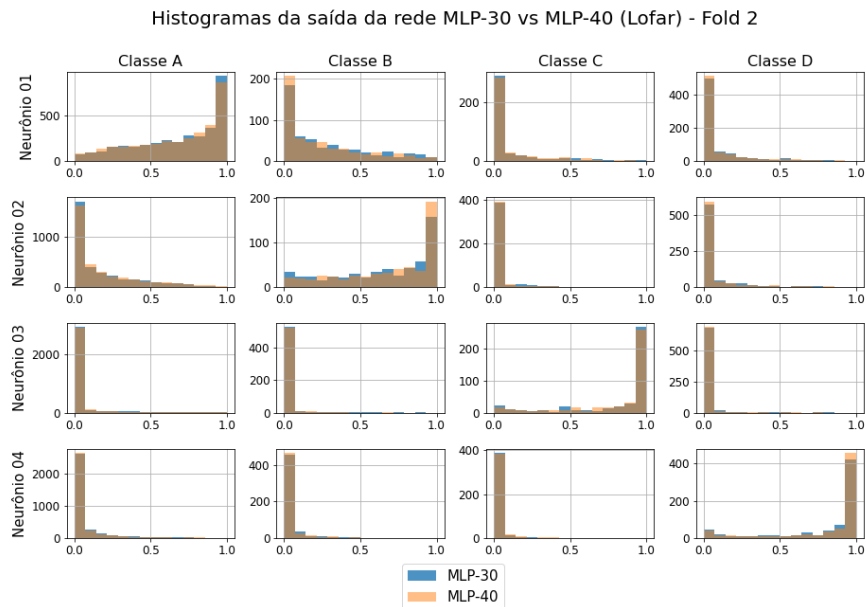
C.4 Histogramas de saída da Rede

C.4.1 Águas Profundas

As Figuras de C.29 a C.33 mostram os histogramas da saída dos 10 *folds* dos modelos LOFAR MLP com 30 e 40 neurônios na camada oculta.



(a) *fold-1*



(b) *fold-2*

Figura C.29: Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (*folds* 1 e 2).

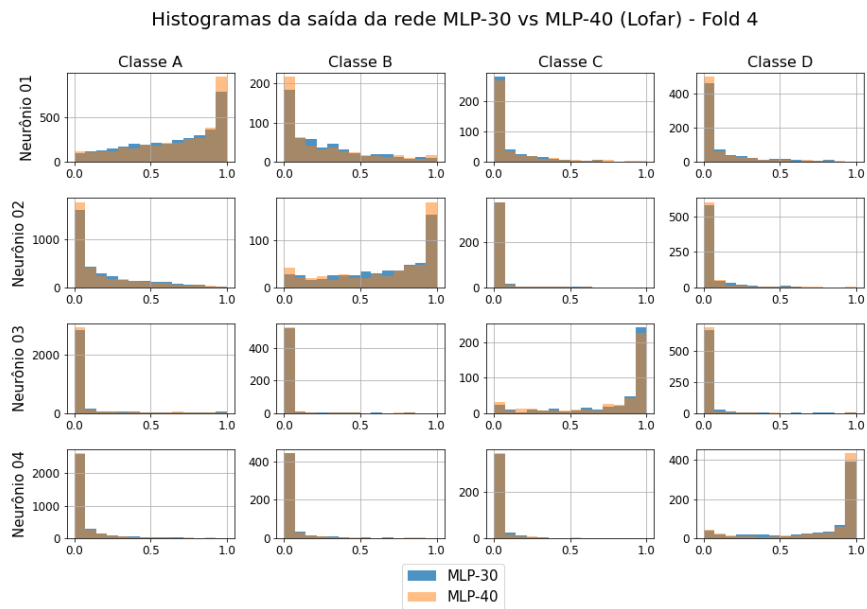
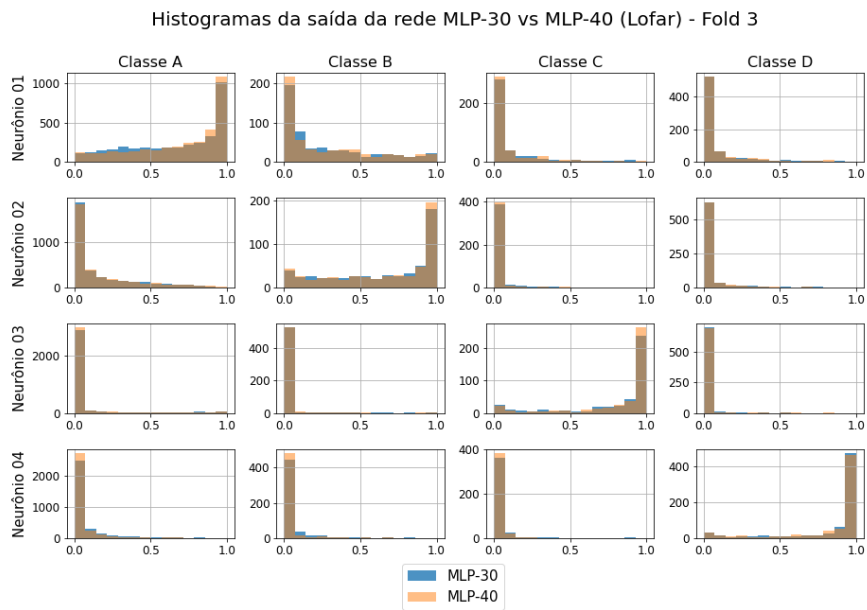


Figura C.30: Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (*fold*s 3 e 4).

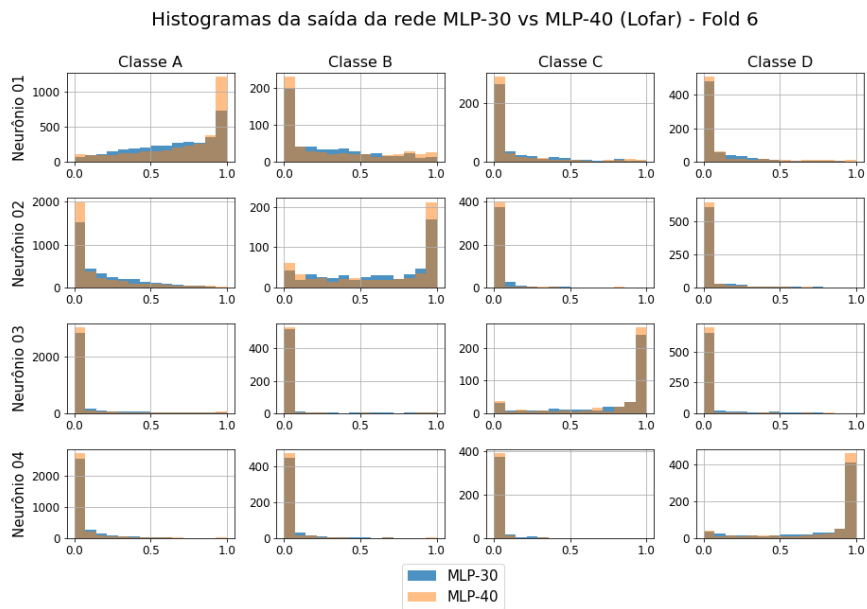
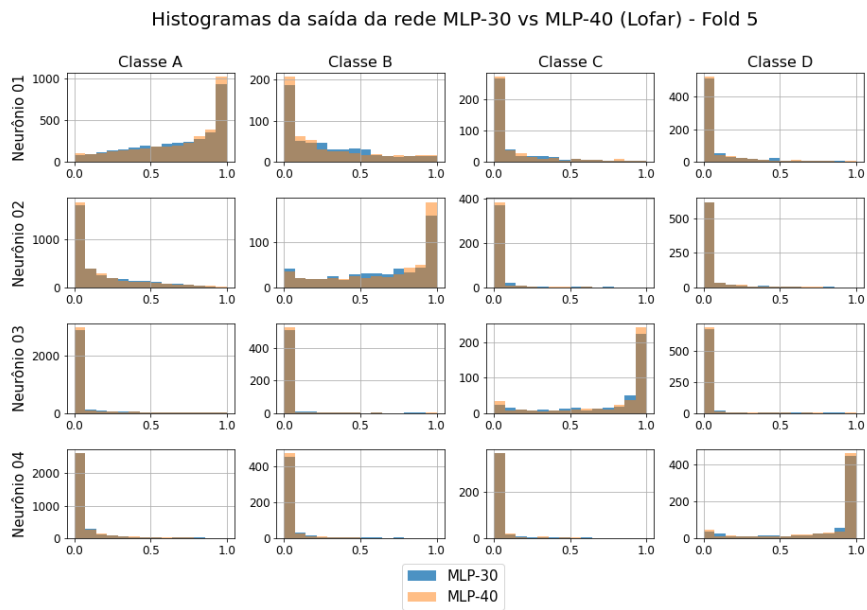


Figura C.31: Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (*fold*s 5 e 6).

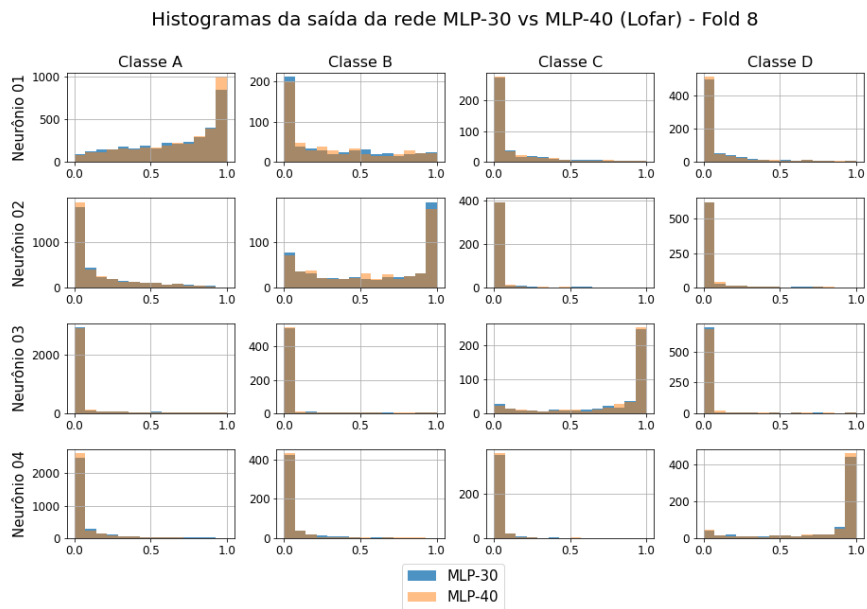
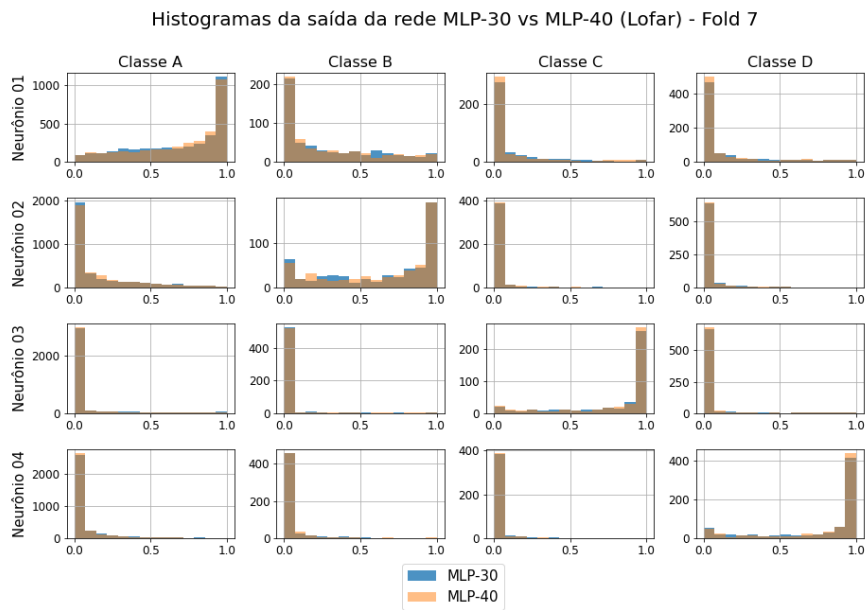


Figura C.32: Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (*folds* 7 e 8).

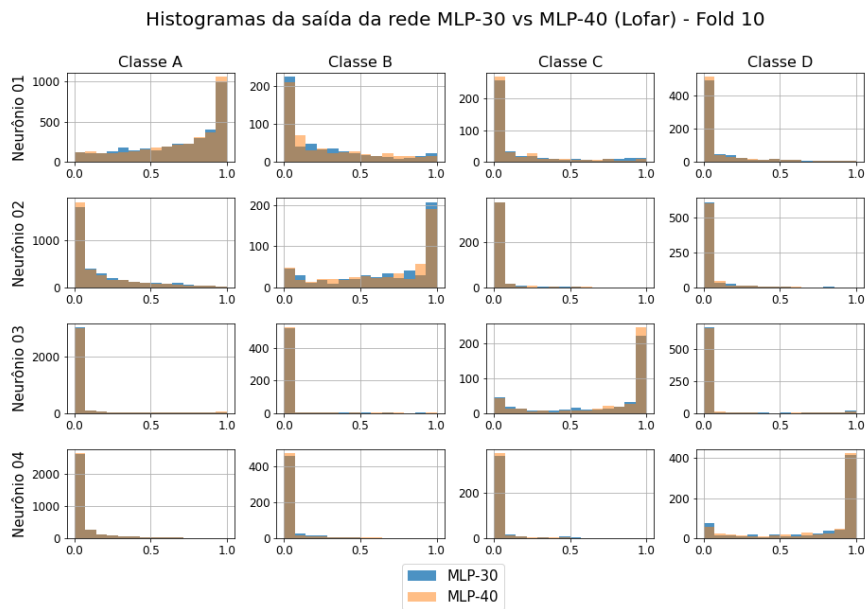
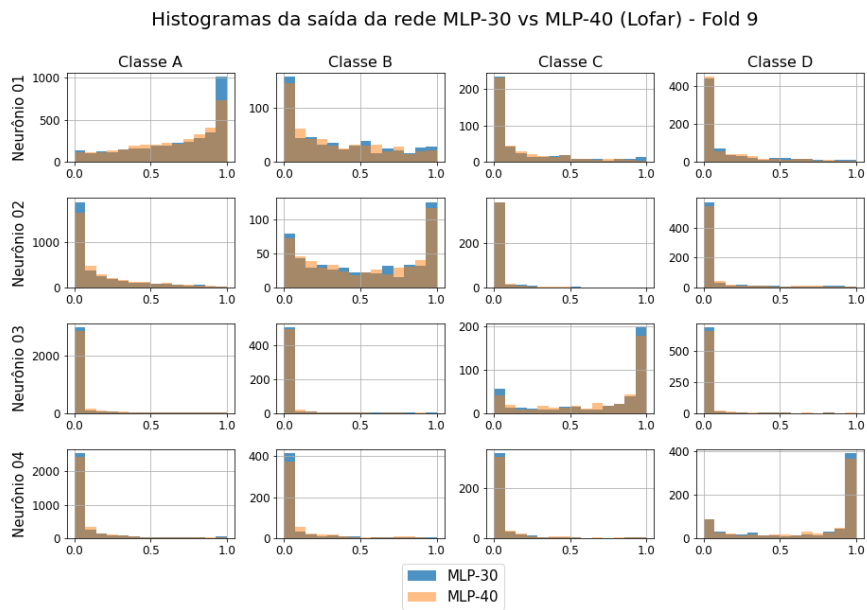


Figura C.33: Histogramas da saída dos modelos LOFAR MLP-30 e MLP-40 (*fold*s 9 e 10).

As Figuras de C.34 a C.38 mostram os histogramas da saída dos 10 *fold*s do modelo temporal MLP com 10 neurônios na camada oculta.

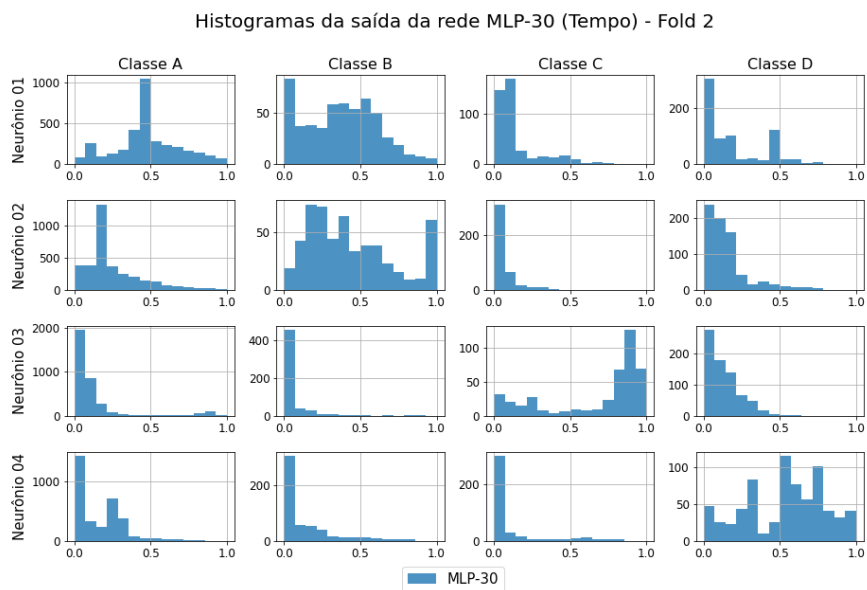
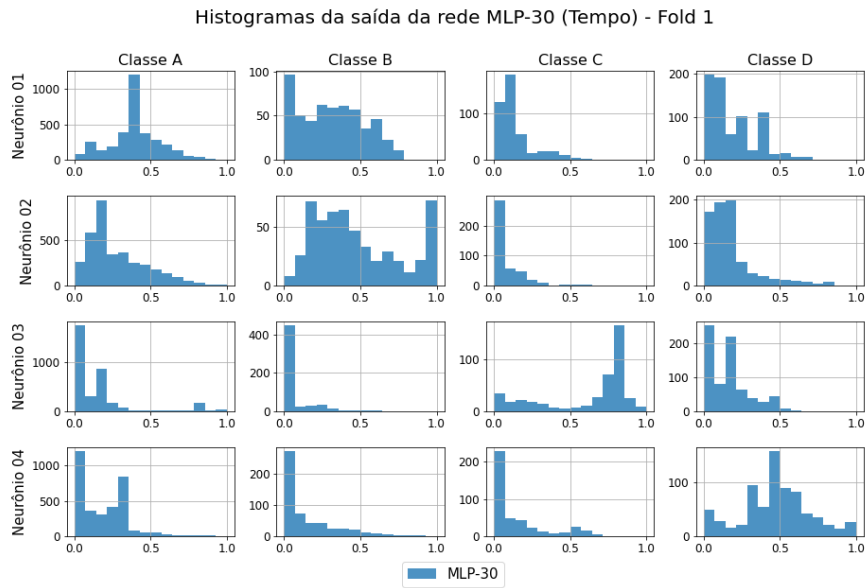
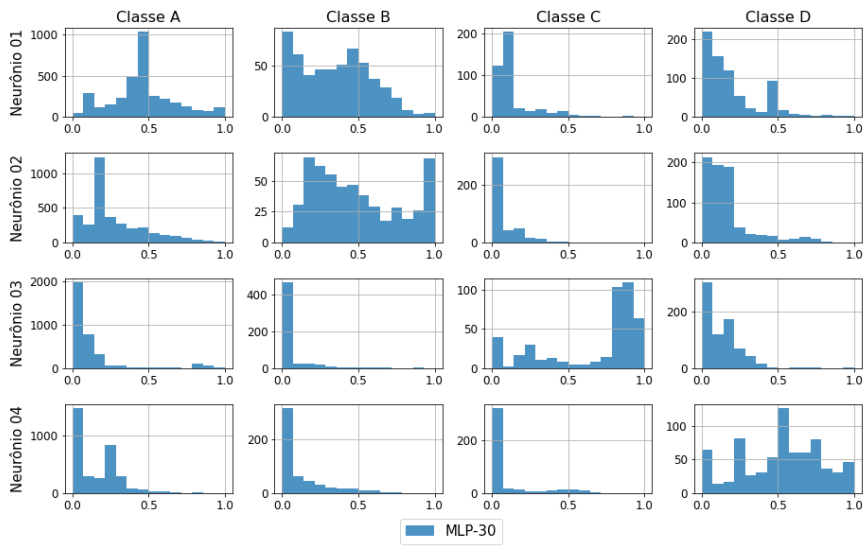


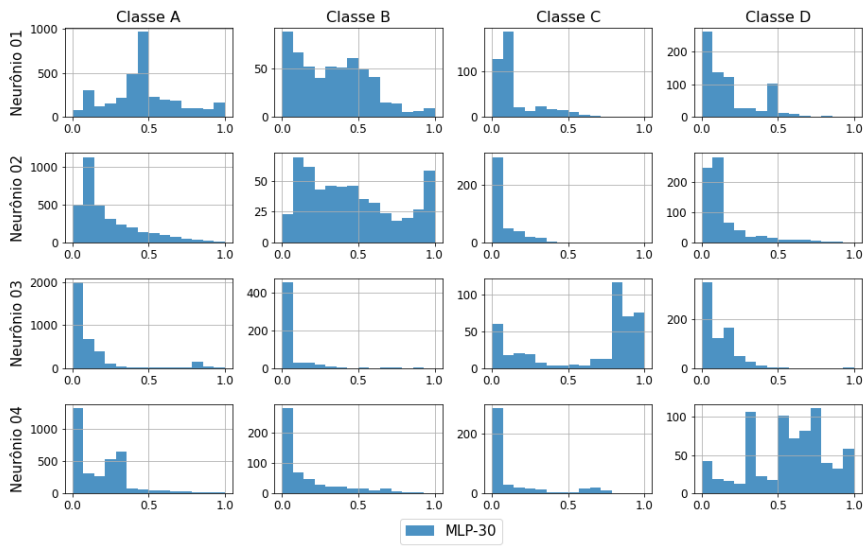
Figura C.34: Histogramas da saída do modelo temporal MLP-30 (*fold*s 1 e 2).

Histogramas da saída da rede MLP-30 (Tempo) - Fold 3



(a) *fold-3*

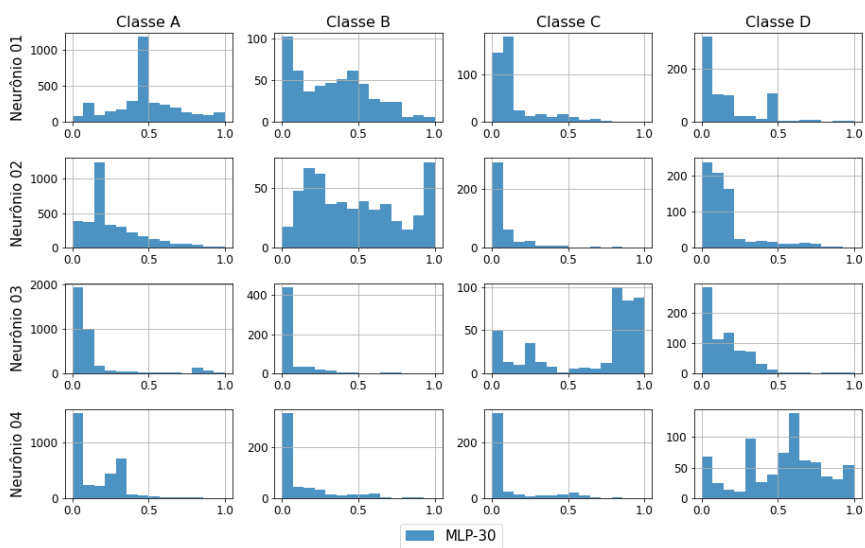
Histogramas da saída da rede MLP-30 (Tempo) - Fold 4



(b) *fold-4*

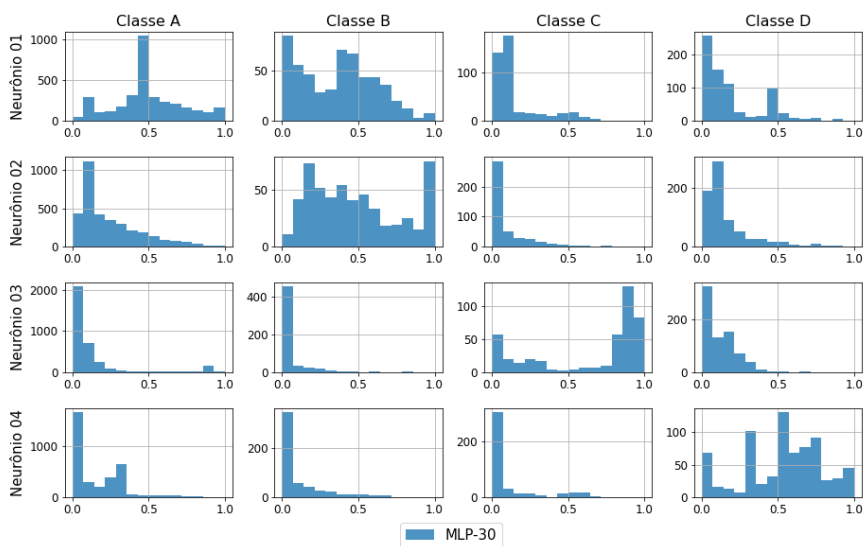
Figura C.35: Histogramas da saída do modelo temporal MLP-30 (*folds* 3 e 4).

Histogramas da saída da rede MLP-30 (Tempo) - Fold 5



(a) *fold-5*

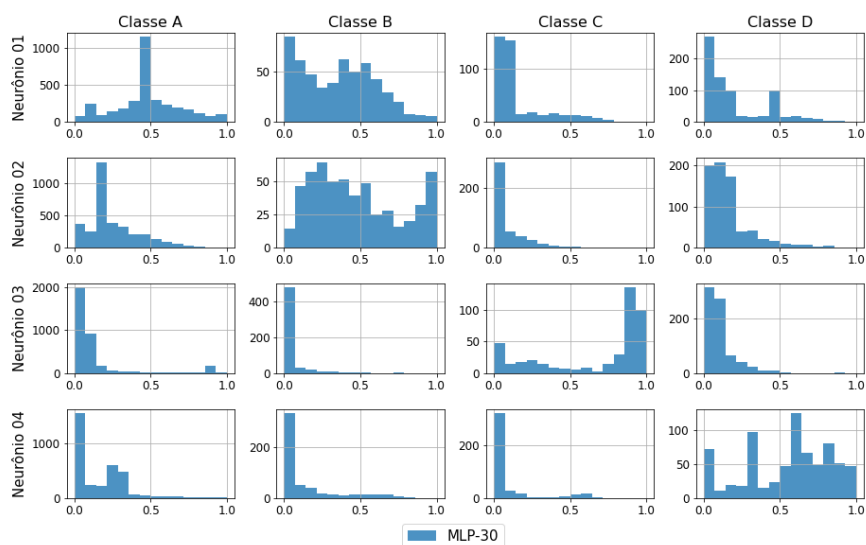
Histogramas da saída da rede MLP-30 (Tempo) - Fold 6



(b) *fold-6*

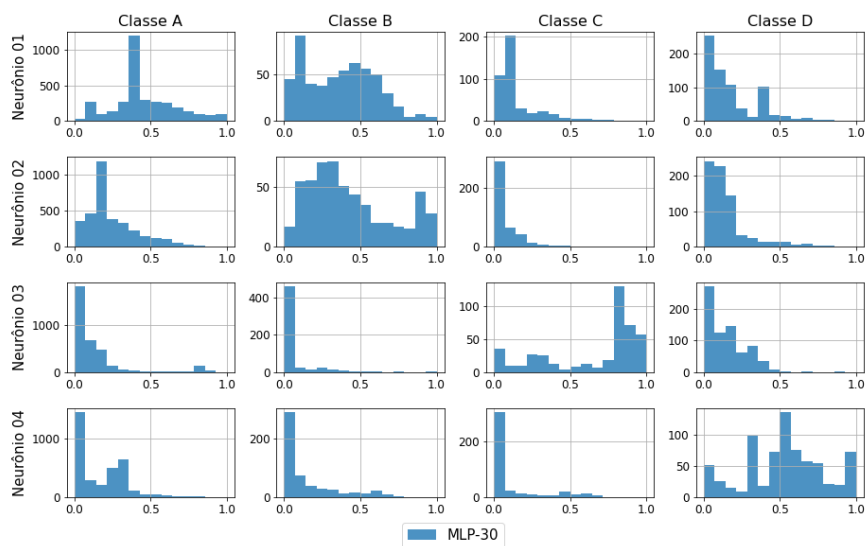
Figura C.36: Histogramas da saída do modelo temporal MLP-30 (*folds* 5 e 6).

Histogramas da saída da rede MLP-30 (Tempo) - Fold 7



(a) *fold-7*

Histogramas da saída da rede MLP-30 (Tempo) - Fold 8



(b) *fold-8*

Figura C.37: Histogramas da saída do modelo temporal MLP-30 (*folds* 7 e 8).

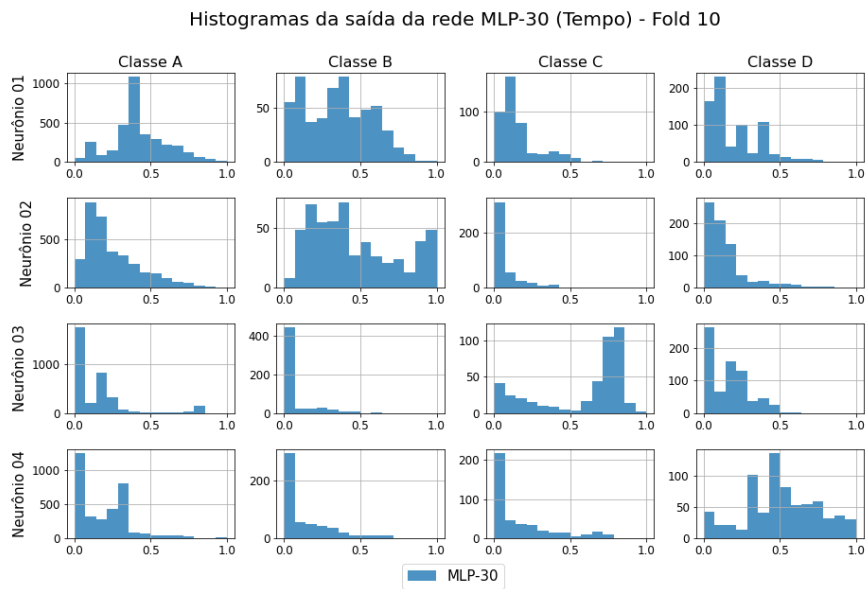
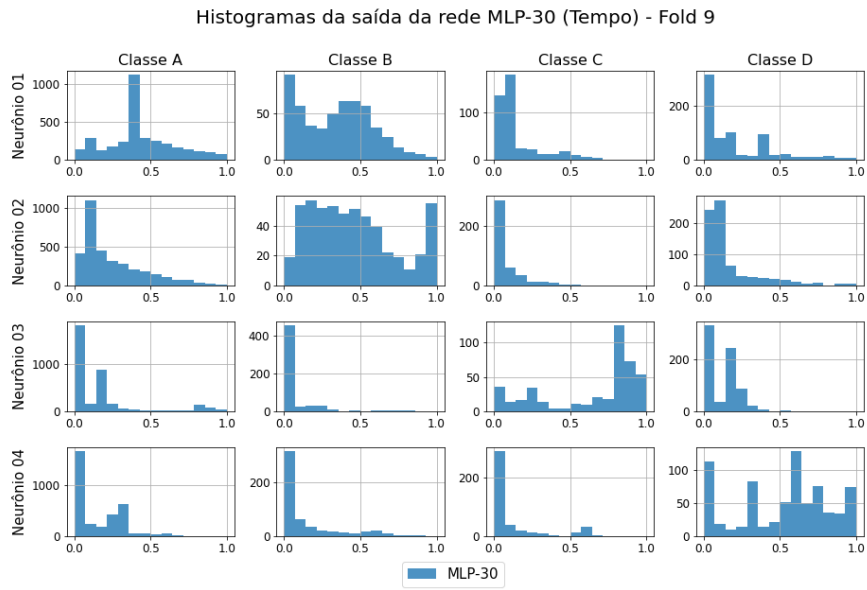
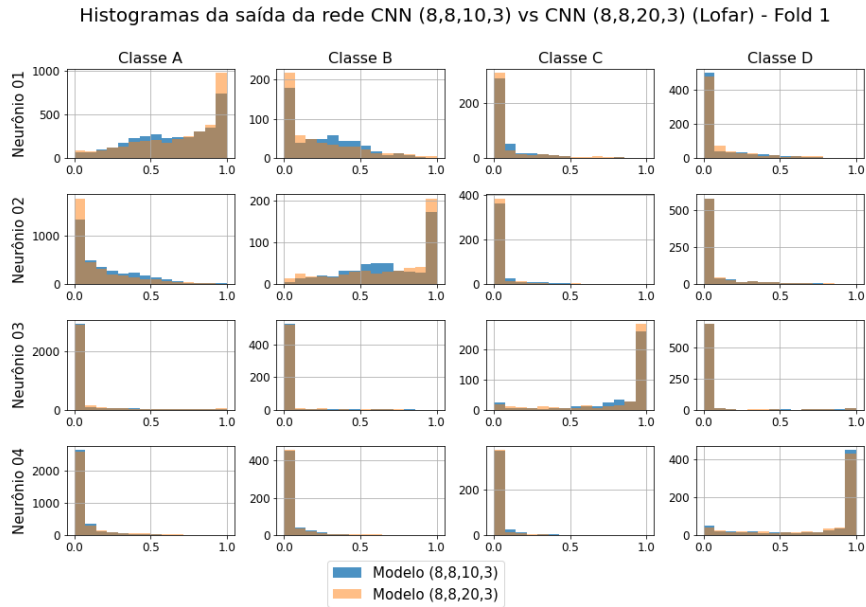
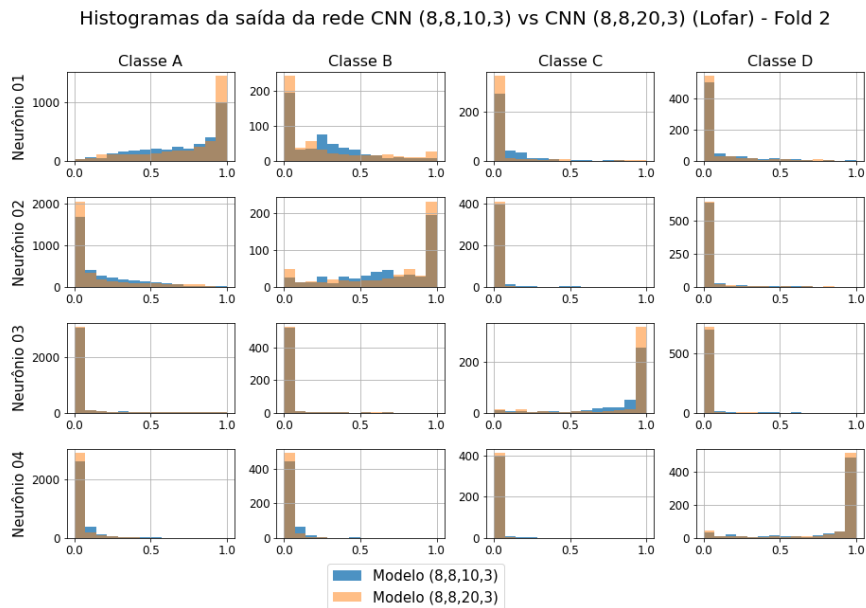


Figura C.38: Histogramas da saída do modelo temporal MLP-30 (*folds* 9 e 10).

As Figuras de C.39 a C.43 mostram os histogramas da saída dos 10 *folds* dos modelos LOFAR CNN com 10 e 20 neurônios na camada *fully connected*, a extração de características é feita por duas camadas convolucionais de 8 filtros cada e *kernel* 3.



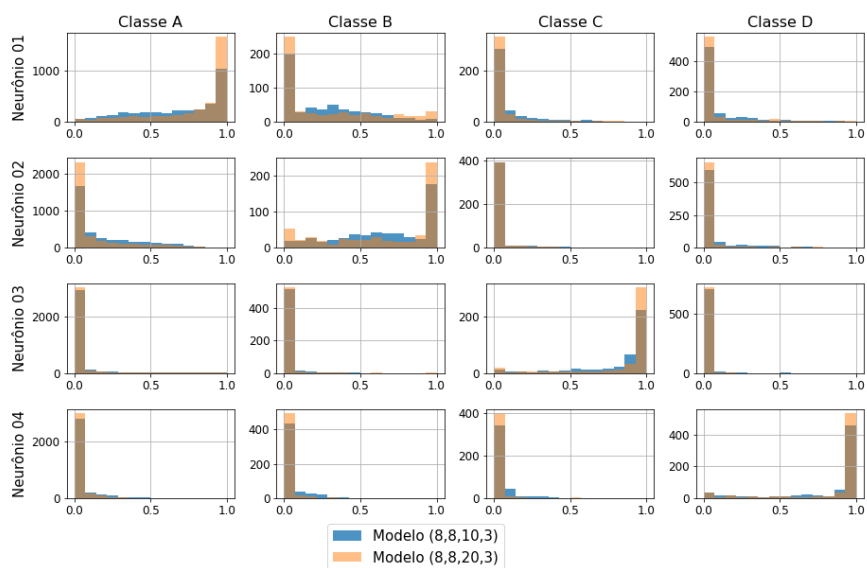
(a) *fold-1*



(b) *fold-2*

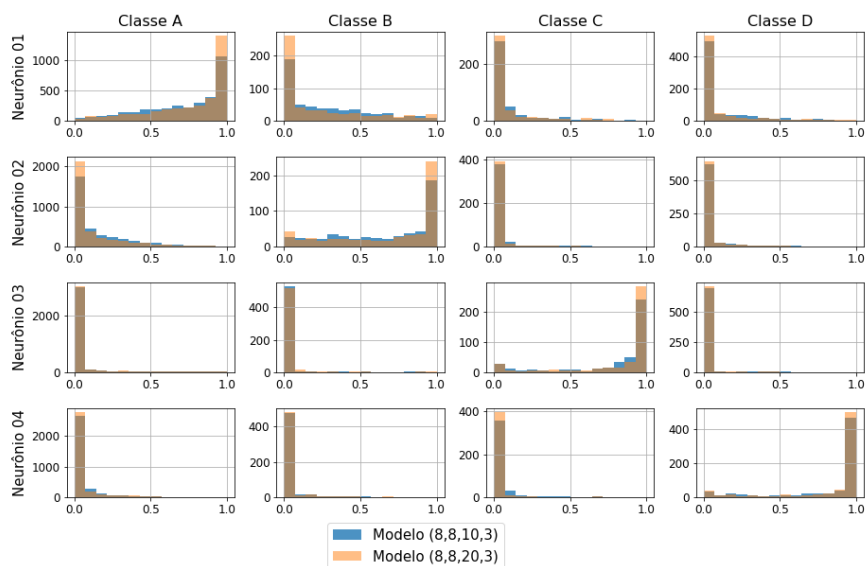
Figura C.39: Histogramas da saída dos modelos LOFAR CNN (*folds* 1 e 2).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 3



(a) *fold-3*

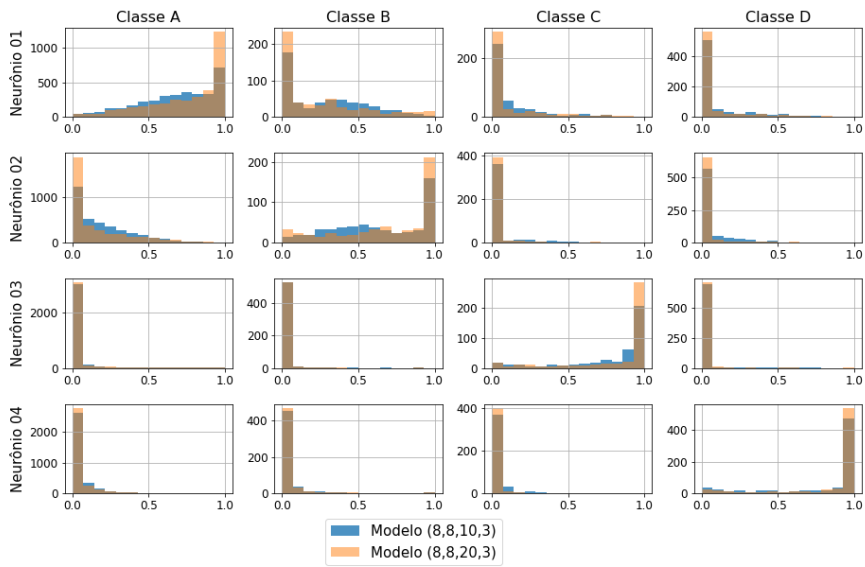
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 4



(b) *fold-4*

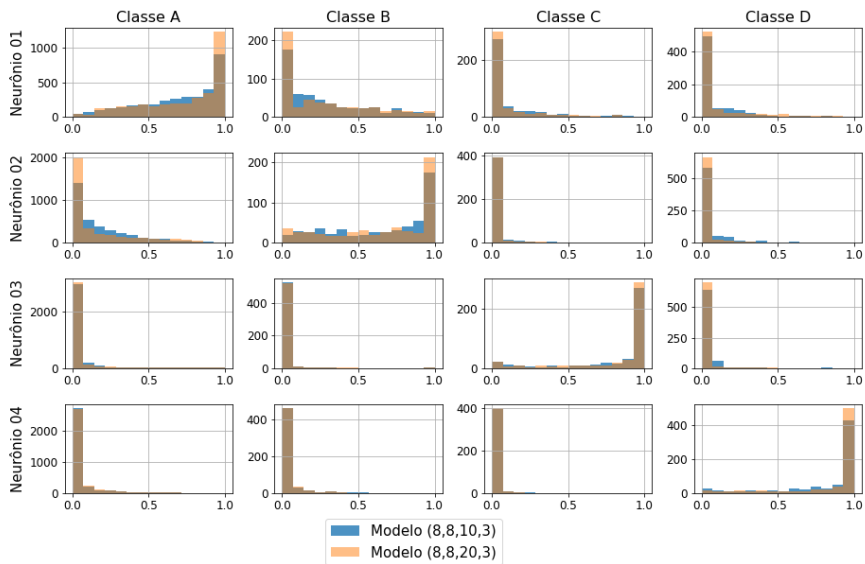
Figura C.40: Histogramas da saída dos modelos LOFAR CNN (*folds* 3 e 4).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 5



(a) *fold-5*

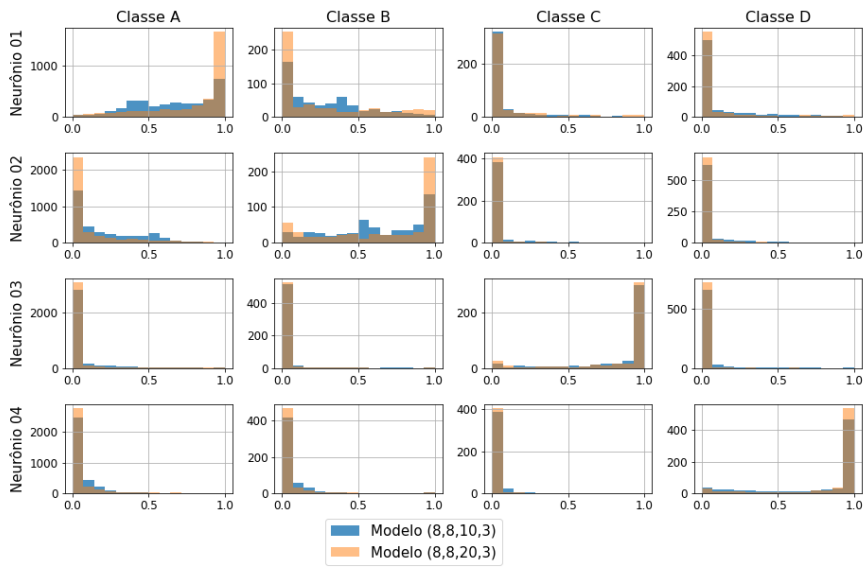
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 6



(b) *fold-6*

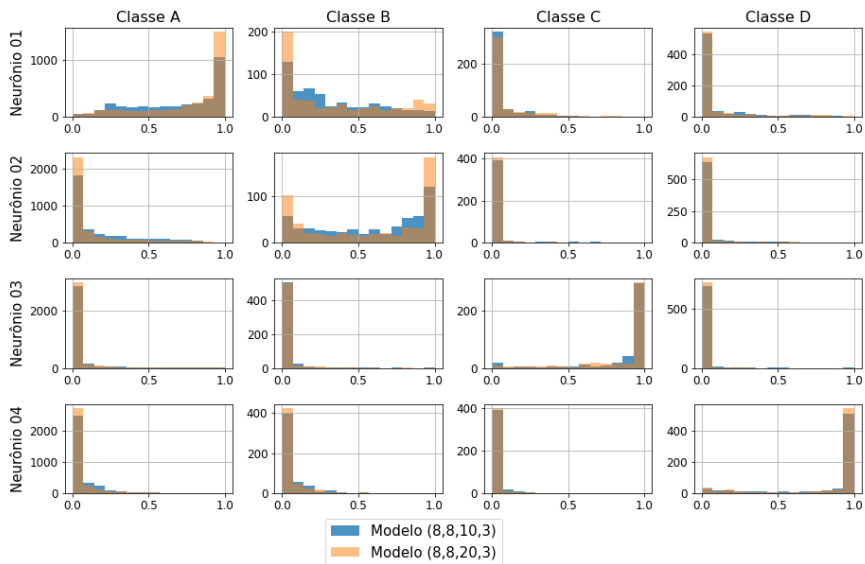
Figura C.41: Histogramas da saída dos modelos LOFAR CNN (*folds* 5 e 6).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 7



(a) *fold-7*

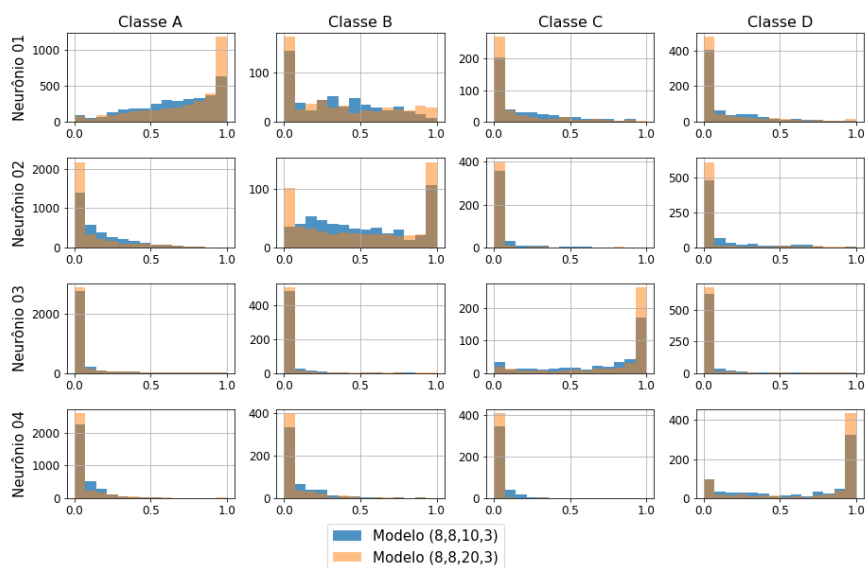
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 8



(b) *fold-8*

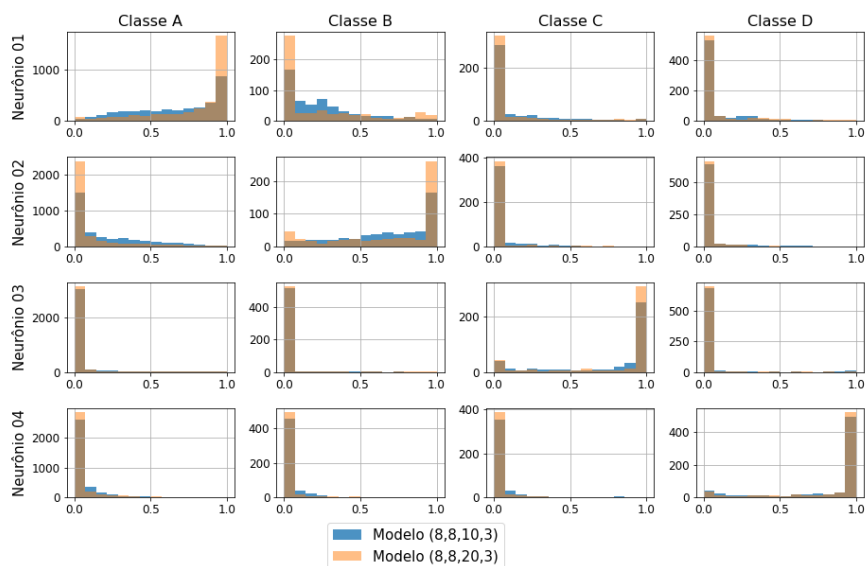
Figura C.42: Histogramas da saída dos modelos LOFAR CNN (*folds* 7 e 8).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 9



(a) *fold-9*

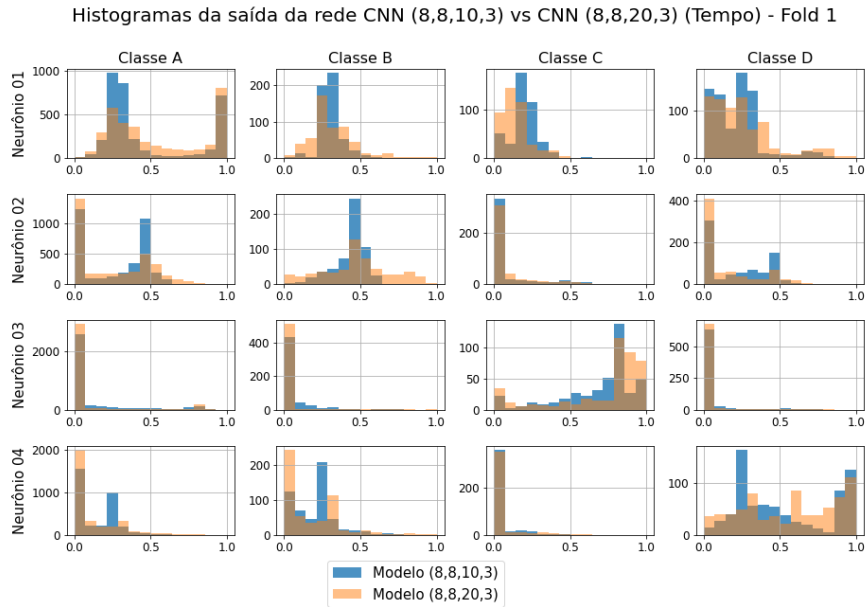
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 10



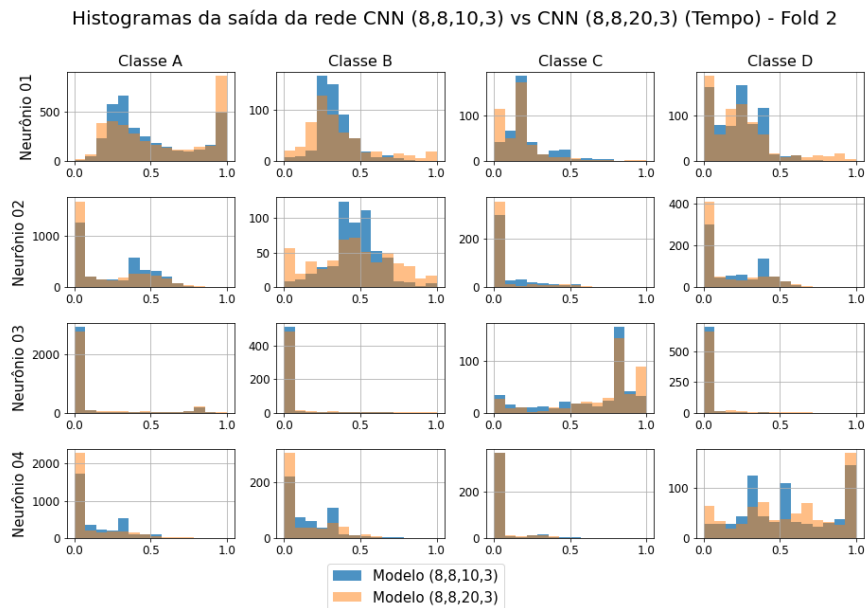
(b) *fold-10*

Figura C.43: Histogramas da saída dos modelos LOFAR CNN (*folders* 9 e 10).

As Figuras de C.44 a C.48 mostram os histogramas da saída dos 10 *fold*s dos modelos temporais CNN com 10 e 20 neurônios na camada *fully connected*, a extração de características é feita por duas camadas convolucionais de 8 filtros cada e *kernel* 3.



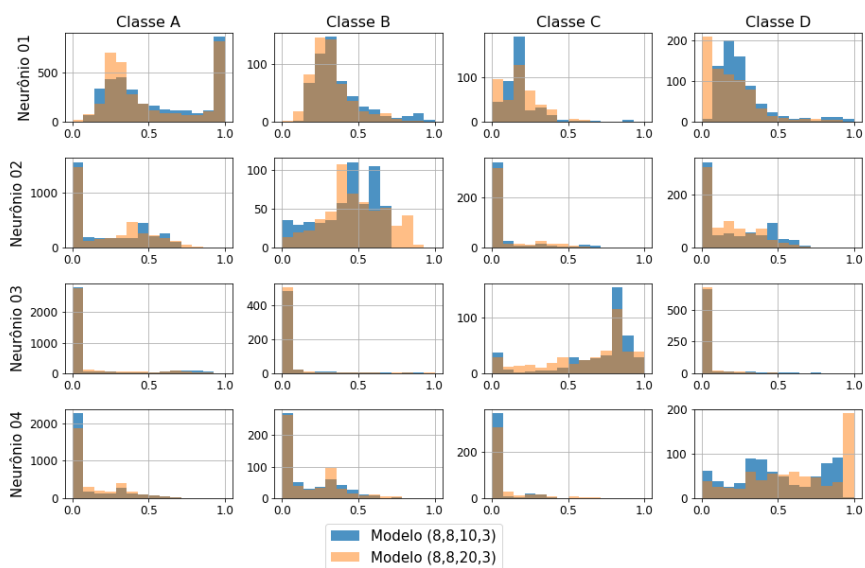
(a) *fold-1*



(b) *fold-2*

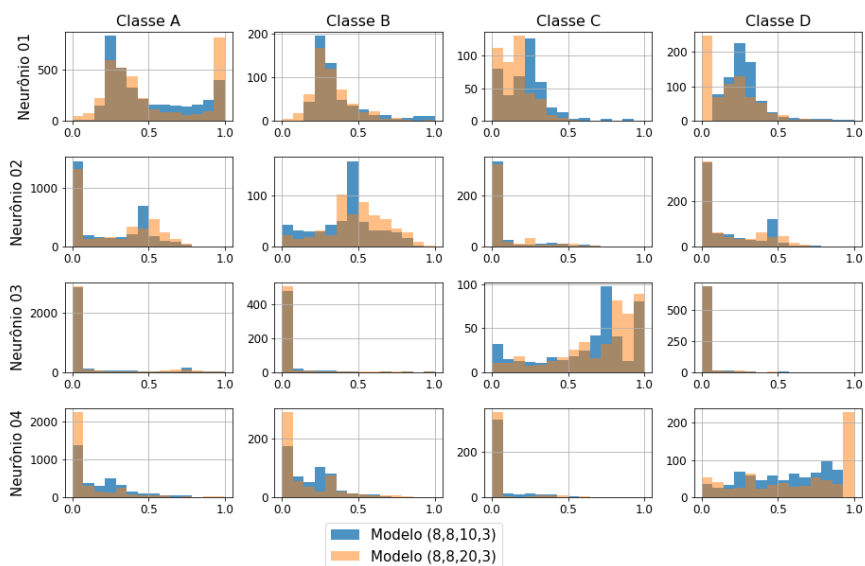
Figura C.44: Histogramas da saída dos modelos temporais CNN (*fold*s 1 e 2).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 3



(a) *fold-3*

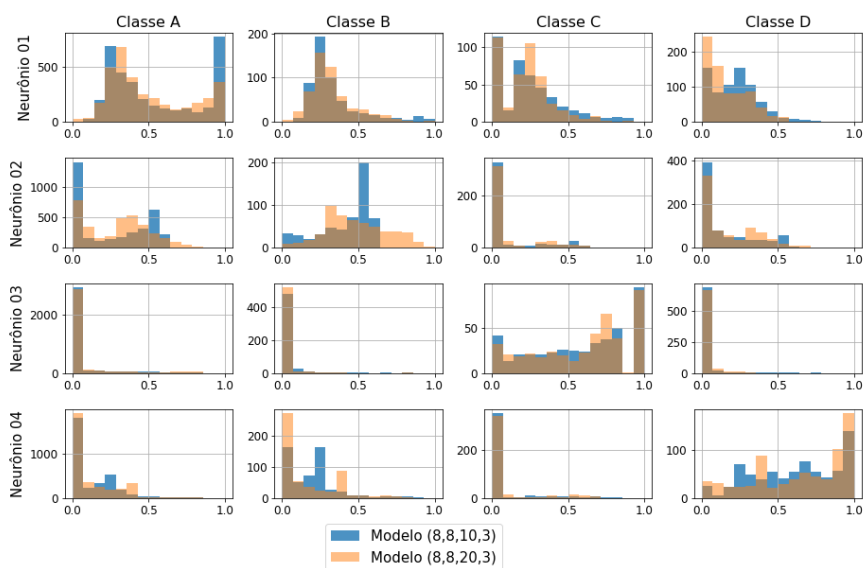
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 4



(b) *fold-4*

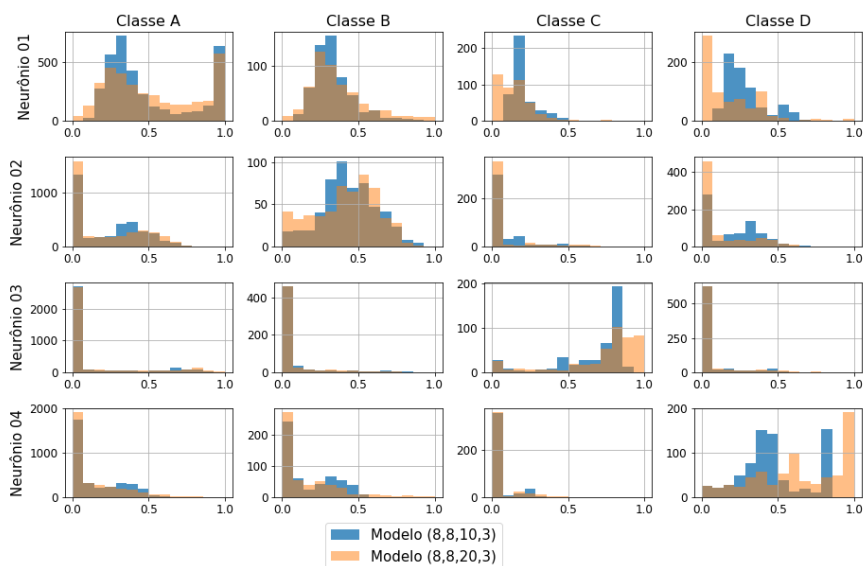
Figura C.45: Histogramas da saída dos modelos temporais CNN (*folds* 3 e 4).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 5



(a) *fold-5*

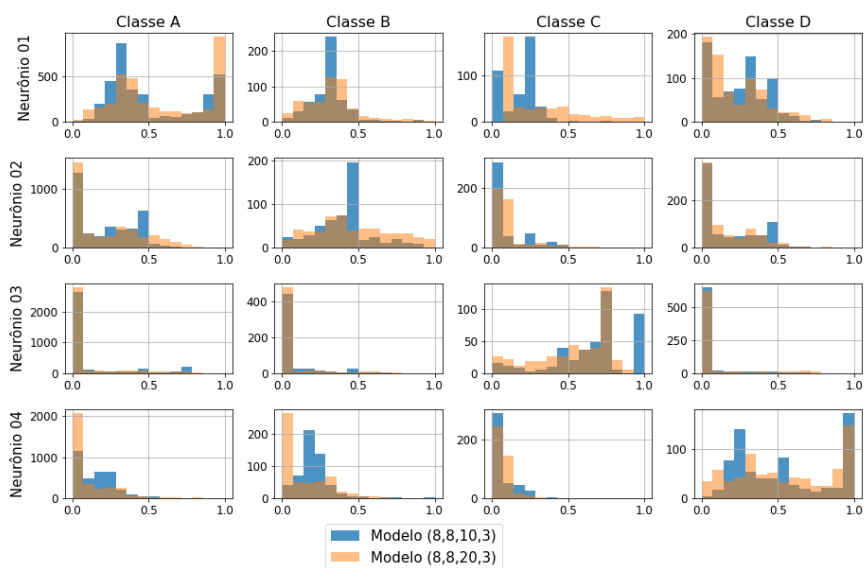
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 6



(b) *fold-6*

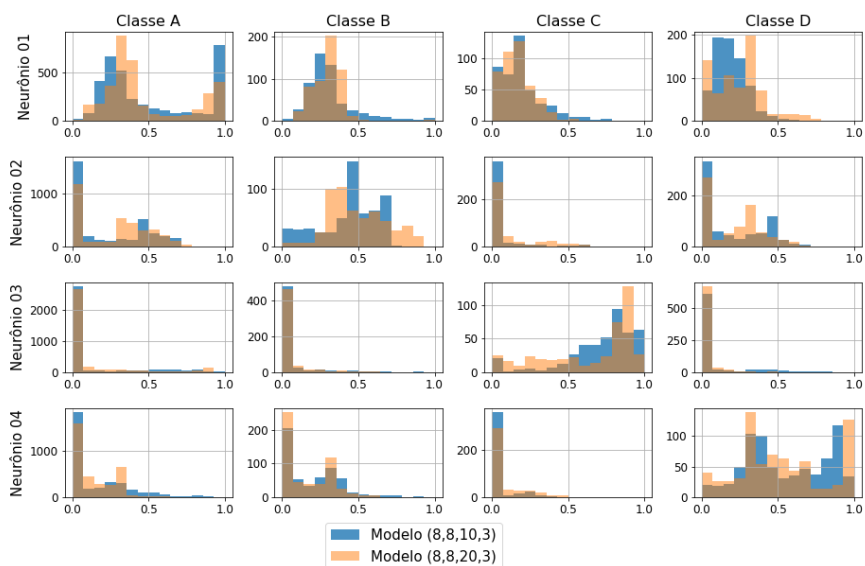
Figura C.46: Histogramas da saída dos modelos temporais CNN (*folds* 5 e 6).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 7



(a) *fold-7*

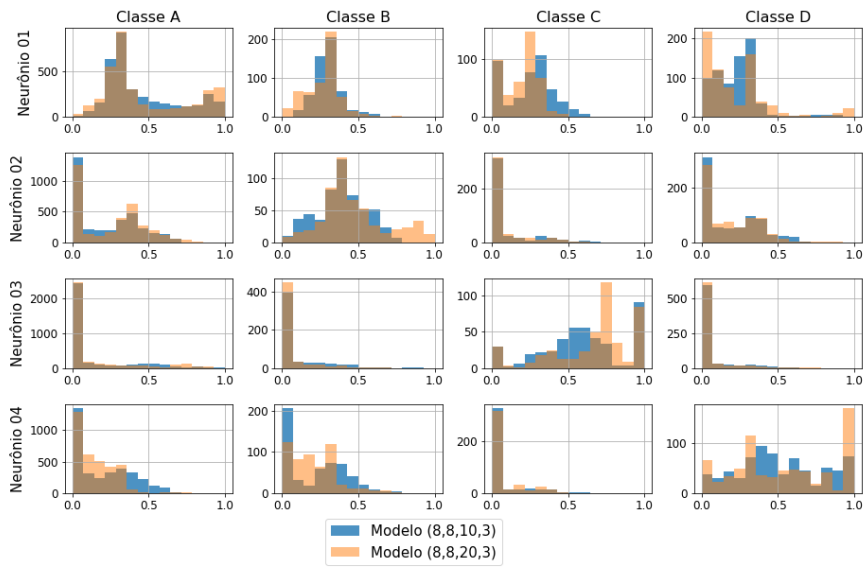
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 8



(b) *fold-8*

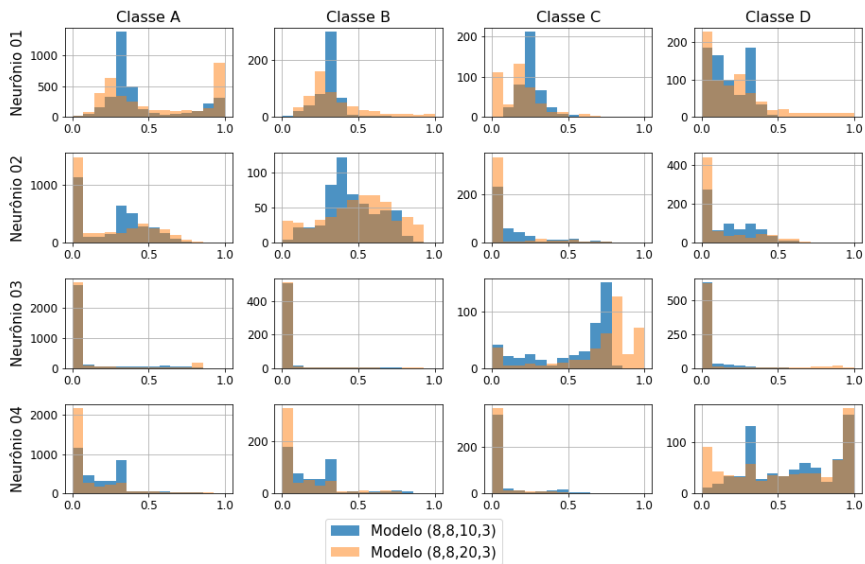
Figura C.47: Histogramas da saída dos modelos temporais CNN (*folds* 7 e 8).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 9



(a) *fold-9*

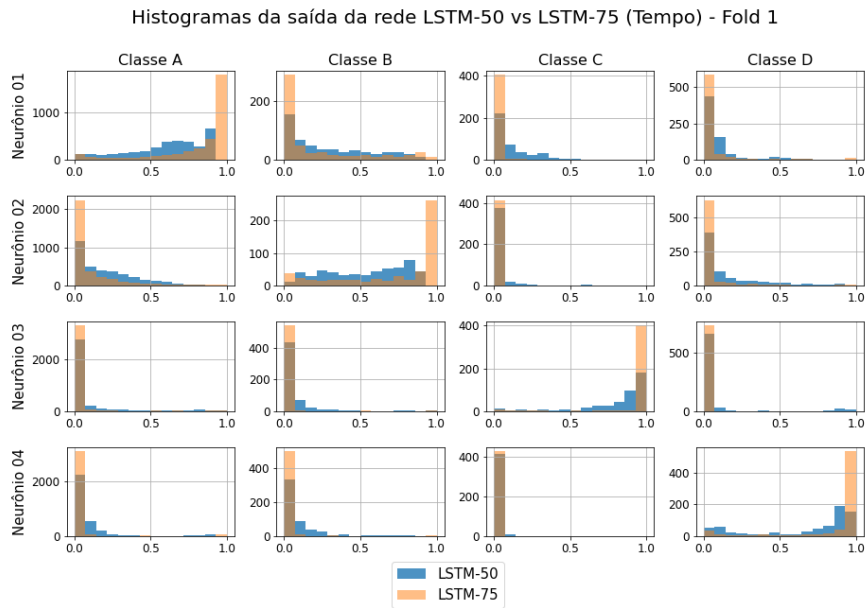
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 10



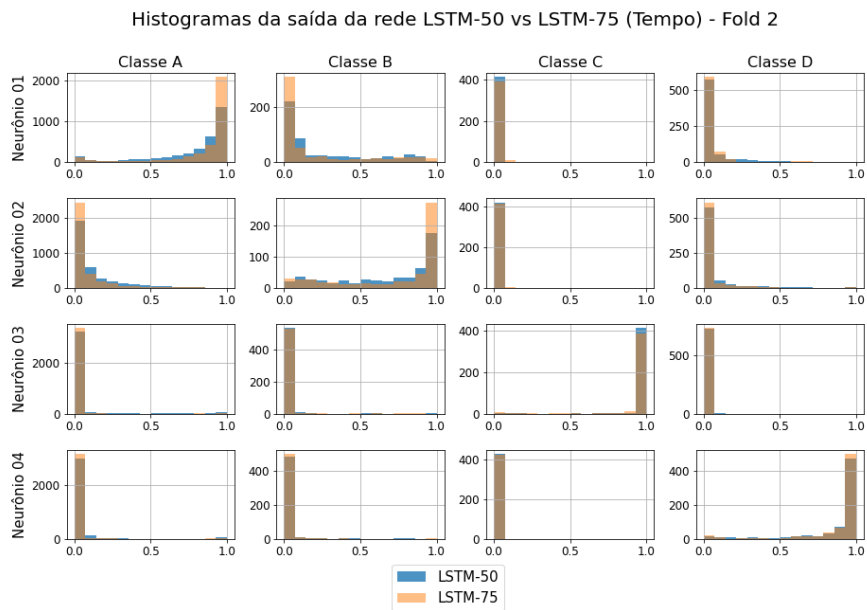
(b) *fold-10*

Figura C.48: Histogramas da saída dos modelos temporais CNN (*folds* 9 e 10).

As Figuras de C.49 a C.53 mostram os histogramas da saída dos 10 *folds* do modelo temporal LSTM com 50 células na camada oculta.



(a) *fold-1*



(b) *fold-2*

Figura C.49: Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (*folds* 1 e 2).

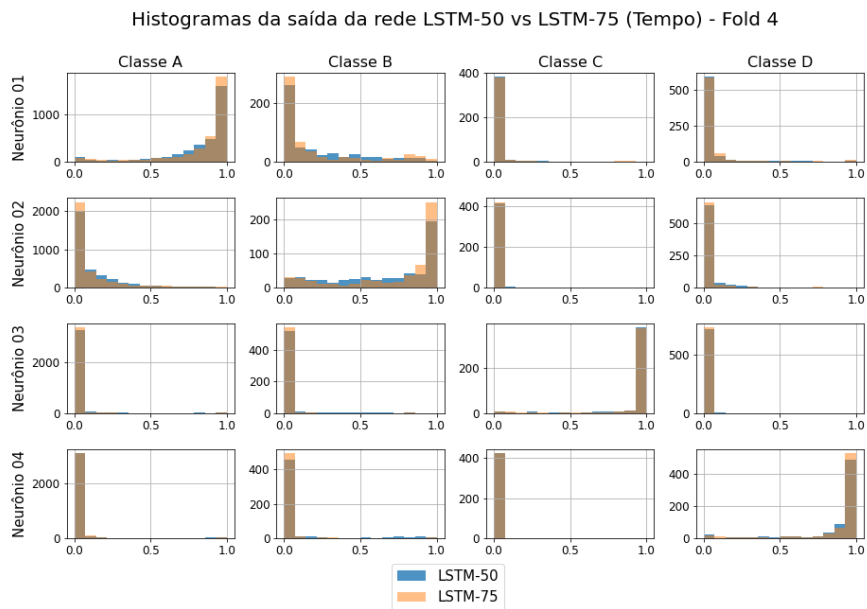
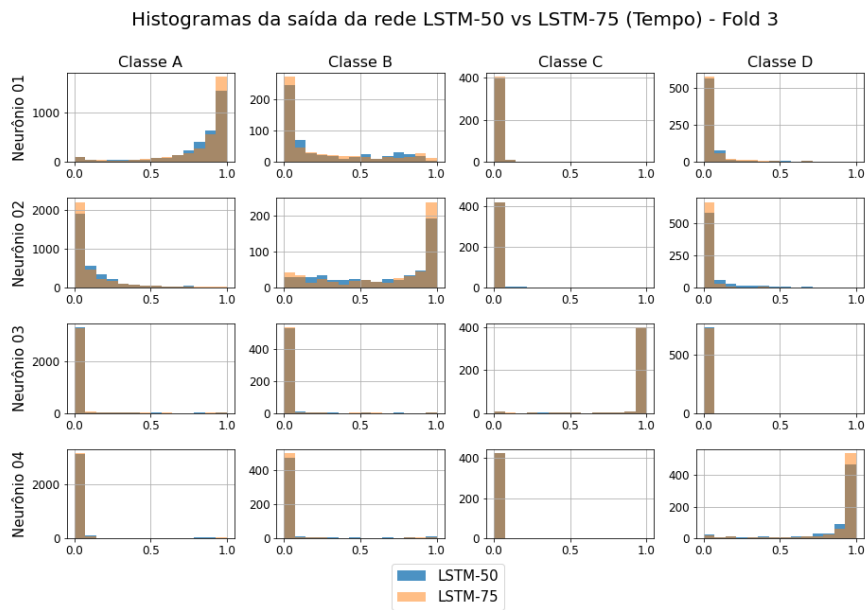


Figura C.50: Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (*fold*s 3 e 4).

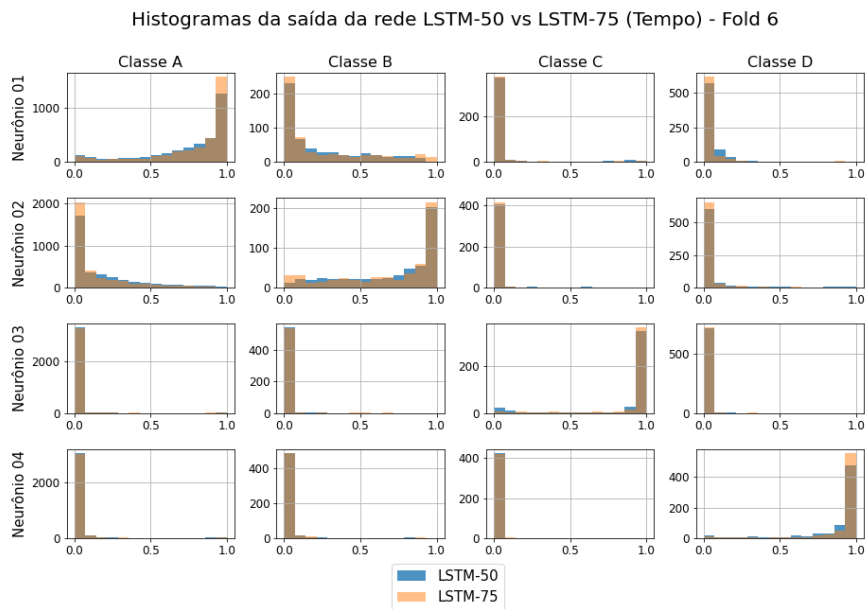
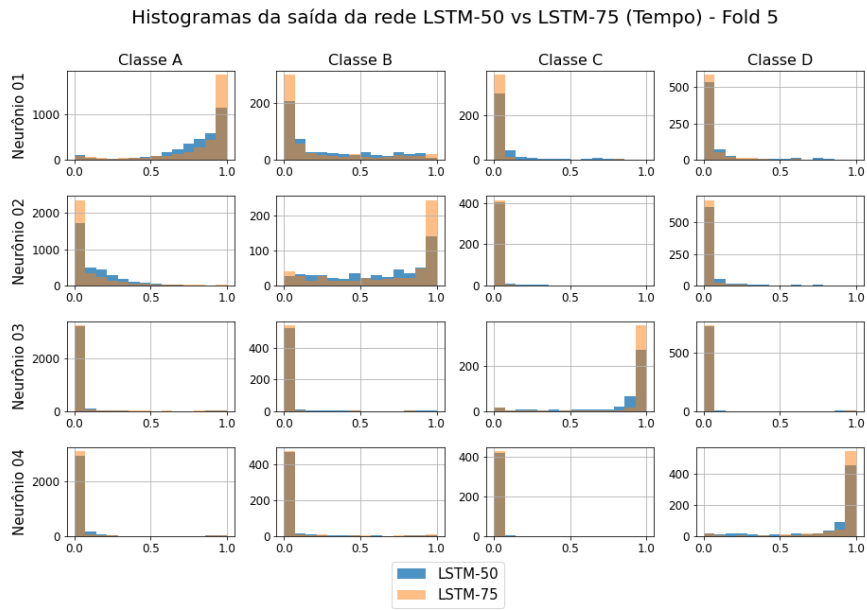


Figura C.51: Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (*fold*s 5 e 6).

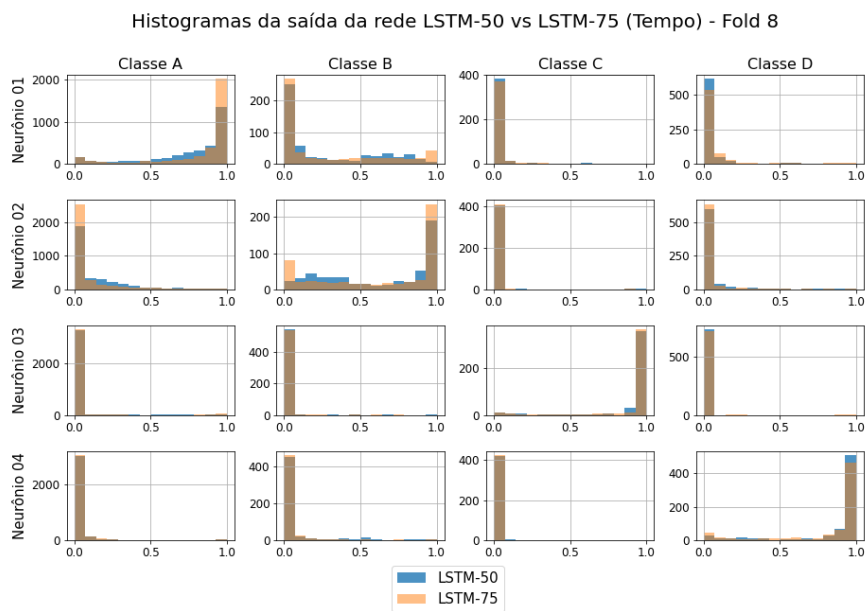
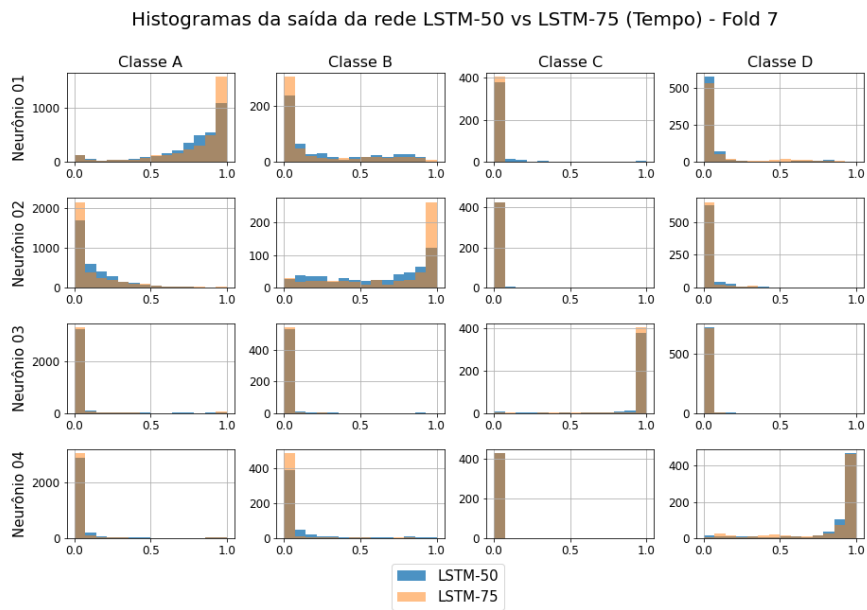


Figura C.52: Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (*folds* 7 e 8).

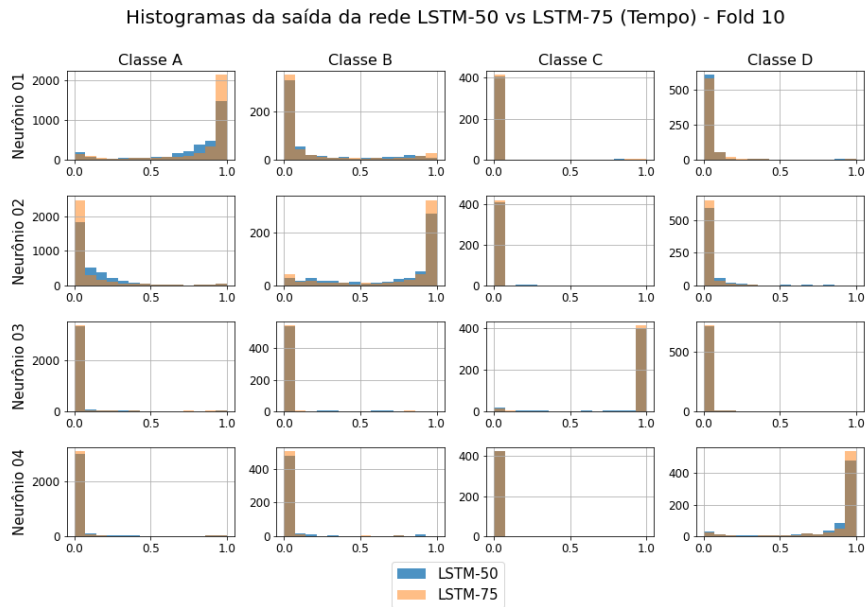
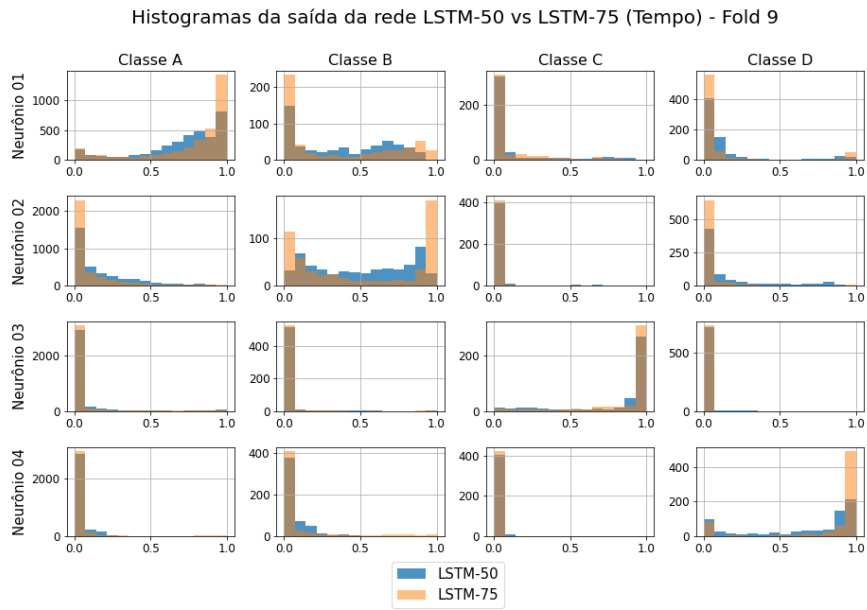
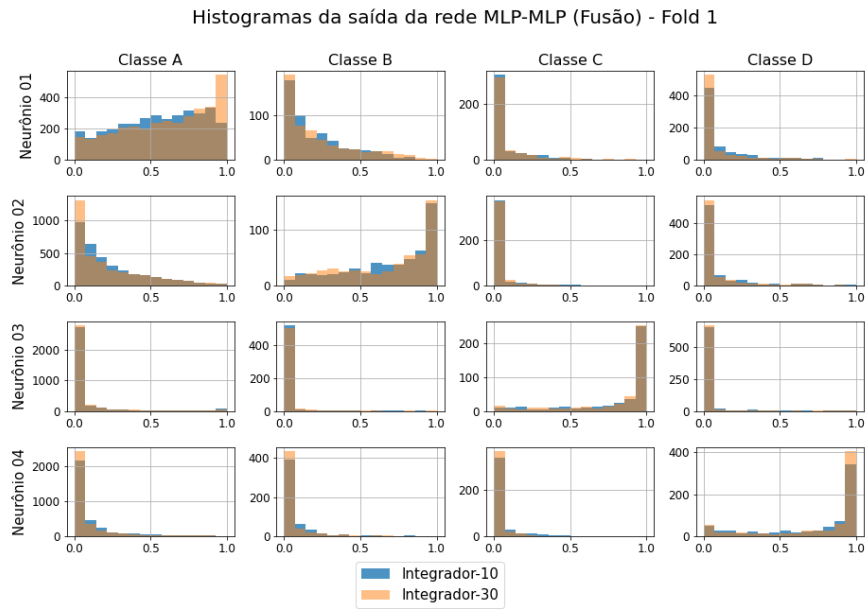


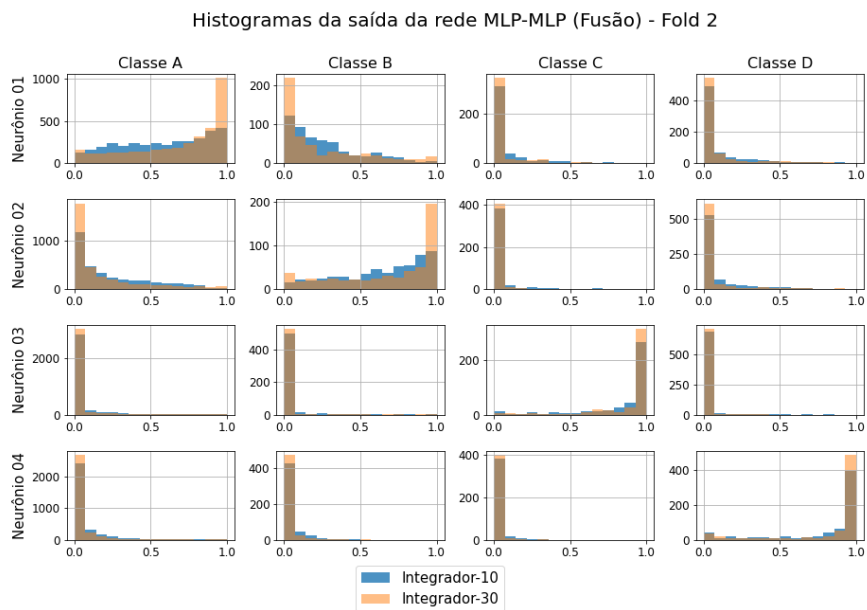
Figura C.53: Histogramas da saída dos modelos temporais LSTM-50 e LSTM-75 (*folds* 9 e 10).

Fusão dos modelos

As Figuras de C.54 a C.58 mostram os histogramas da saída dos 10 *folds* da fusão dos modelos MLP com 10 e 30 neurônios da camada oculta da rede de fusão.



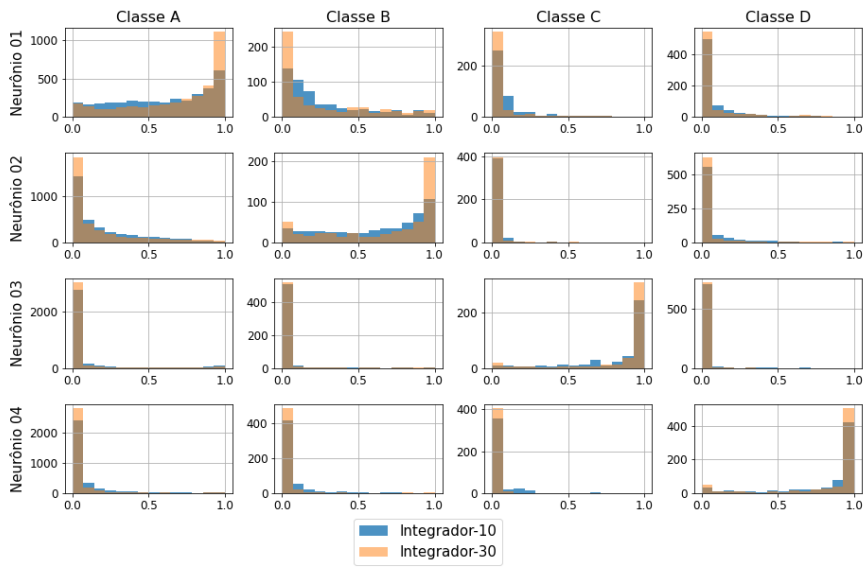
(a) *fold-1*



(b) *fold-2*

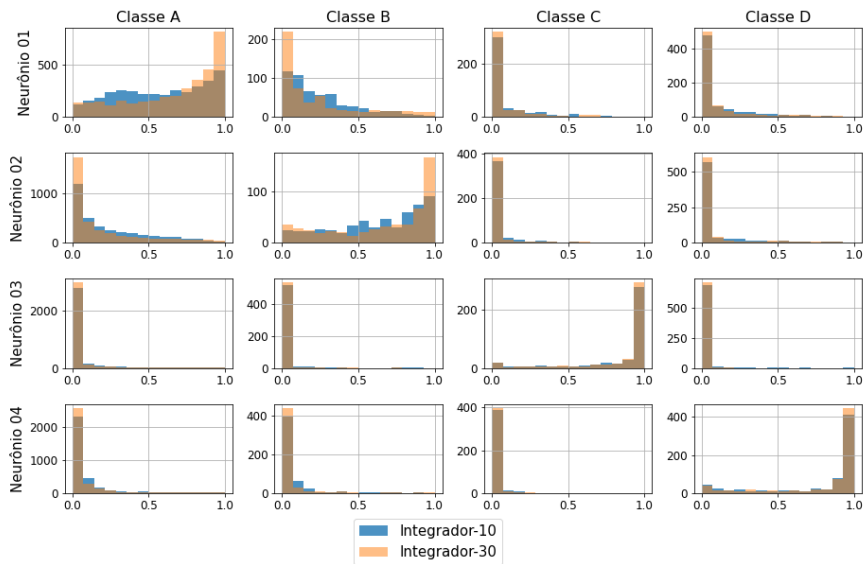
Figura C.54: Histogramas da saída da fusão dos modelos MLP (*folds* 1 e 2).

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 3



(a) *fold-3*

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 4



(b) *fold-4*

Figura C.55: Histogramas da saída da fusão dos modelos MLP (*folds* 3 e 4).

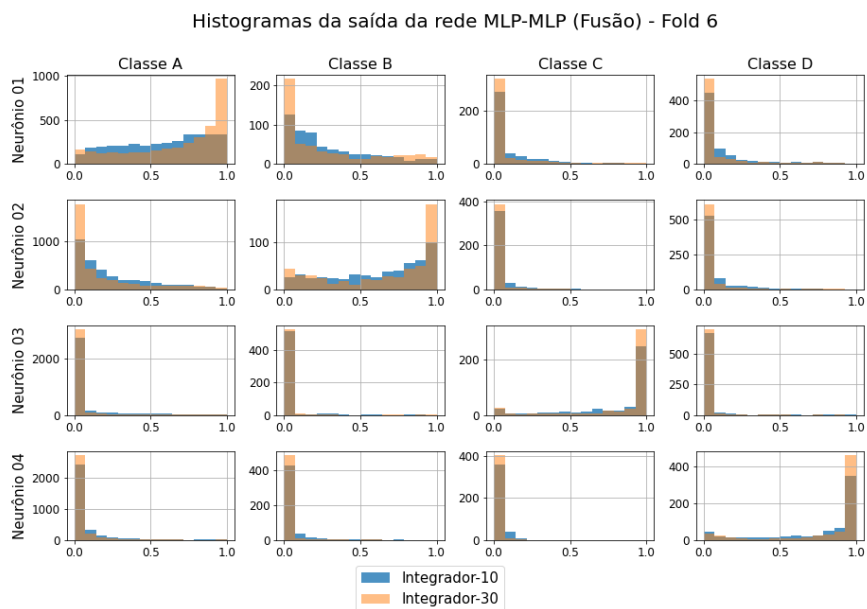
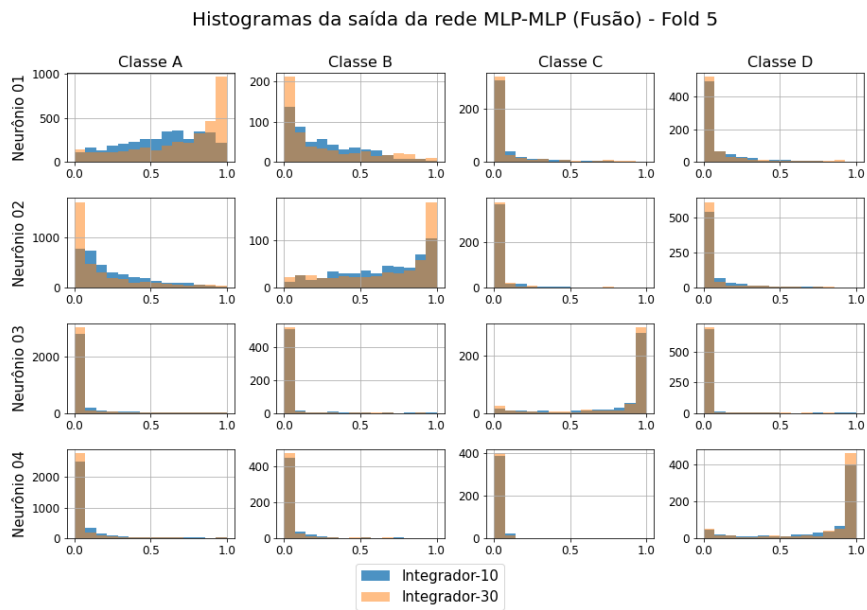
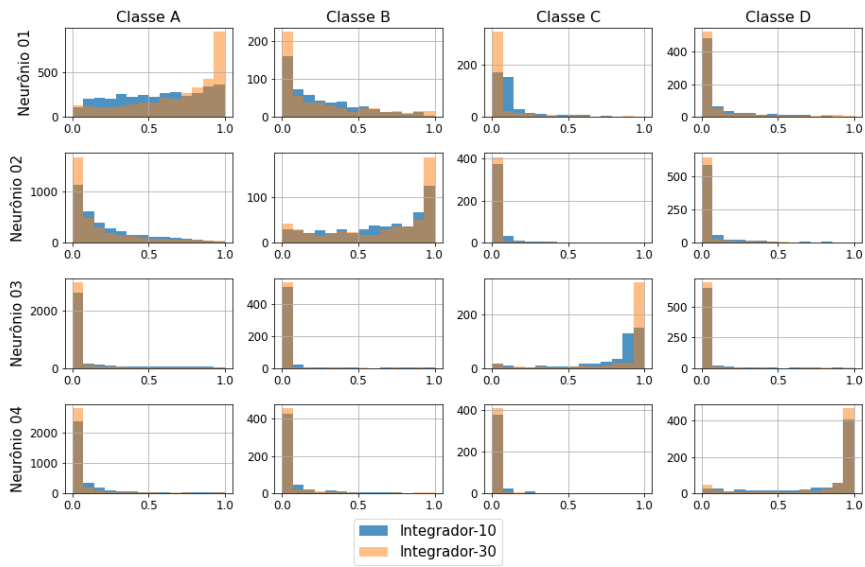


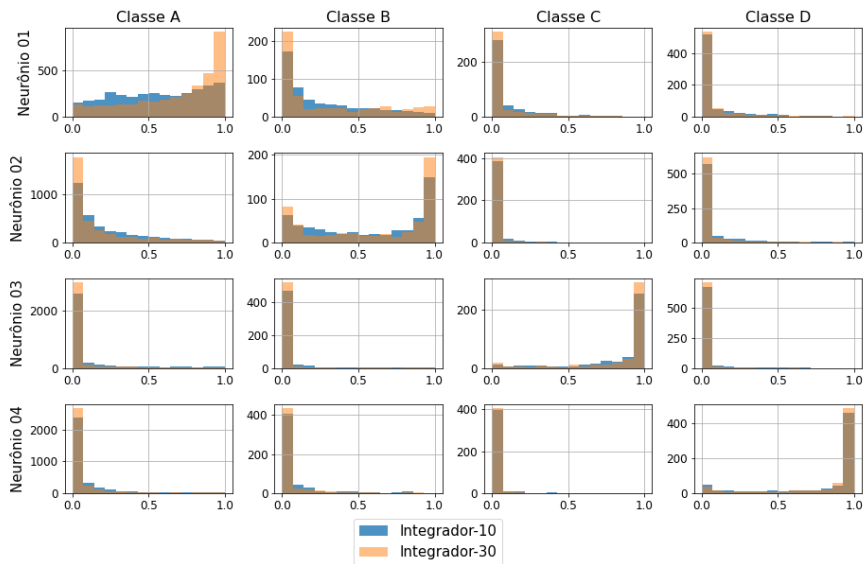
Figura C.56: Histogramas da saída da fusão dos modelos MLP (*folds* 5 e 6).

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 7



(a) *fold-7*

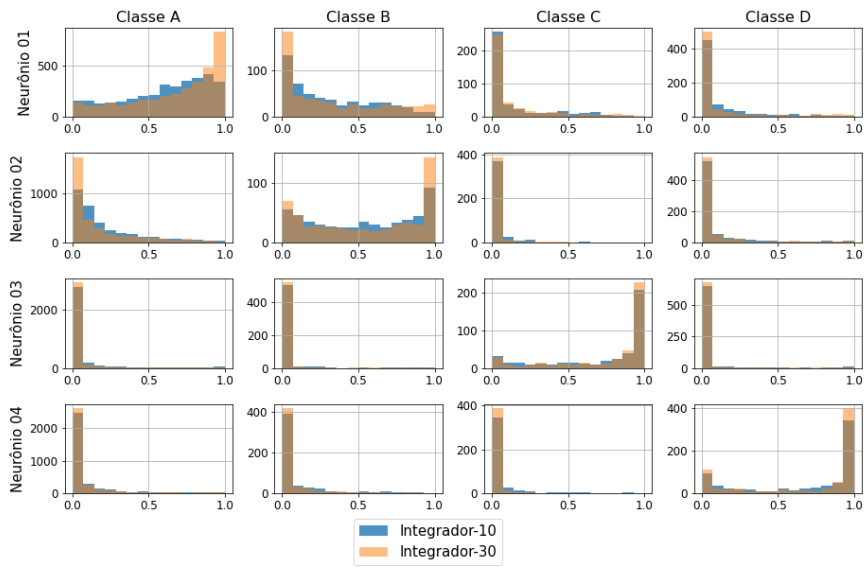
Histogramas da saída da rede MLP-MLP (Fusão) - Fold 8



(b) *fold-8*

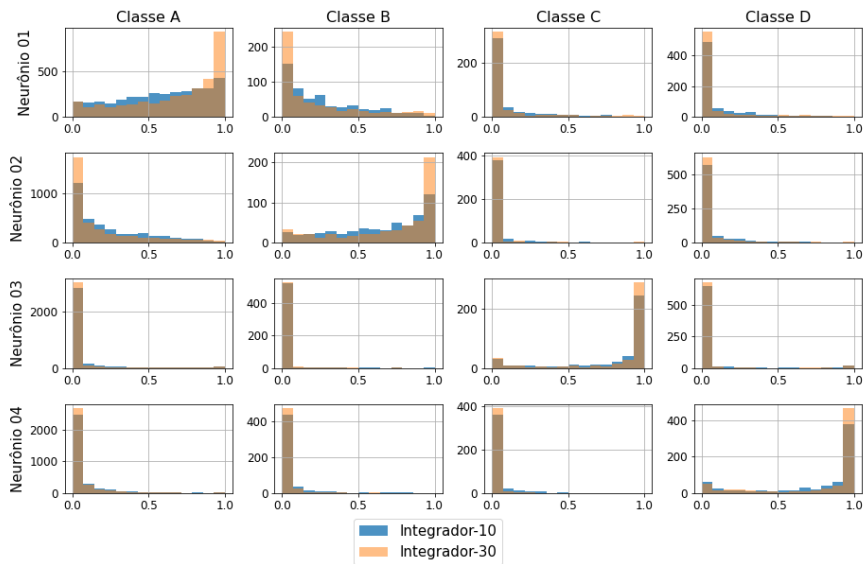
Figura C.57: Histogramas da saída da fusão dos modelos MLP (*folds* 7 e 8).

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 9



(a) *fold-9*

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 10



(b) *fold-10*

Figura C.58: Histogramas da saída da fusão dos modelos MLP (*folds* 9 e 10).

As Figuras de C.59 a C.63 mostram os histogramas da saída dos 10 *fold*s da fusão dos modelos CNN com e sem a rede de fusão.

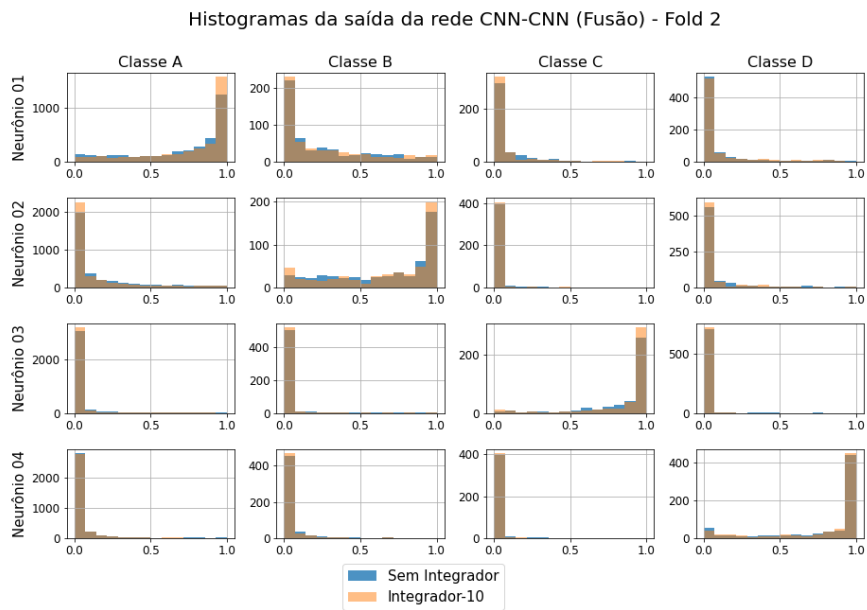
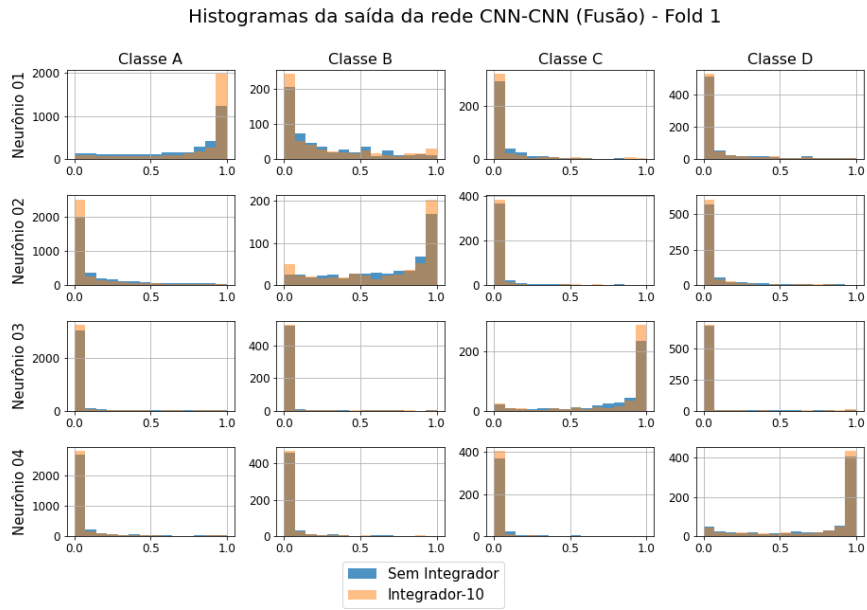
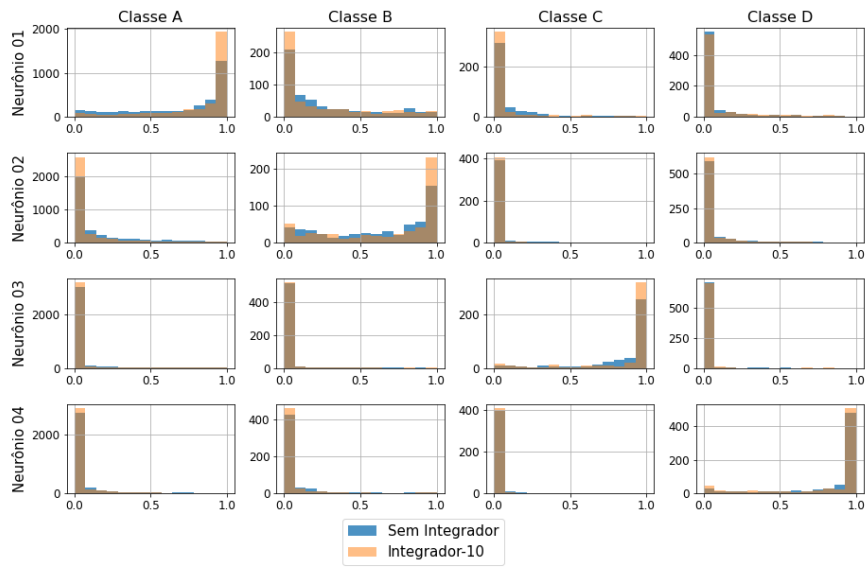


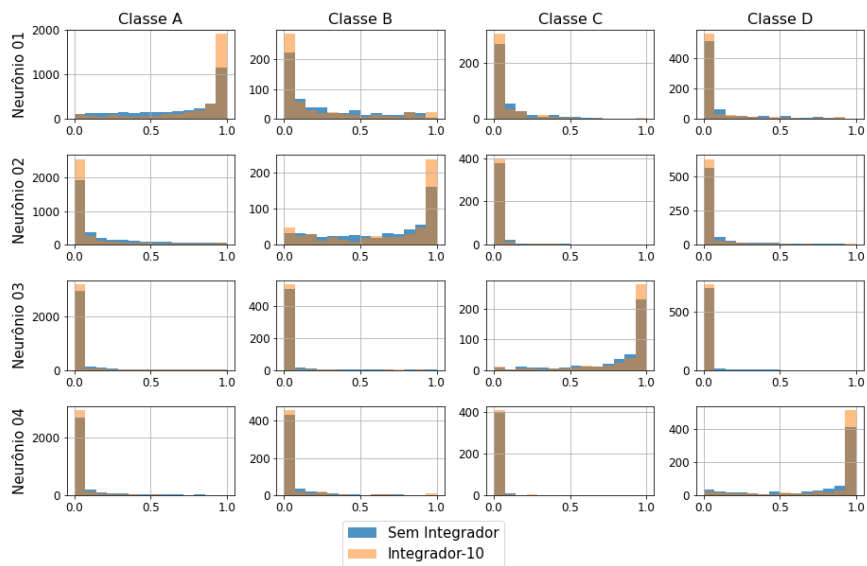
Figura C.59: Histogramas da saída do modelo CNN da fusão (*fold*s 1 e 2).

Histogramas da saída da rede CNN-CNN (Fusão) - Fold 3



(a) *fold-3*

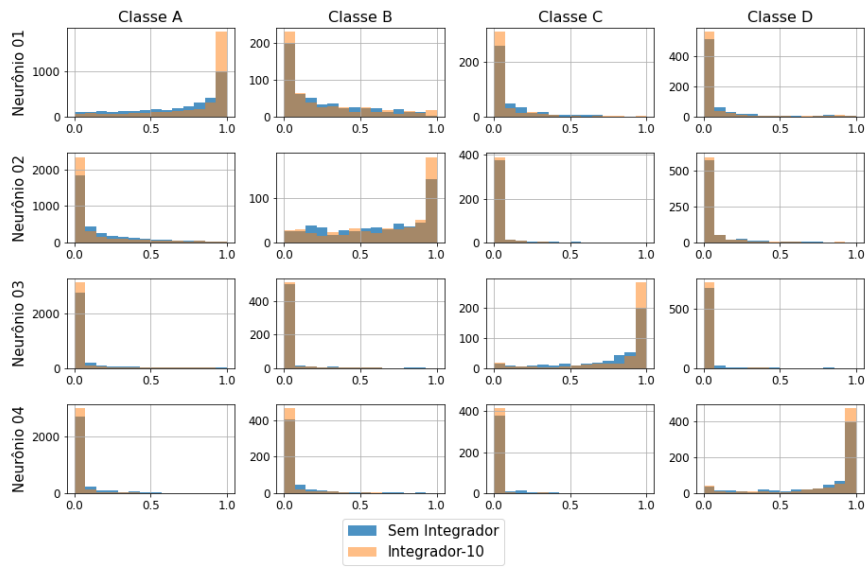
Histogramas da saída da rede CNN-CNN (Fusão) - Fold 4



(b) *fold-4*

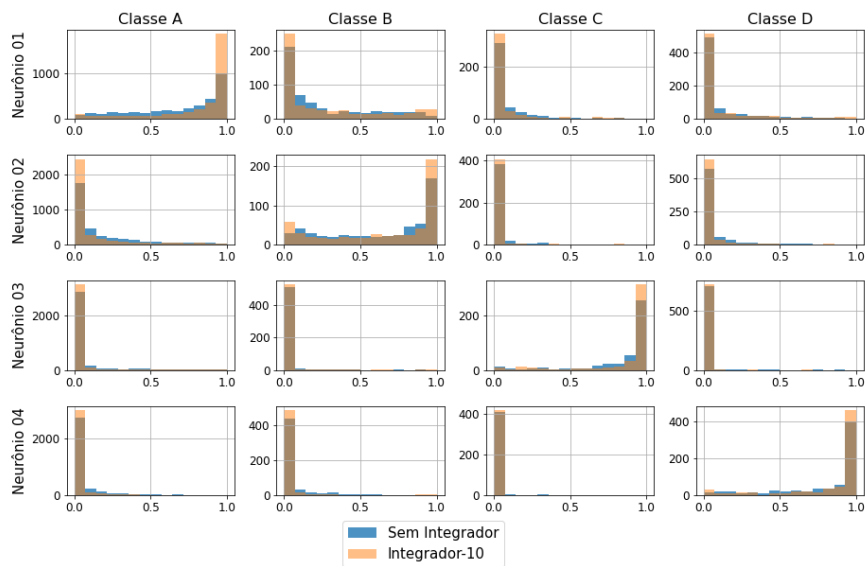
Figura C.60: Histogramas da saída do modelo CNN da fusão (*folds* 3 e 4).

Histogramas da saída da rede CNN-CNN (Fusão) - Fold 5



(a) *fold-5*

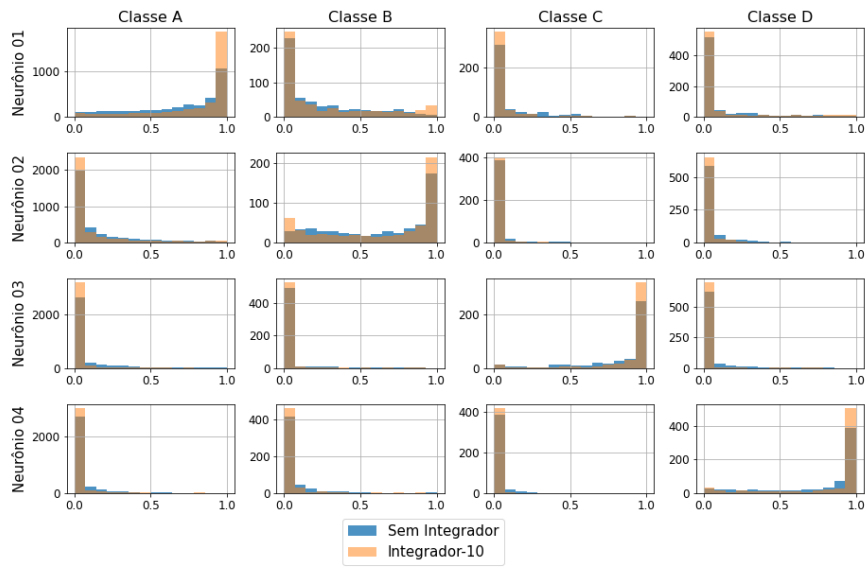
Histogramas da saída da rede CNN-CNN (Fusão) - Fold 6



(b) *fold-6*

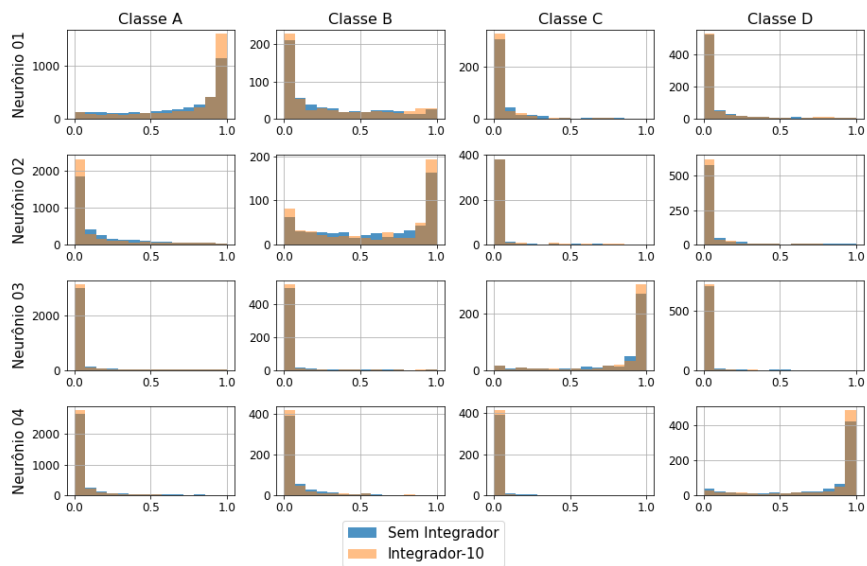
Figura C.61: Histogramas da saída do modelo CNN da fusão (*folds* 5 e 6).

Histogramas da saída da rede CNN-CNN (Fusão) - Fold 7



(a) *fold-7*

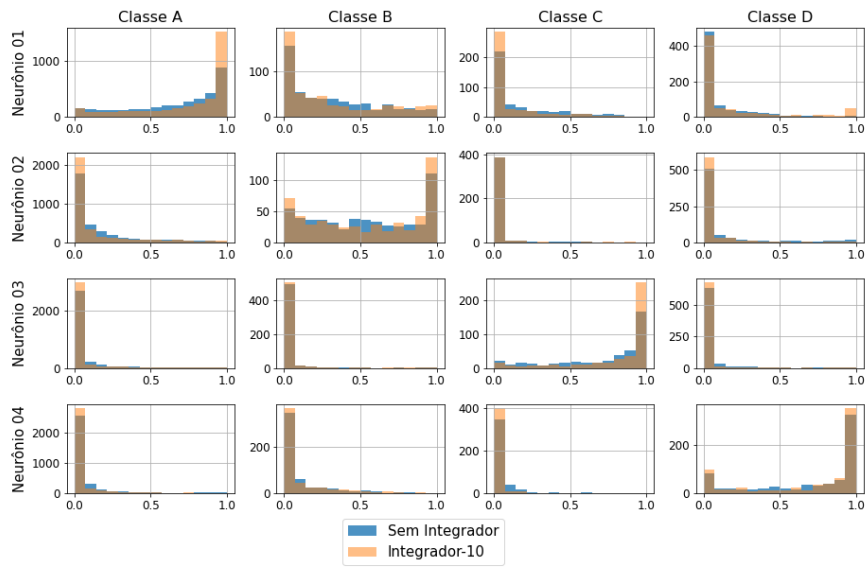
Histogramas da saída da rede CNN-CNN (Fusão) - Fold 8



(b) *fold-8*

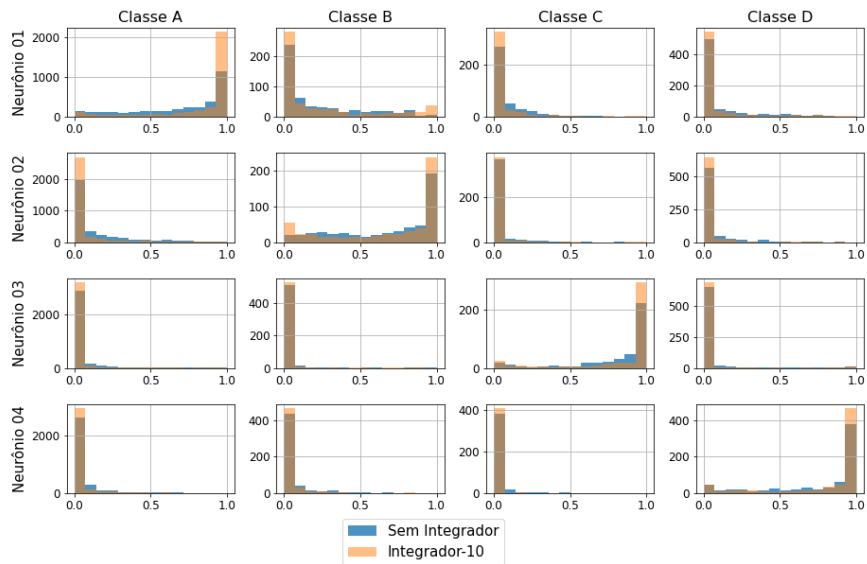
Figura C.62: Histogramas da saída do modelo CNN da fusão (*folds* 7 e 8).

Histogramas da saída da rede CNN-CNN (Fusão) - Fold 9



(a) *fold-9*

Histogramas da saída da rede CNN-CNN (Fusão) - Fold 10



(b) *fold-10*

Figura C.63: Histogramas da saída do modelo CNN da fusão (*folds* 9 e 10).

As Figuras de C.64 a C.68 mostram os histogramas da saída dos 10 *folds* da fusão dos modelos MLP e LSTM com e sem a rede de fusão.

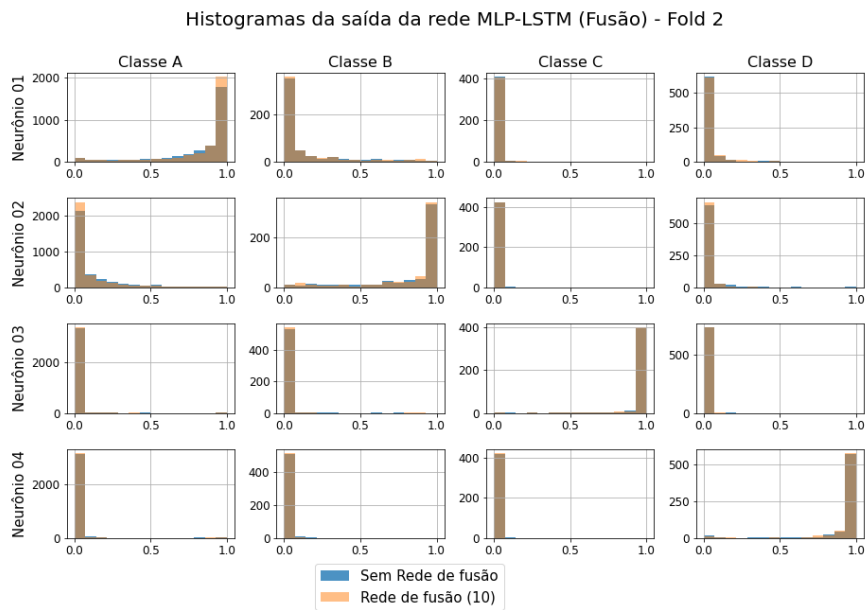
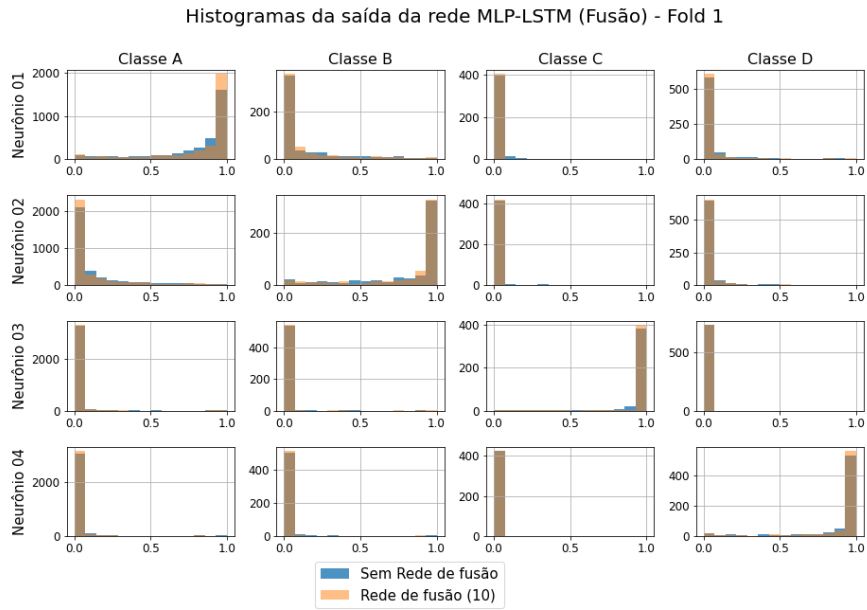


Figura C.64: Histogramas da saída da fusão do modelo MLP com LSTM (*folds* 1 e 2).

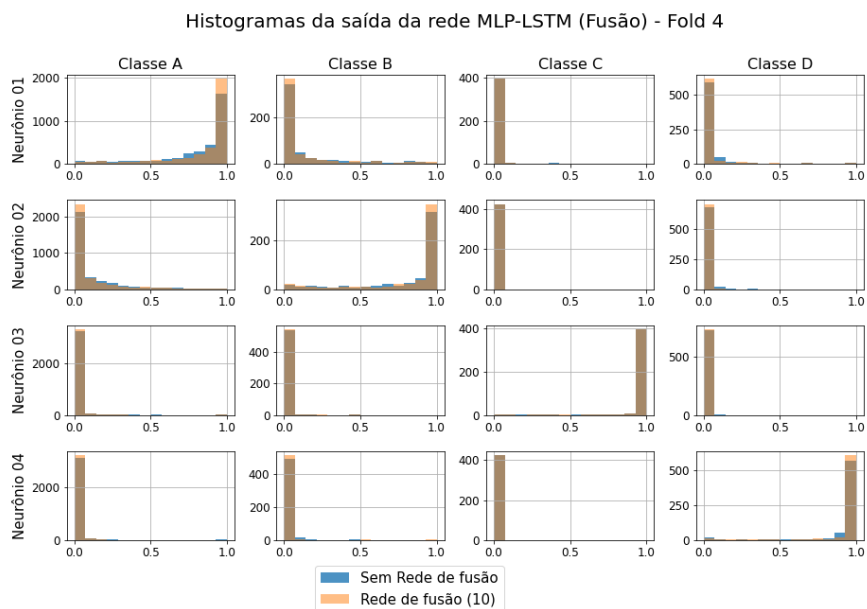
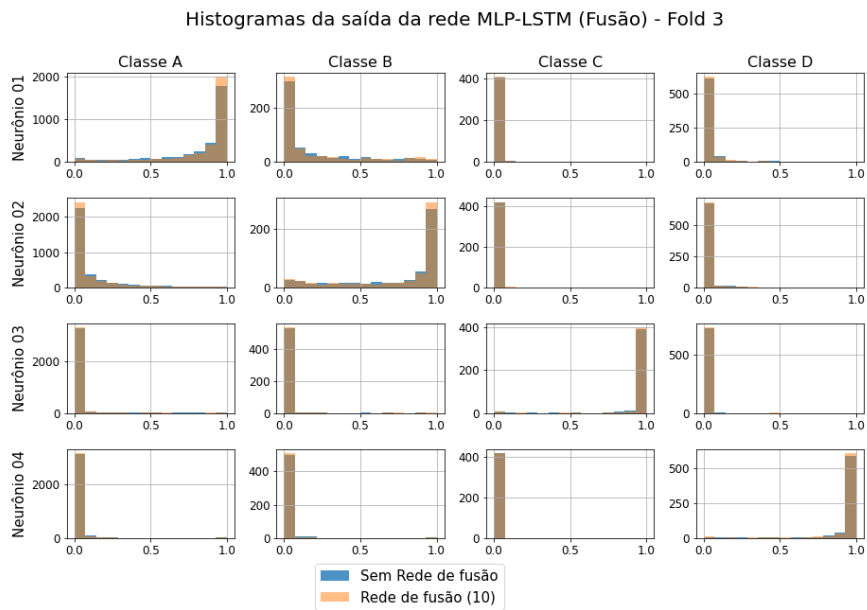


Figura C.65: Histogramas da saída da fusão do modelo MLP com LSTM (*folds* 3 e 4).

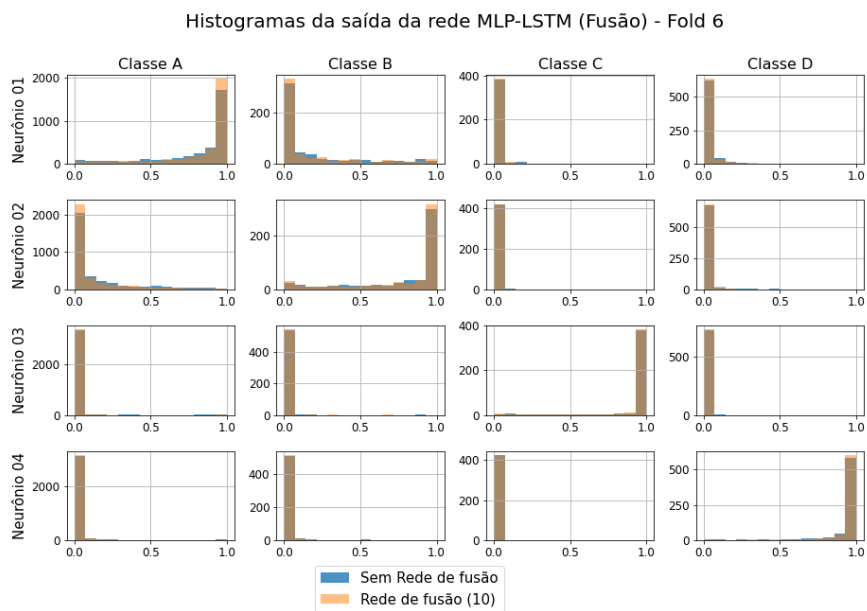
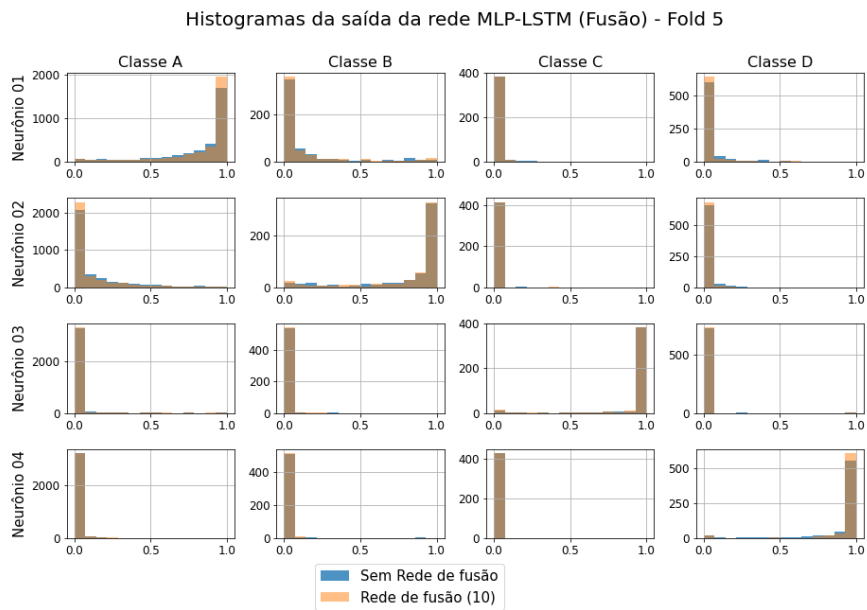


Figura C.66: Histogramas da saída da fusão do modelo MLP com LSTM (*folds* 5 e 6).

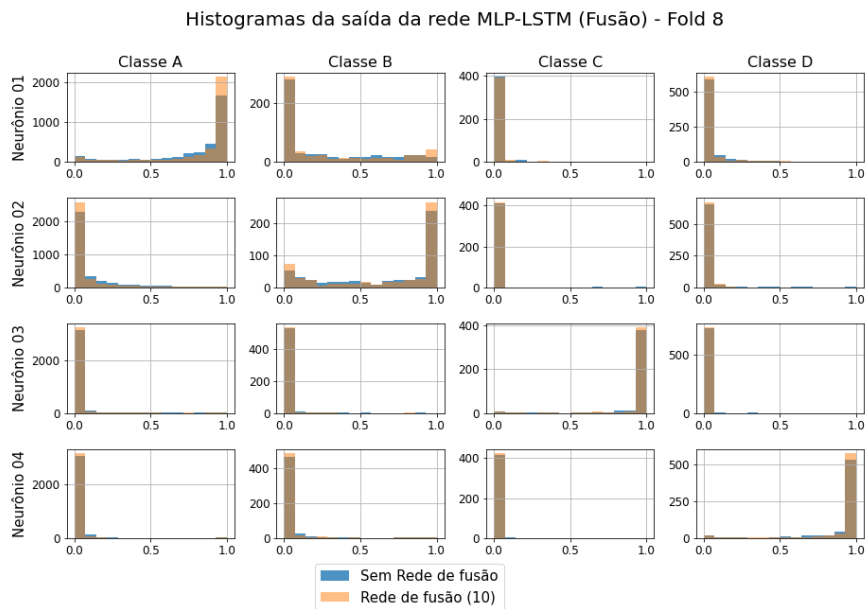
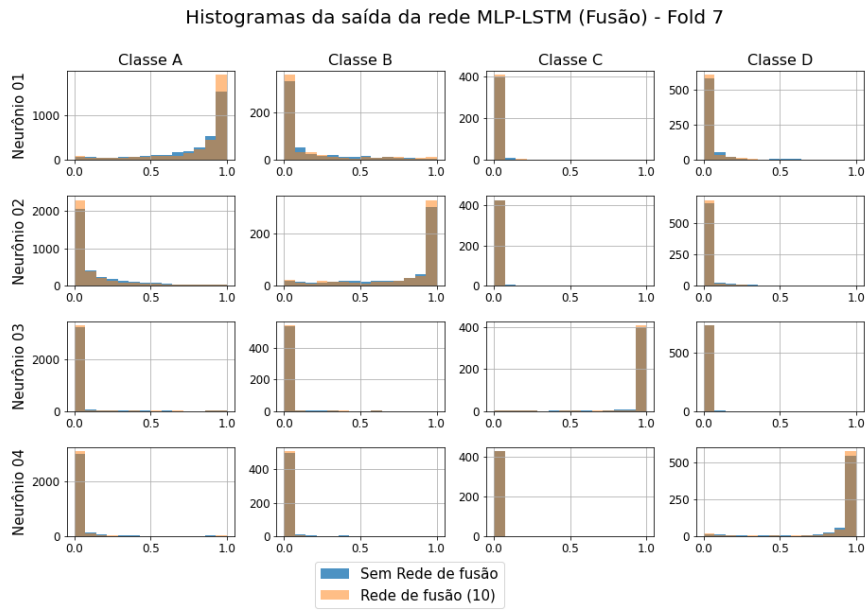


Figura C.67: Histogramas da saída da fusão do modelo MLP com LSTM (*folds* 7 e 8).

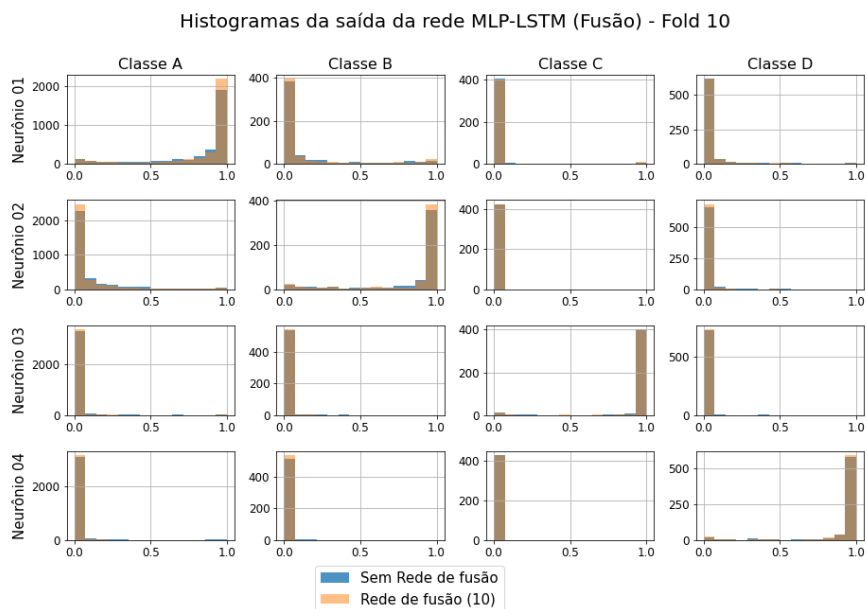
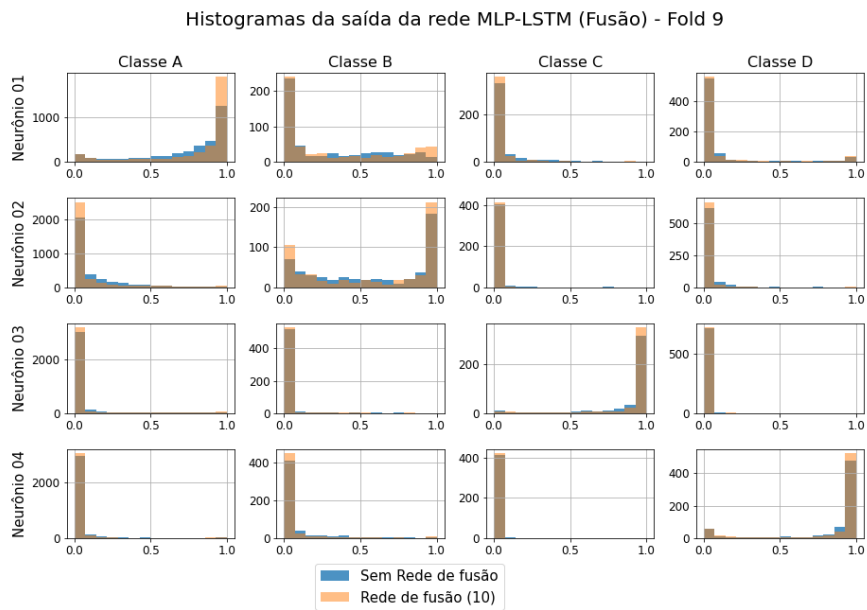


Figura C.68: Histogramas da saída da fusão do modelo MLP com LSTM (*folds* 9 e 10).

As Figuras de C.69 a C.73 mostram os histogramas da saída dos 10 *folds* da fusão dos modelos CNN e LSTM sem e com a rede de fusão de 10 neurônios.

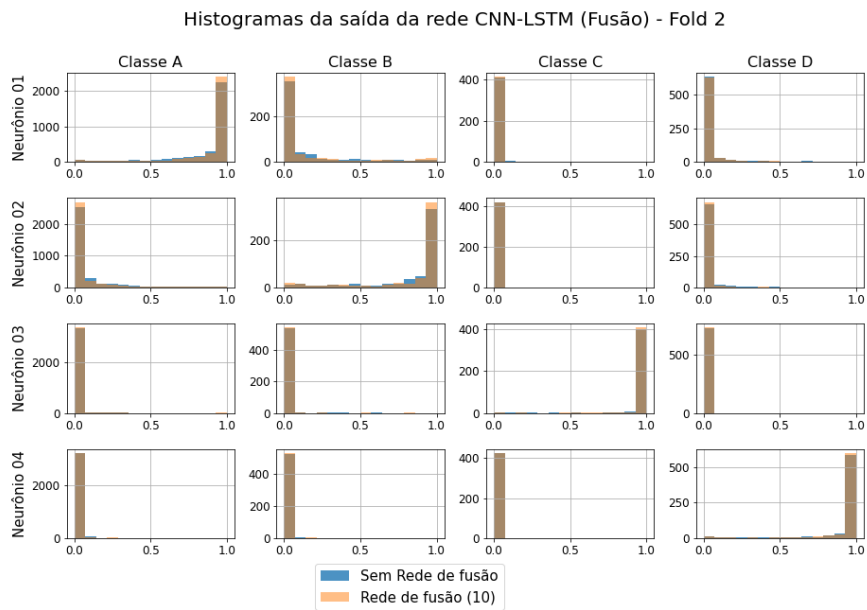
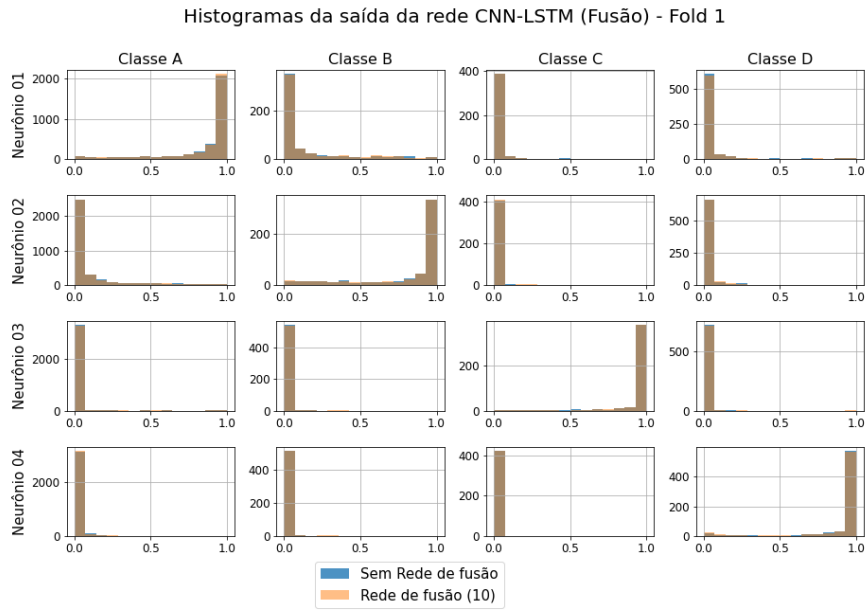


Figura C.69: Histogramas da saída da fusão do modelo CNN com LSTM (*folds* 1 e 2).

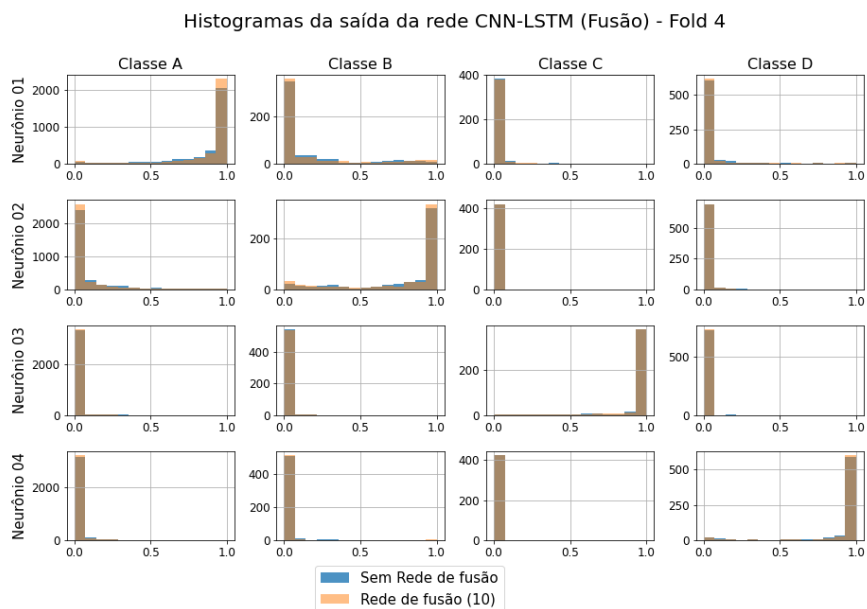
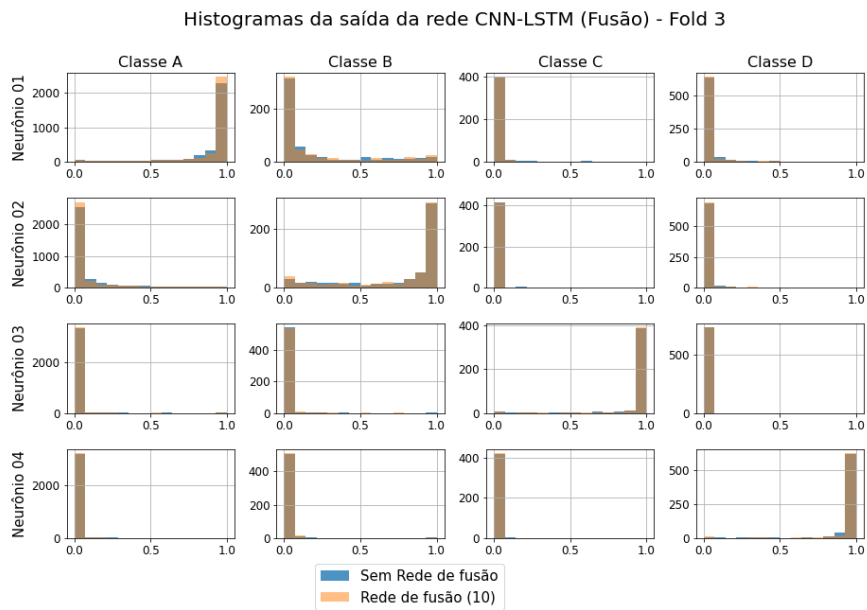


Figura C.70: Histogramas da saída da fusão do modelo CNN com LSTM (*folds* 3 e 4).

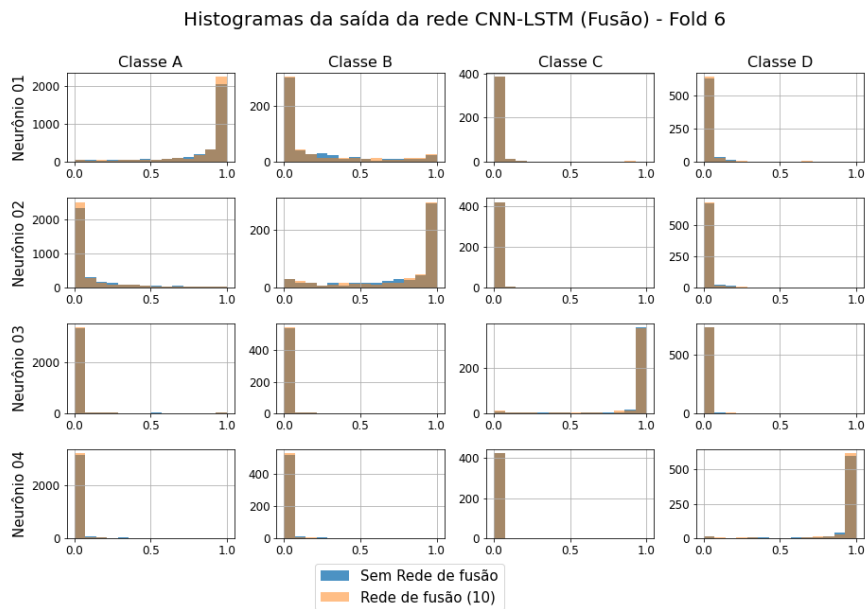
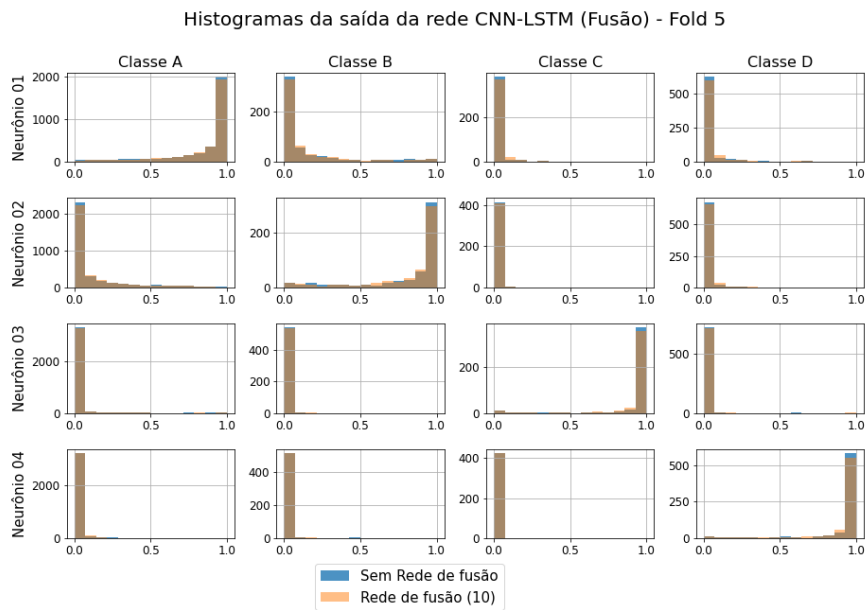


Figura C.71: Histogramas da saída da fusão do modelo CNN com LSTM (*folds* 5 e 6).

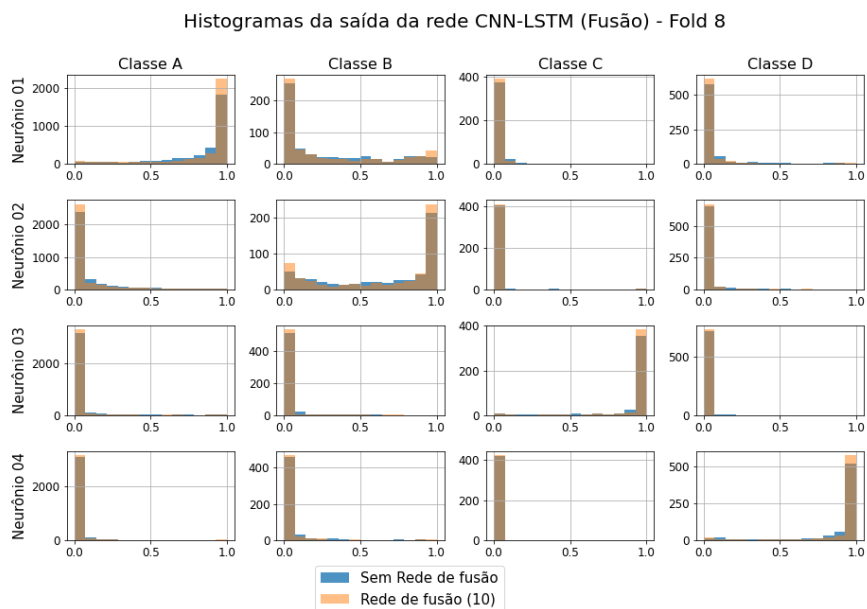
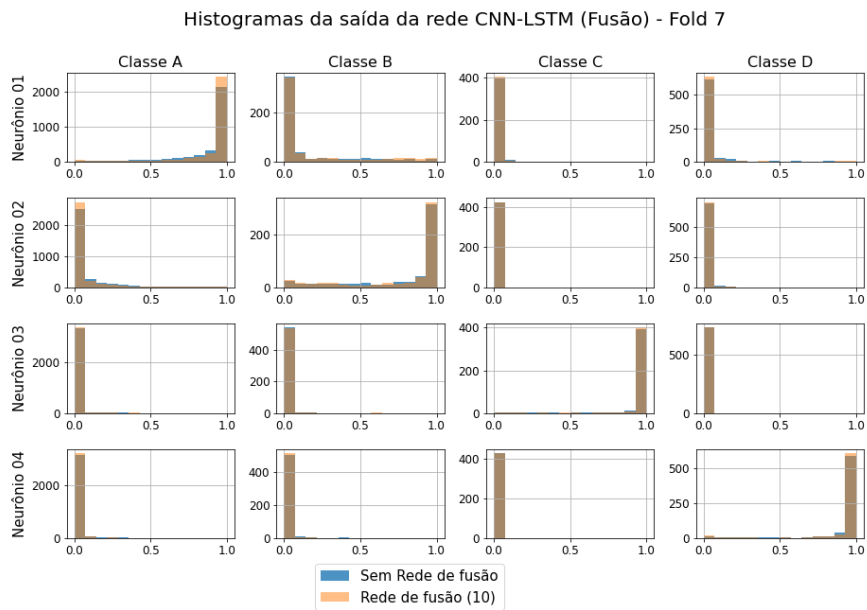


Figura C.72: Histogramas da saída da fusão do modelo CNN com LSTM (*folds 7 e 8*).

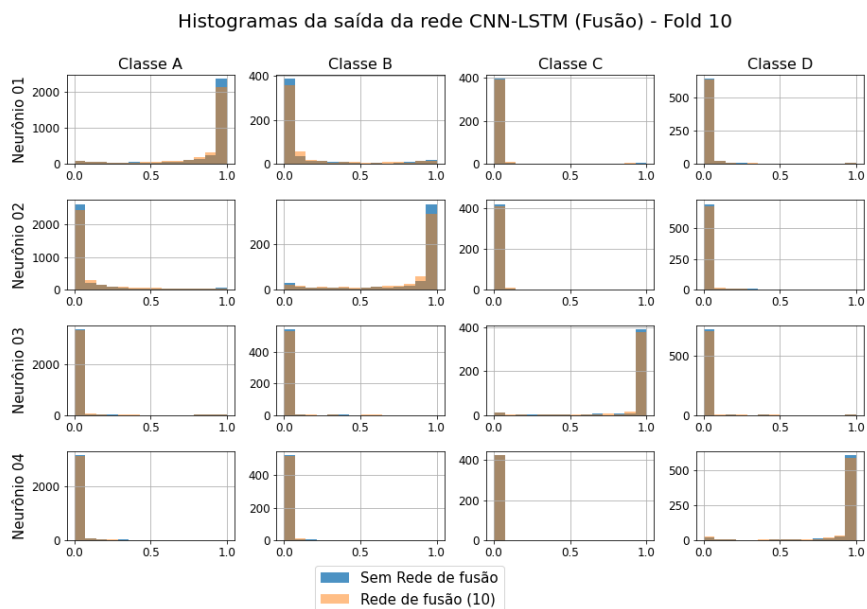
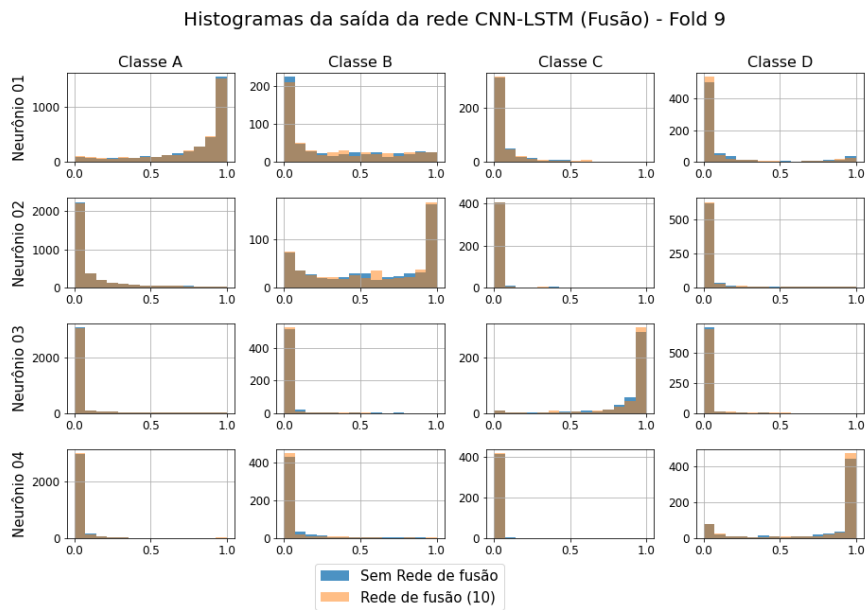
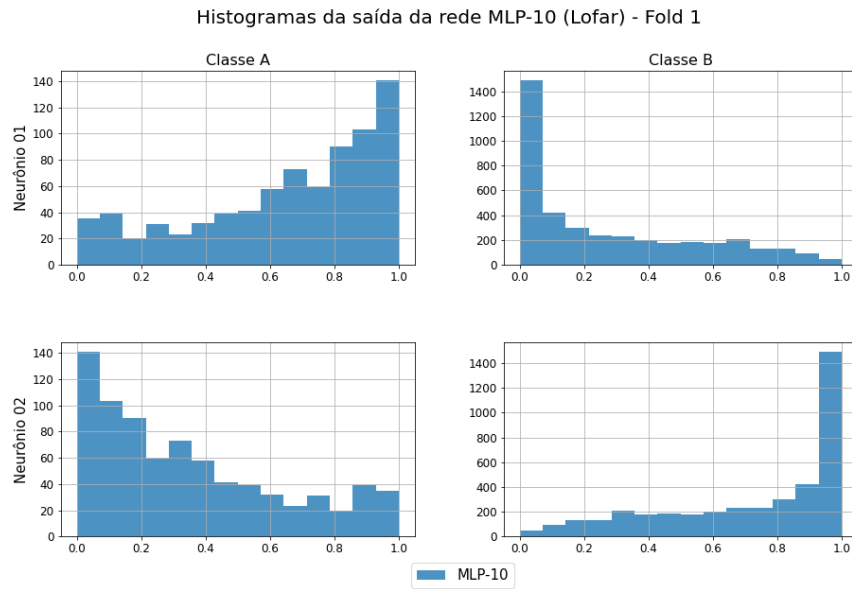


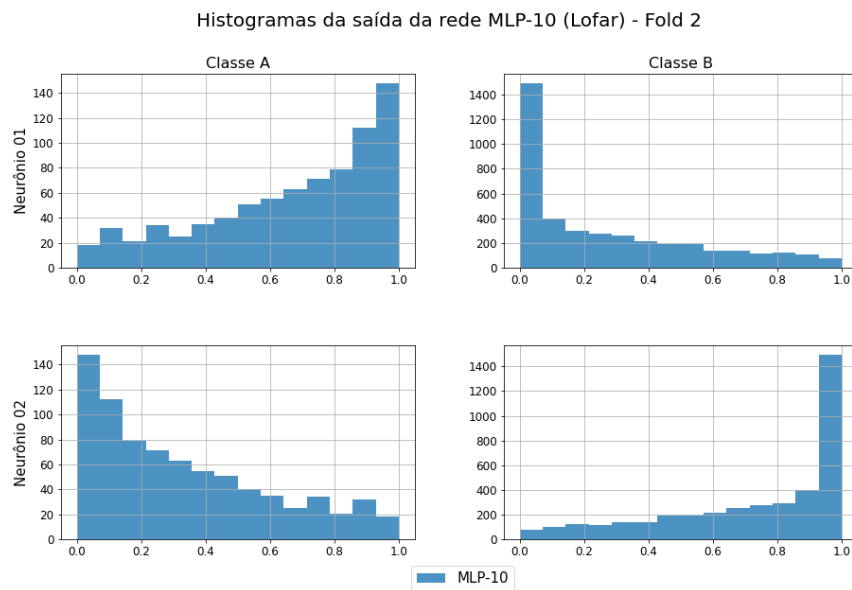
Figura C.73: Histogramas da saída da fusão do modelo CNN com LSTM (*folds* 9 e 10).

C.4.2 Águas Rasas

As Figuras de C.74 a C.78 mostram os histogramas da saída dos 10 *folds* do modelo LOFAR MLP com 10 neurônios na camada oculta.



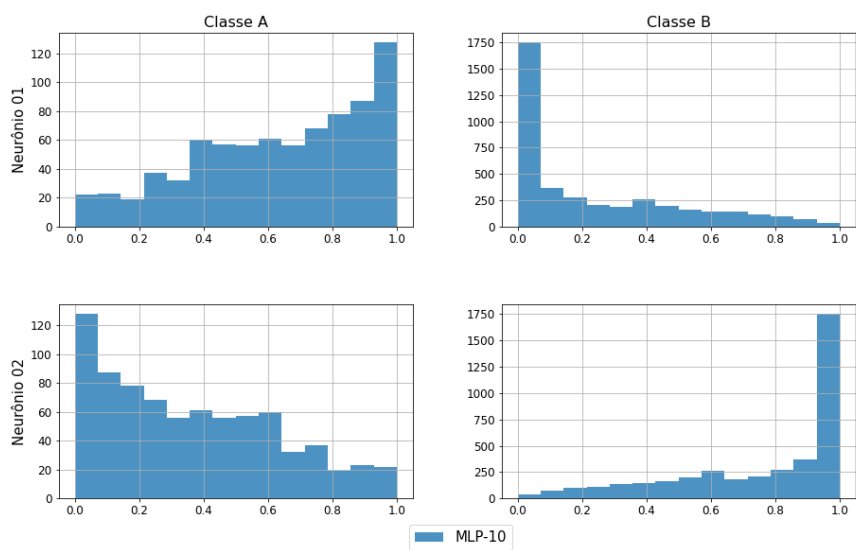
(a) *fold-1*



(b) *fold-2*

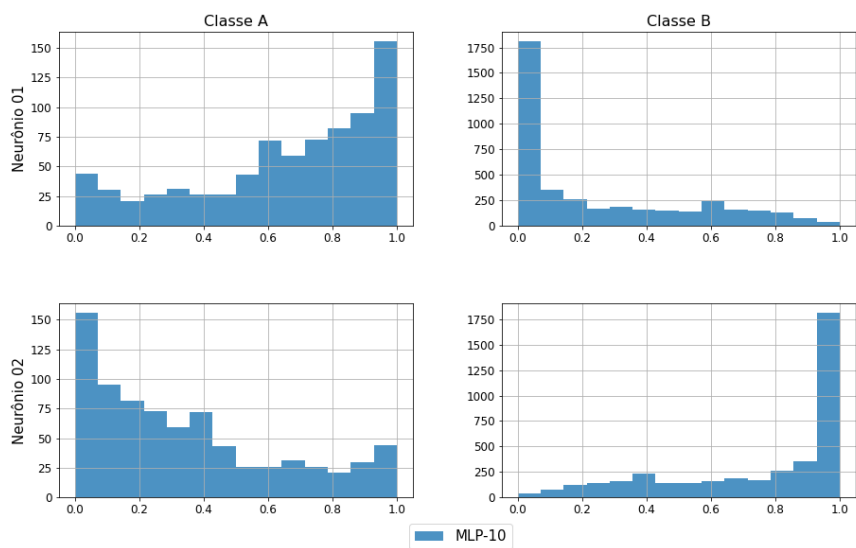
Figura C.74: Histogramas da saída do modelo LOFAR MLP-10 (*folds* 1 e 2).

Histogramas da saída da rede MLP-10 (Lofar) - Fold 3



(a) *fold-3*

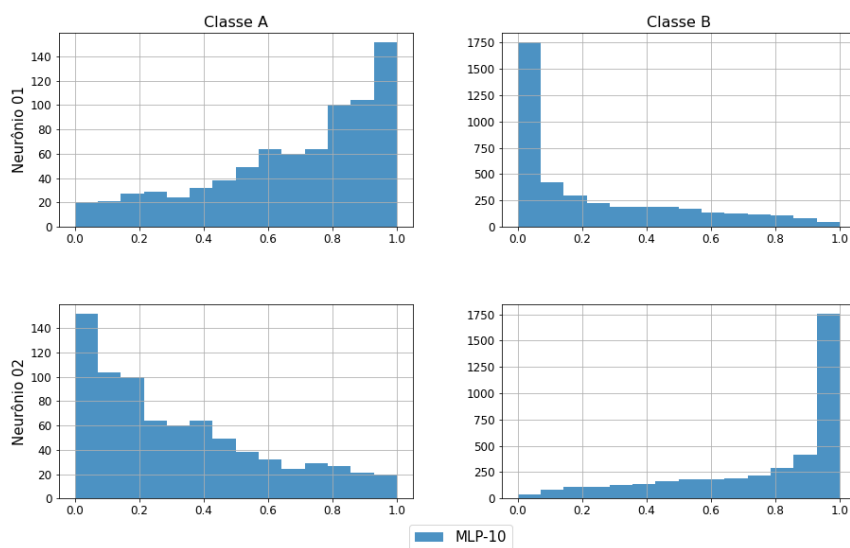
Histogramas da saída da rede MLP-10 (Lofar) - Fold 4



(b) *fold-4*

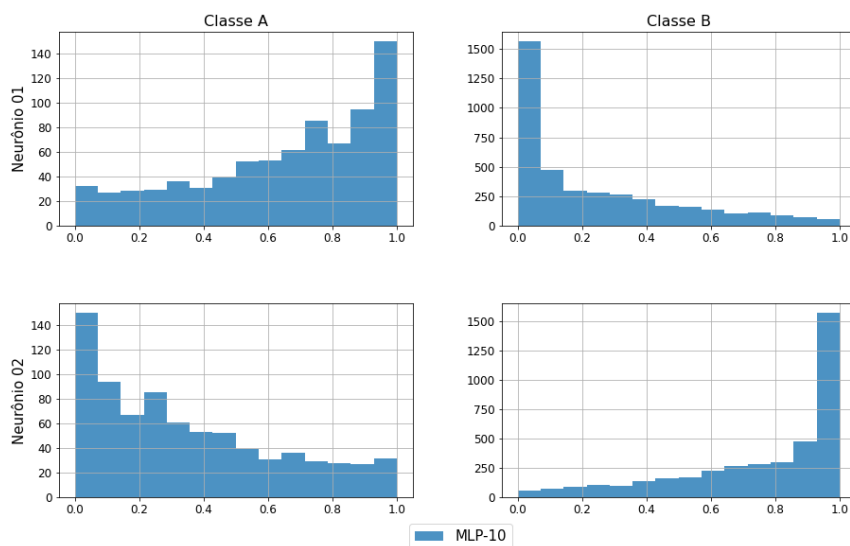
Figura C.75: Histogramas da saída do modelo LOFAR MLP-10 (*folds* 3 e 4).

Histogramas da saída da rede MLP-10 (Lofar) - Fold 5



(a) *fold-5*

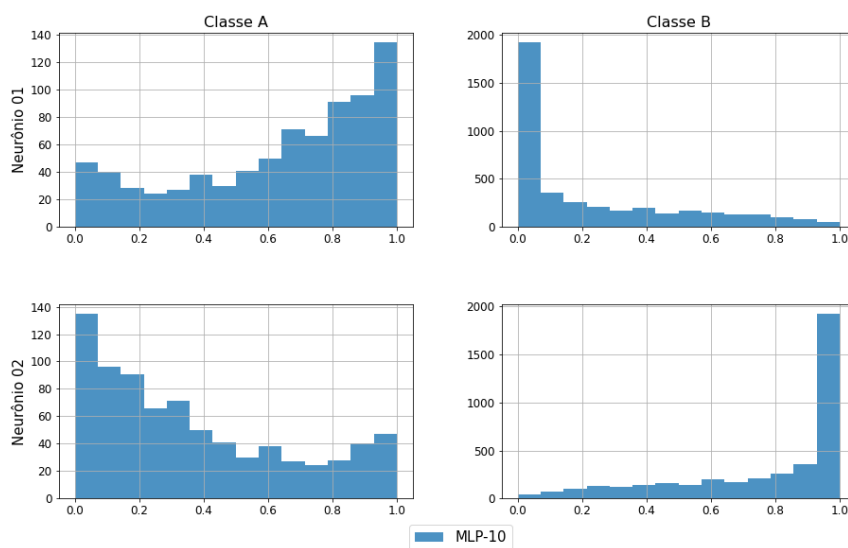
Histogramas da saída da rede MLP-10 (Lofar) - Fold 6



(b) *fold-6*

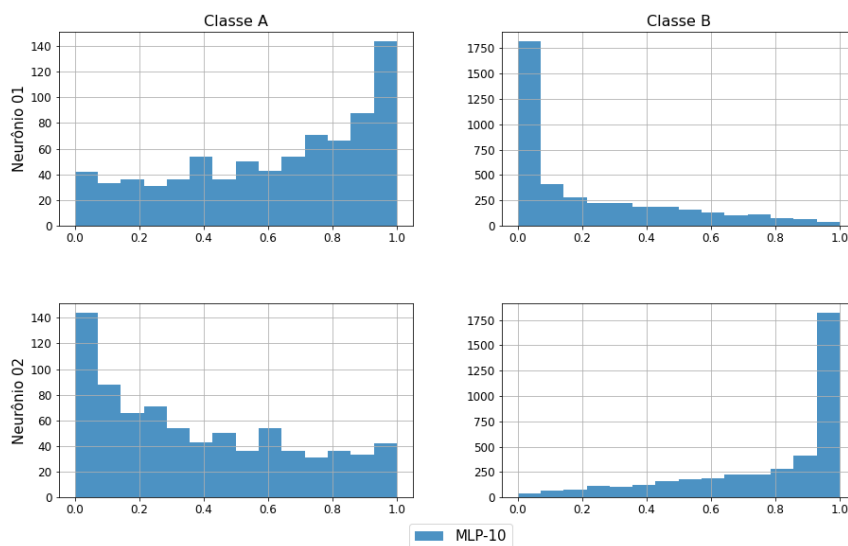
Figura C.76: Histogramas da saída do modelo LOFAR MLP-10 (*folds* 5 e 6).

Histogramas da saída da rede MLP-10 (Lofar) - Fold 7



(a) *fold-7*

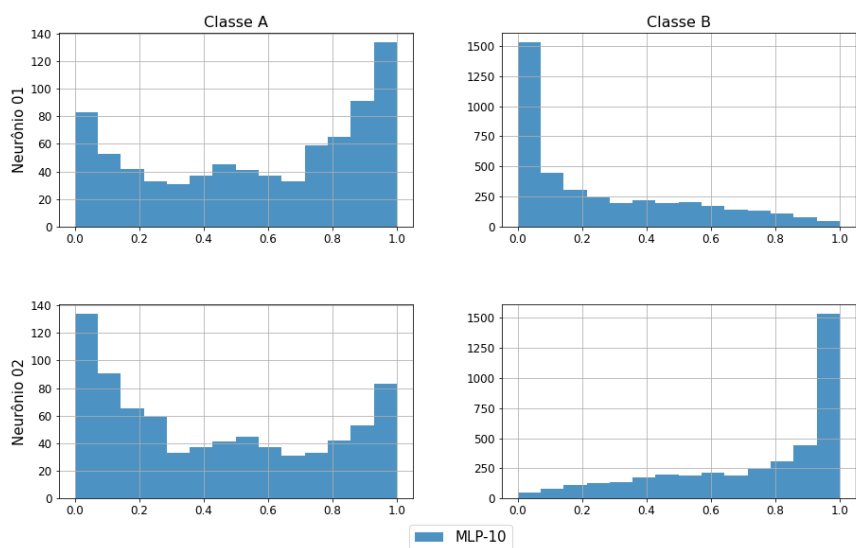
Histogramas da saída da rede MLP-10 (Lofar) - Fold 8



(b) *fold-8*

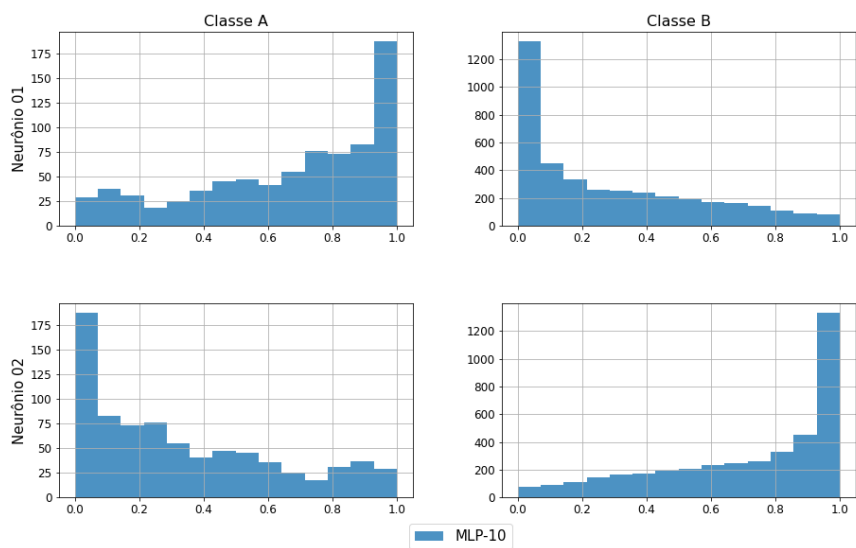
Figura C.77: Histogramas da saída do modelo LOFAR MLP-10 (*folds* 7 e 8).

Histogramas da saída da rede MLP-10 (Lofar) - Fold 9



(a) *fold-9*

Histogramas da saída da rede MLP-10 (Lofar) - Fold 10



(b) *fold-10*

Figura C.78: Histogramas da saída do modelo LOFAR MLP-10 (*folds* 9 e 10).

As Figuras de C.79 a C.83 mostram os histogramas da saída dos 10 *fold*s dos modelos temporais MLP com 10 e 20 neurônios na camada oculta.

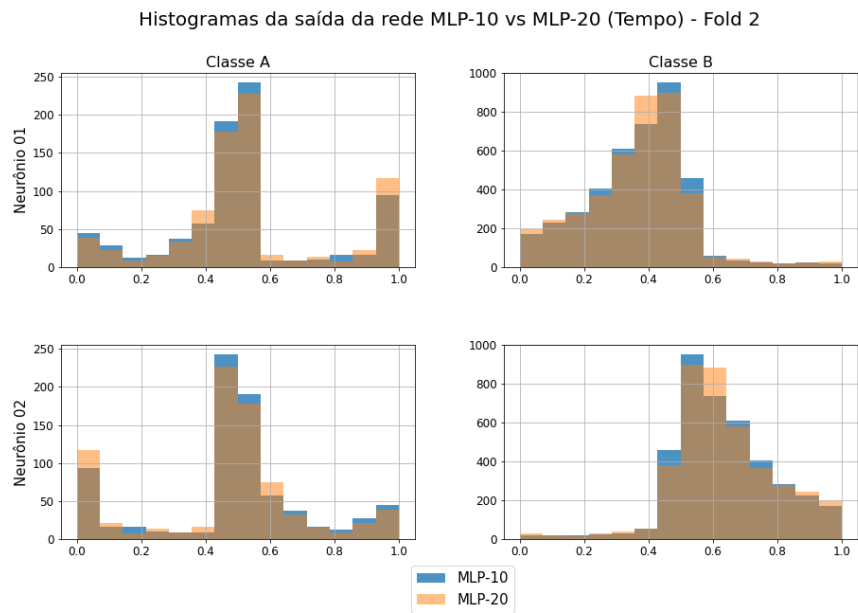
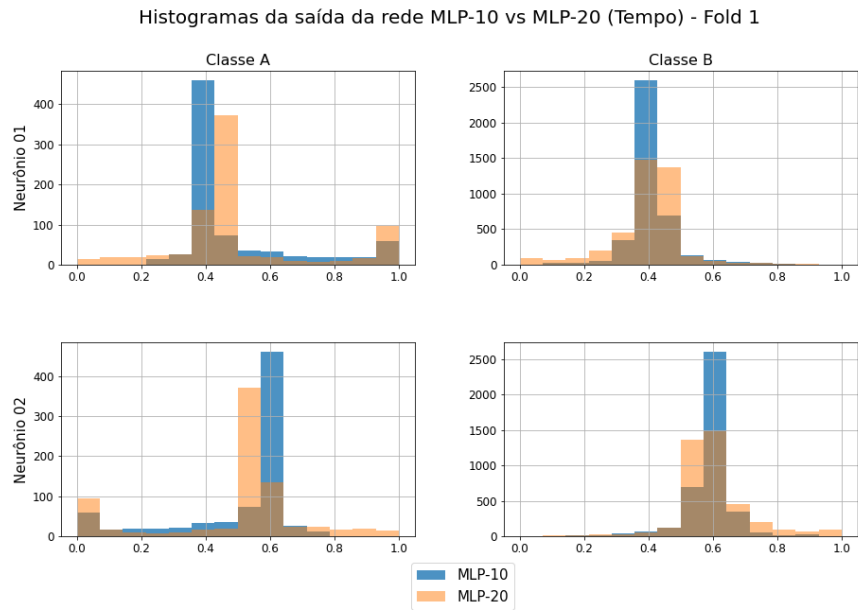
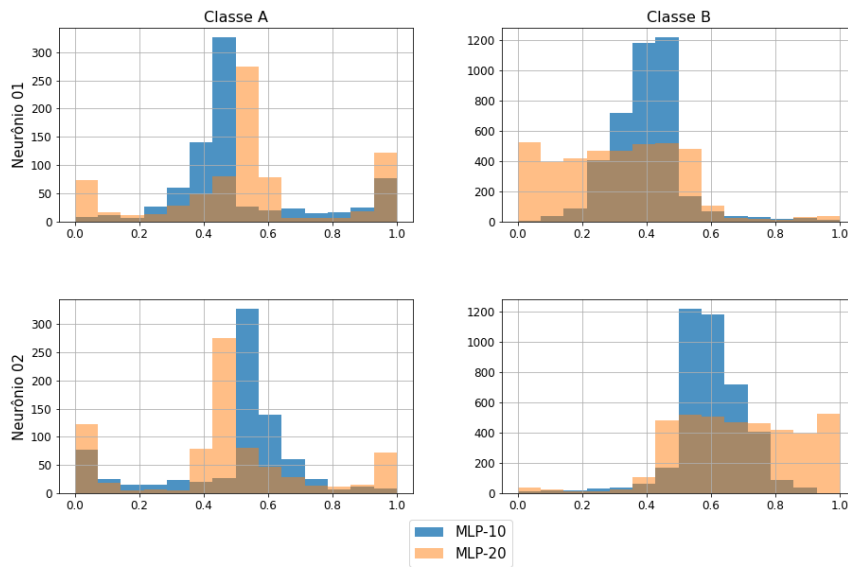


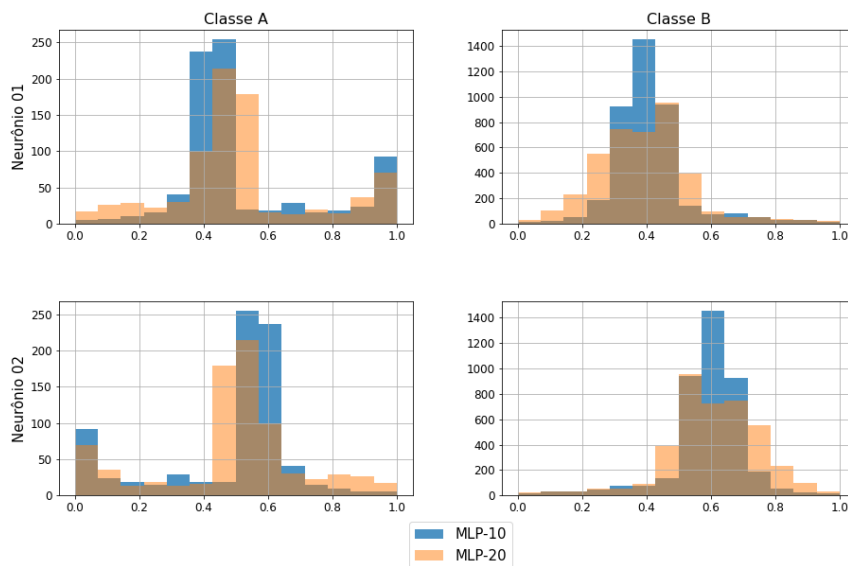
Figura C.79: Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (*fold*s 1 e 2).

Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 3



(a) *fold-3*

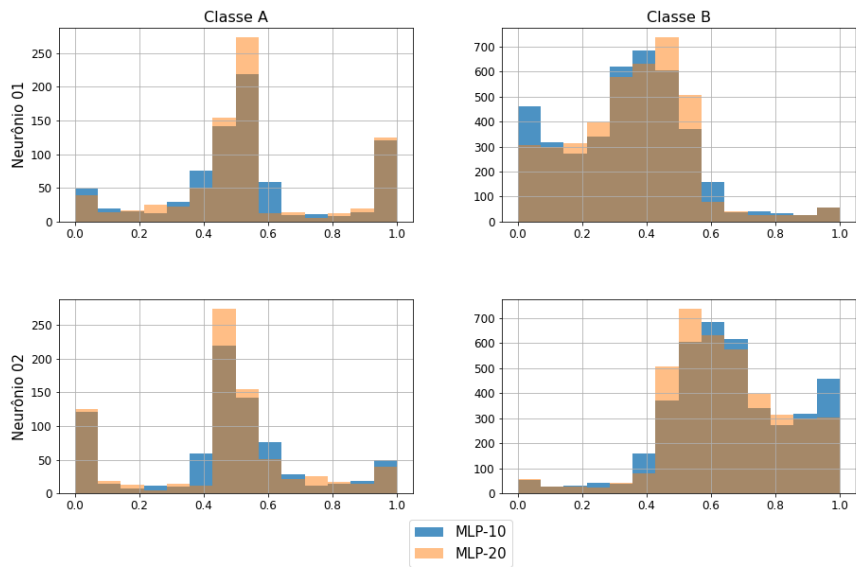
Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 4



(b) *fold-4*

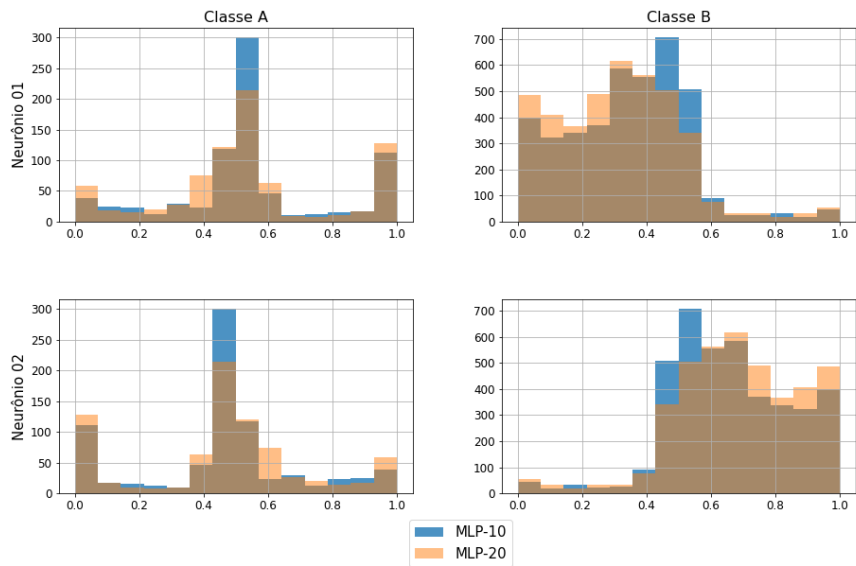
Figura C.80: Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (*folds* 3 e 4).

Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 5



(a) *fold-5*

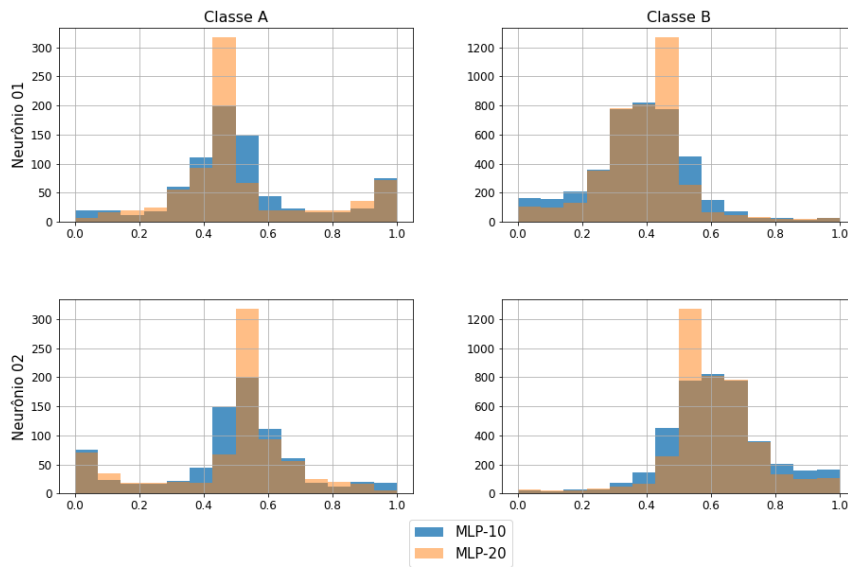
Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 6



(b) *fold-6*

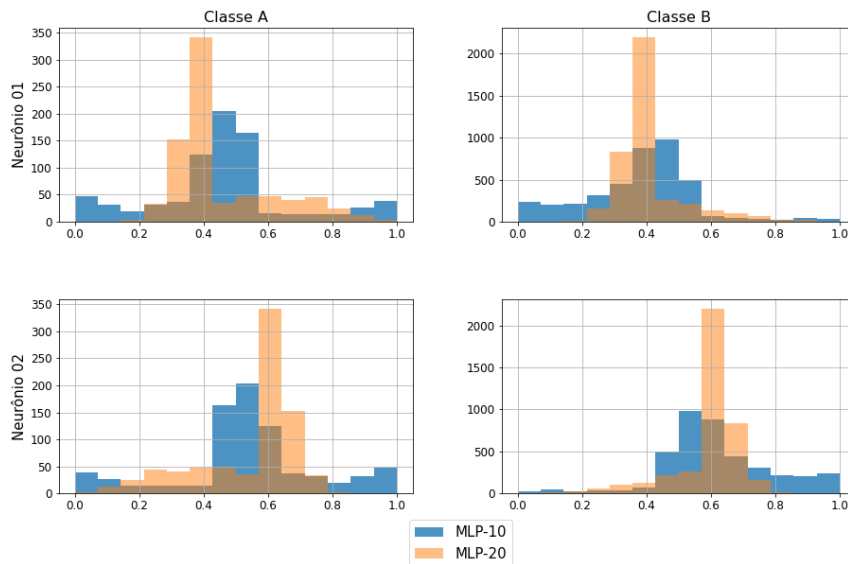
Figura C.81: Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (*fold*s 5 e 6).

Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 7



(a) *fold-7*

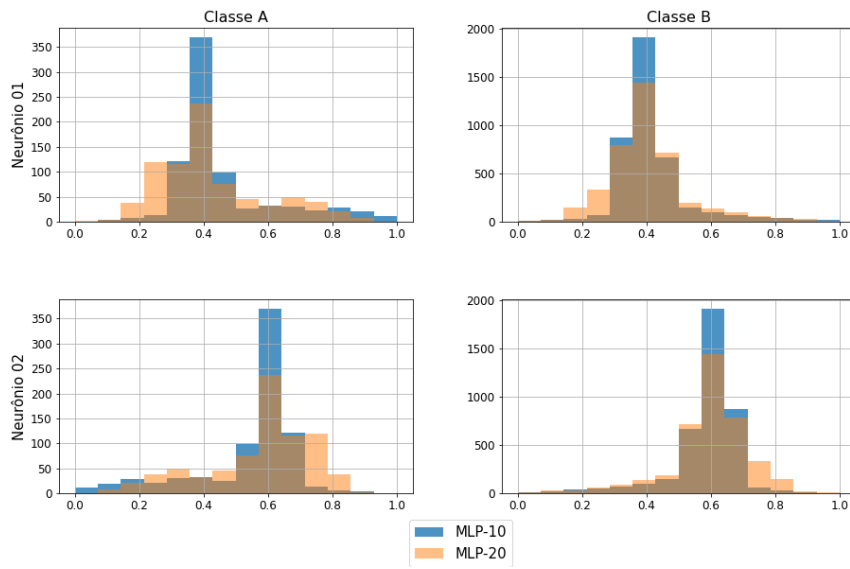
Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 8



(b) *fold-8*

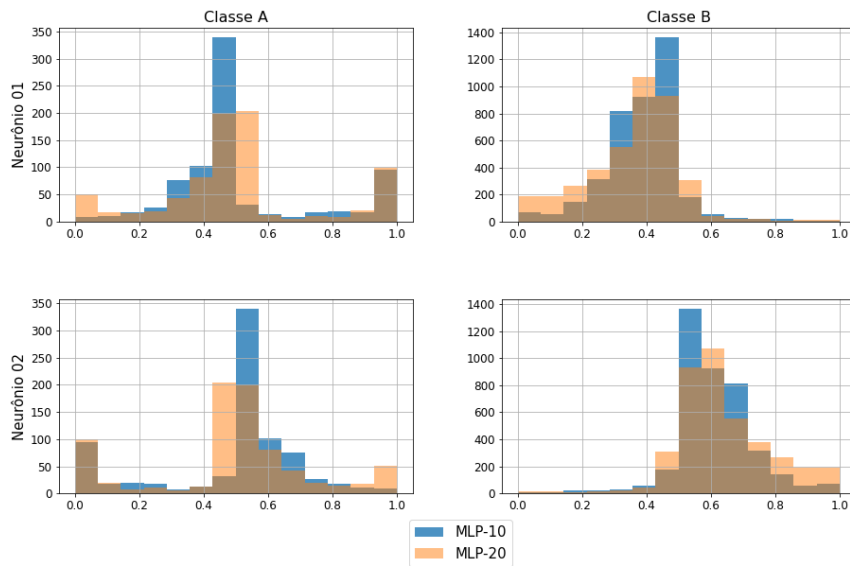
Figura C.82: Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (*folds* 7 e 8).

Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 9



(a) *fold-9*

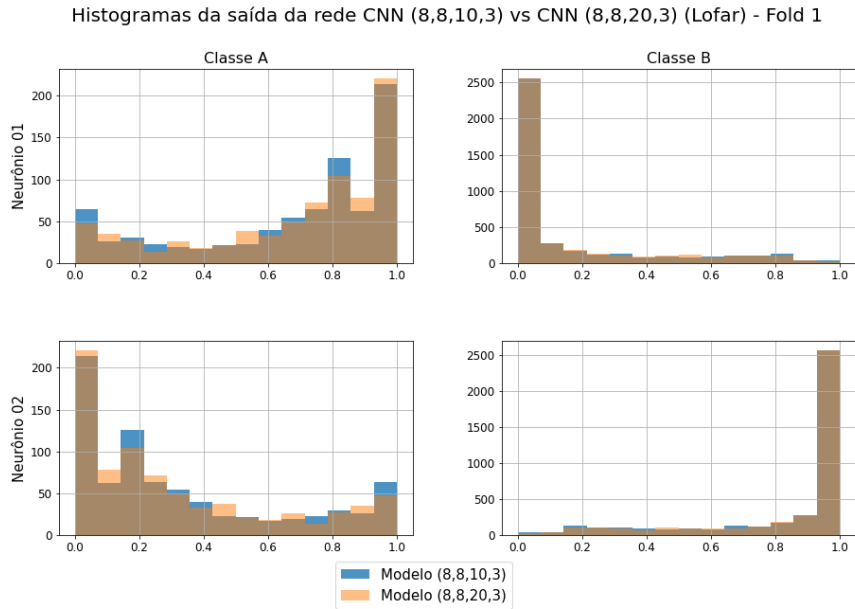
Histogramas da saída da rede MLP-10 vs MLP-20 (Tempo) - Fold 10



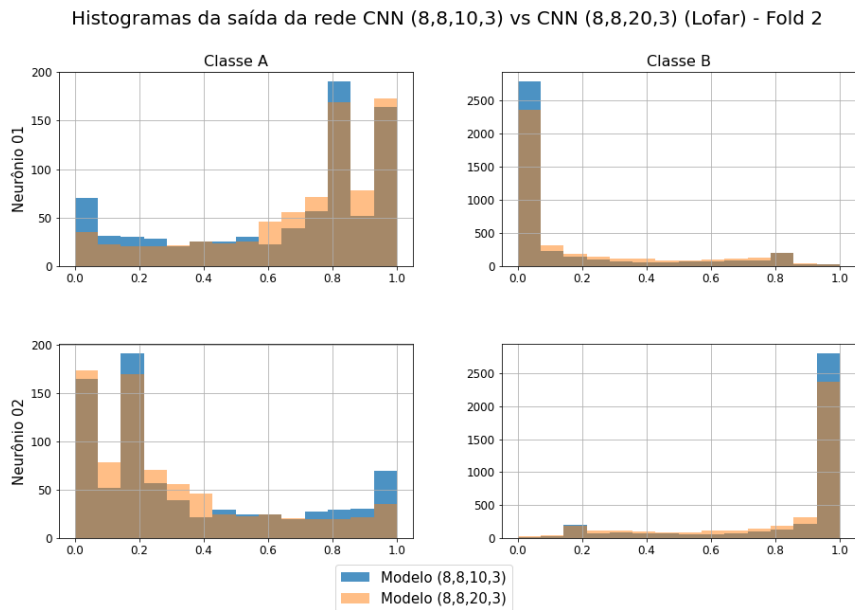
(b) *fold-10*

Figura C.83: Histogramas da saída dos modelos temporais MLP-10 e MLP-20 (*fold*s 9 e 10).

As Figuras de C.84 a C.88 mostram os histogramas da saída dos 10 *fold*s dos modelos LOFAR CNN com 10 e 20 neurônios na camada *fully connected*, a extração de características é feita por duas camadas convolucionais de 8 filtros cada e *kernel* 3.



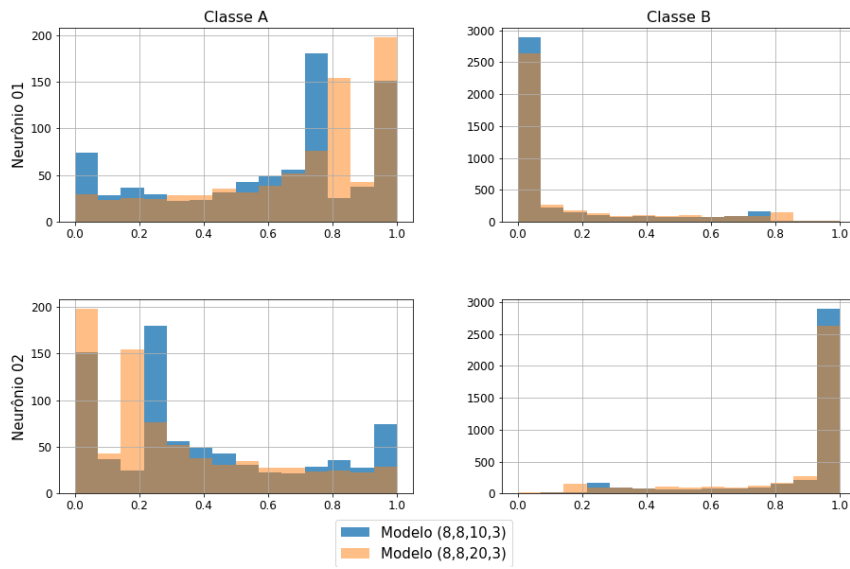
(a) *fold-1*



(b) *fold-2*

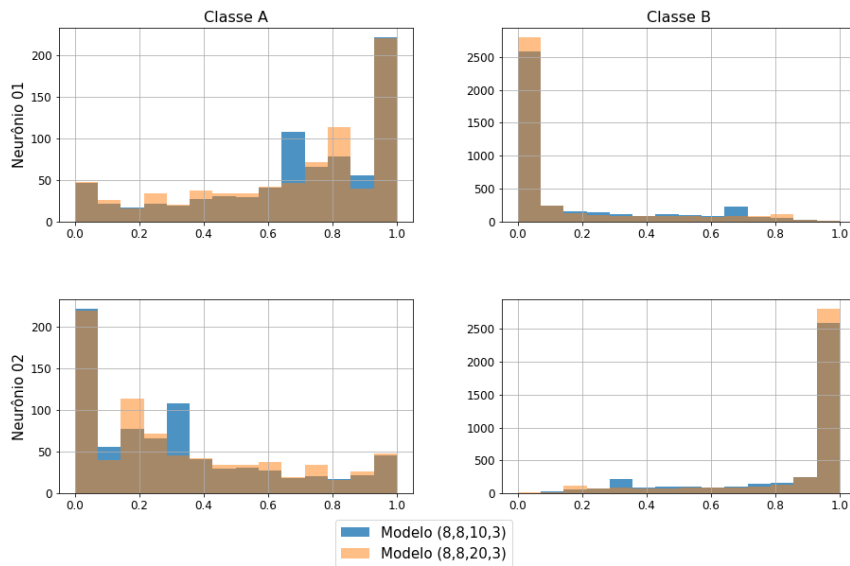
Figura C.84: Histogramas da saída dos modelos LOFAR CNN (*fold*s 1 e 2).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 3



(a) *fold-3*

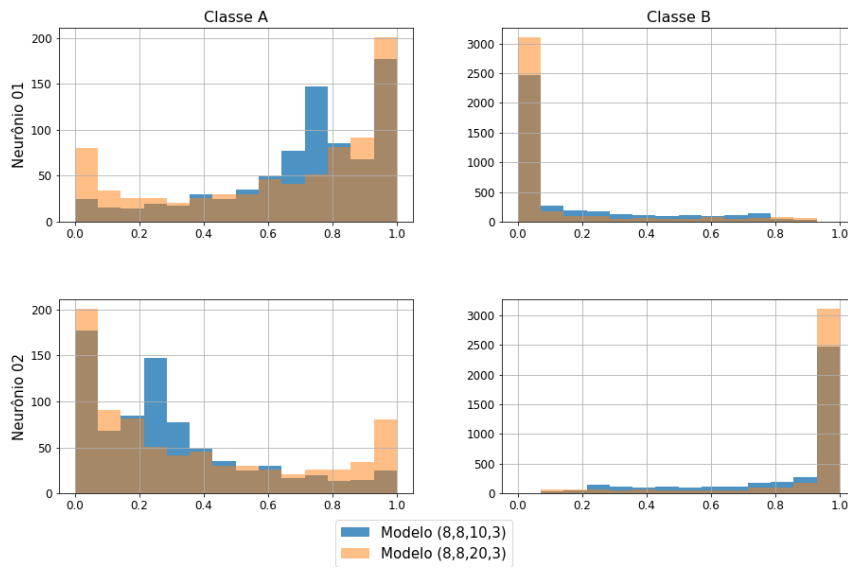
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 4



(b) *fold-4*

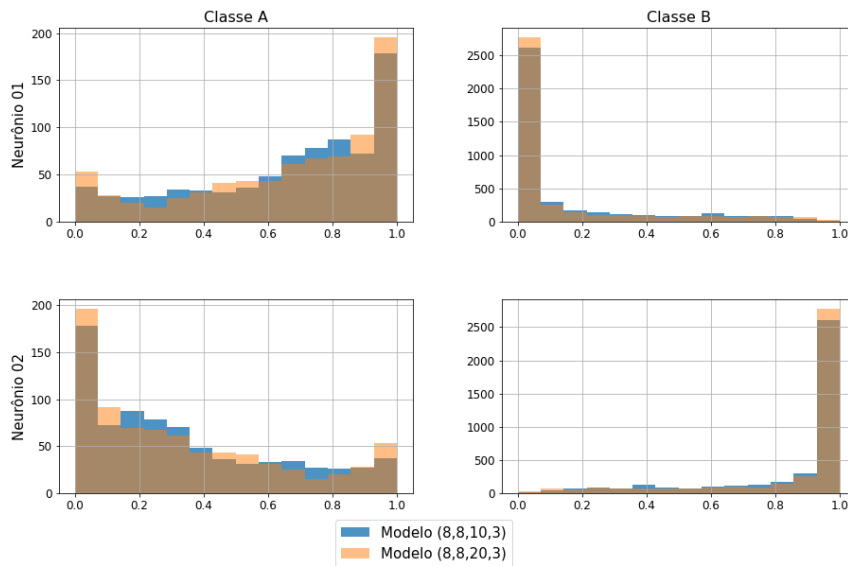
Figura C.85: Histogramas da saída dos modelos LOFAR CNN (*folds* 3 e 4).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 5



(a) *fold-5*

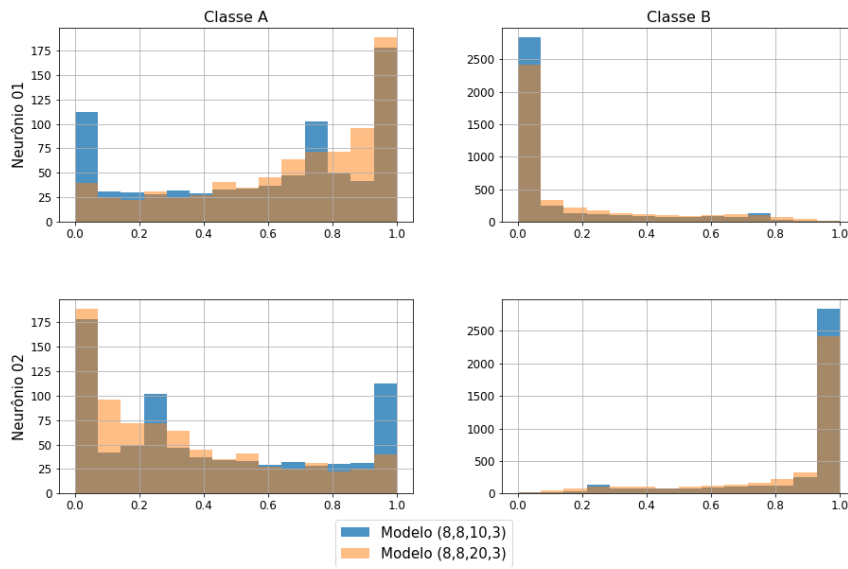
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 6



(b) *fold-6*

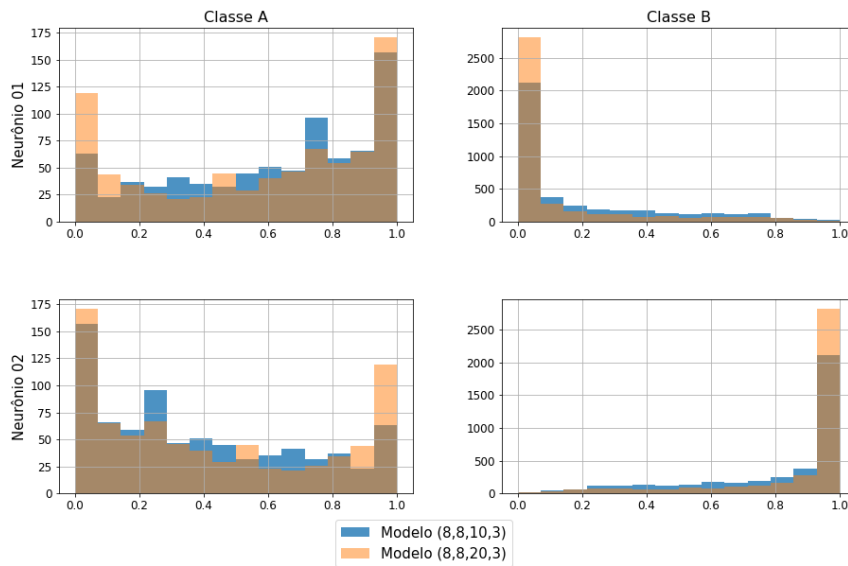
Figura C.86: Histogramas da saída dos modelos LOFAR CNN (*folds* 5 e 6).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 7



(a) *fold-7*

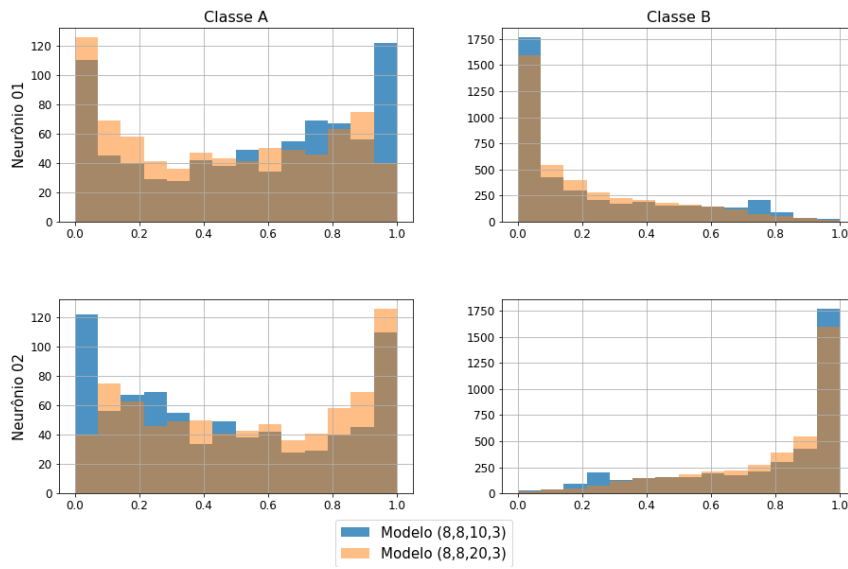
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 8



(b) *fold-8*

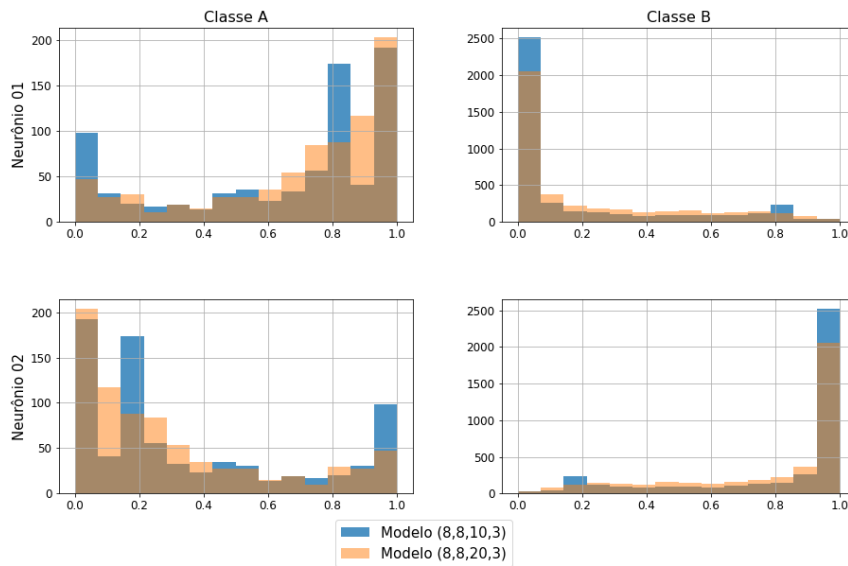
Figura C.87: Histogramas da saída dos modelos LOFAR CNN (*folders* 7 e 8).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 9



(a) *fold-9*

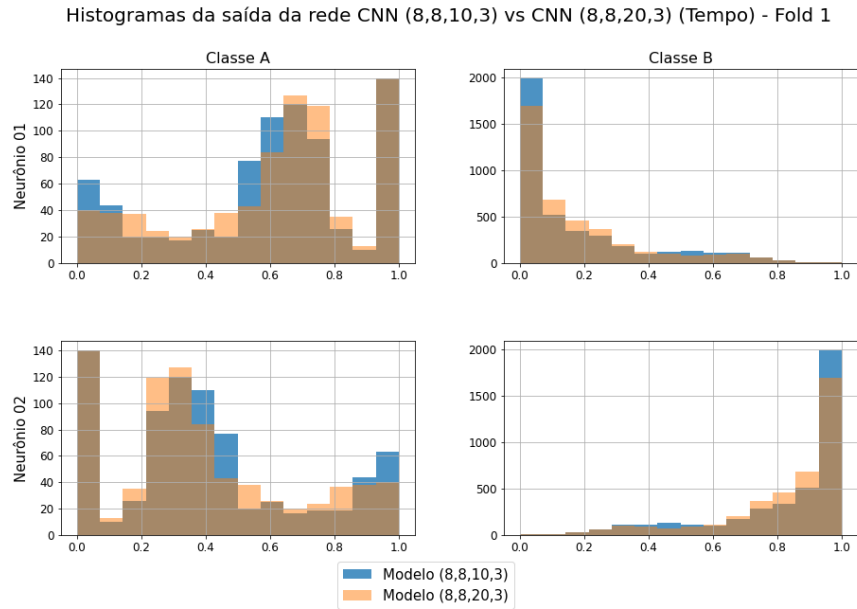
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Lofar) - Fold 10



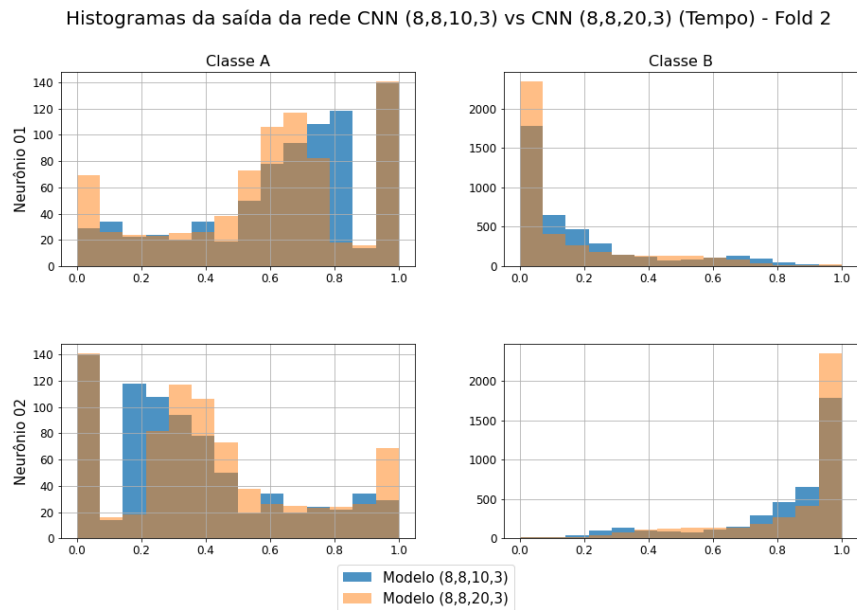
(b) *fold-10*

Figura C.88: Histogramas da saída dos modelos LOFAR CNN (*folds* 9 e 10).

As Figuras de C.89 a C.93 mostram os histogramas da saída dos 10 *folds* dos modelos temporais CNN com 10 e 20 neurônios na camada *fully connected*, a extração de características é feita por duas camadas convolucionais de 8 filtros cada e *kernel* 3.



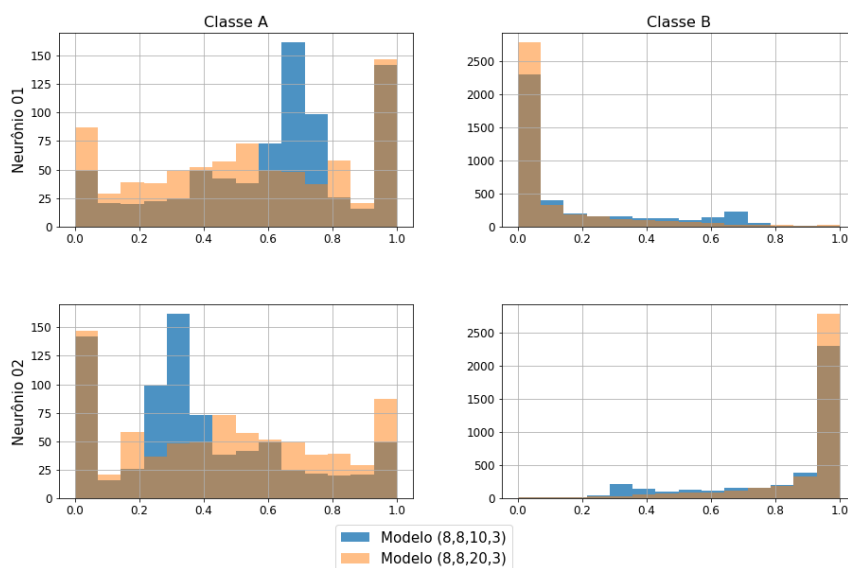
(a) *fold-1*



(b) *fold-2*

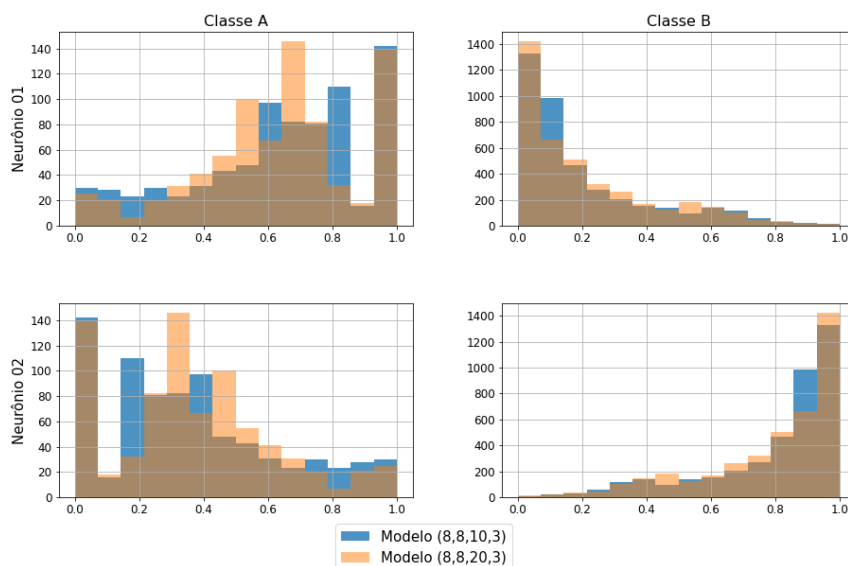
Figura C.89: Histogramas da saída dos modelos temporais CNN (*folds* 1 e 2).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 3



(a) *fold-3*

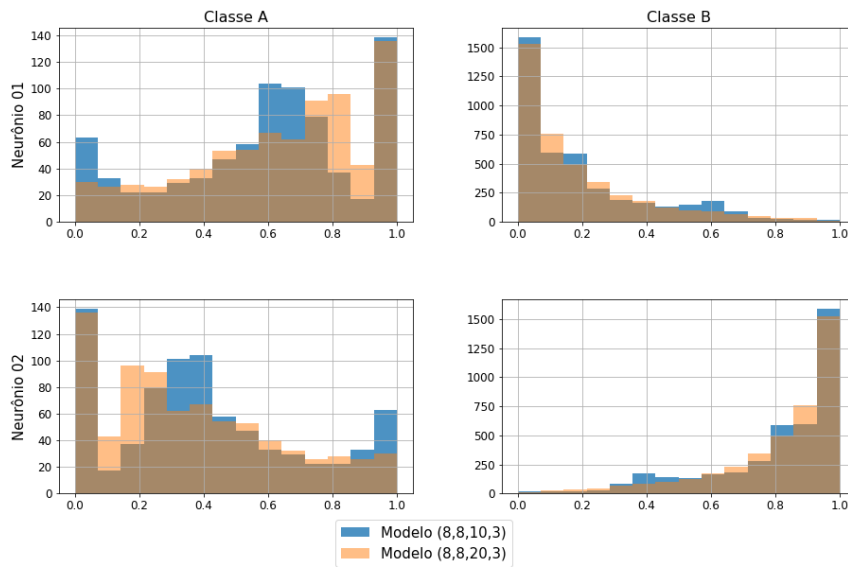
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 4



(b) *fold-4*

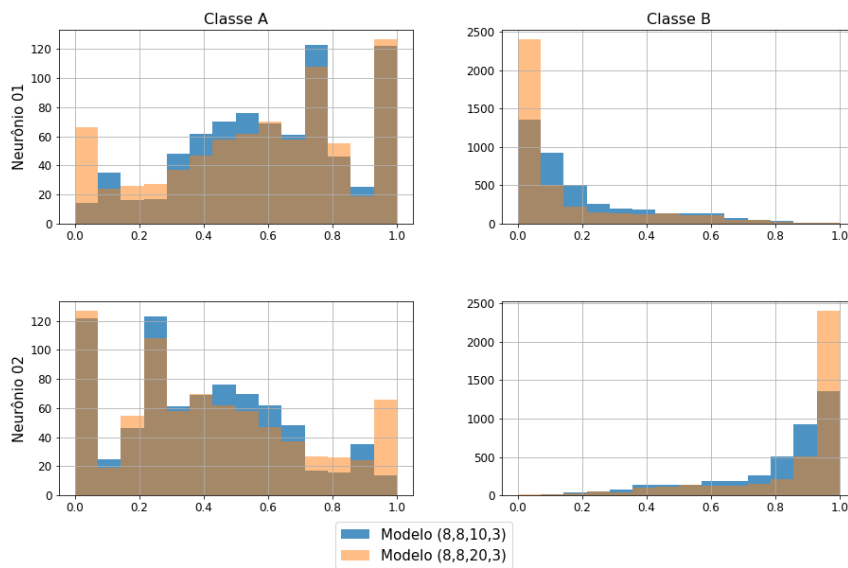
Figura C.90: Histogramas da saída dos modelos temporais CNN (*folds* 3 e 4).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 5



(a) *fold-5*

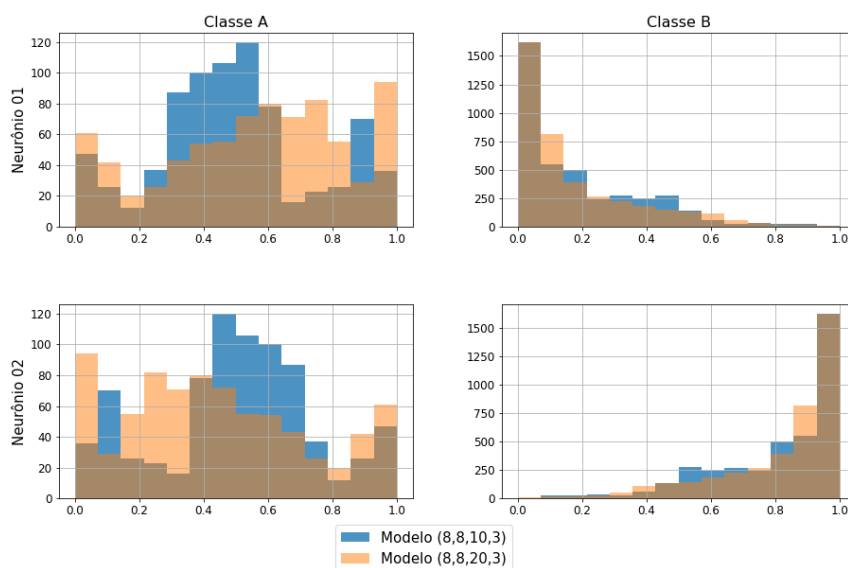
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 6



(b) *fold-6*

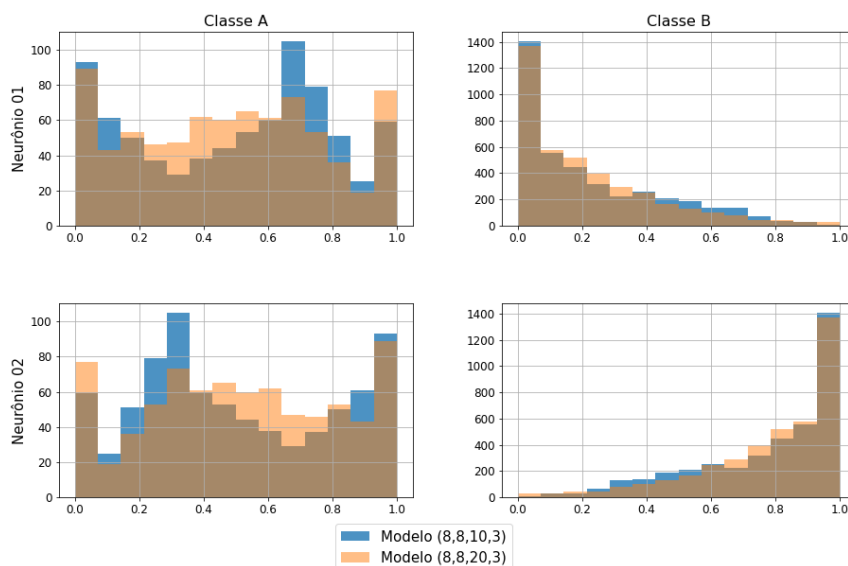
Figura C.91: Histogramas da saída dos modelos temporais CNN (*fold*s 5 e 6).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 7



(a) *fold-7*

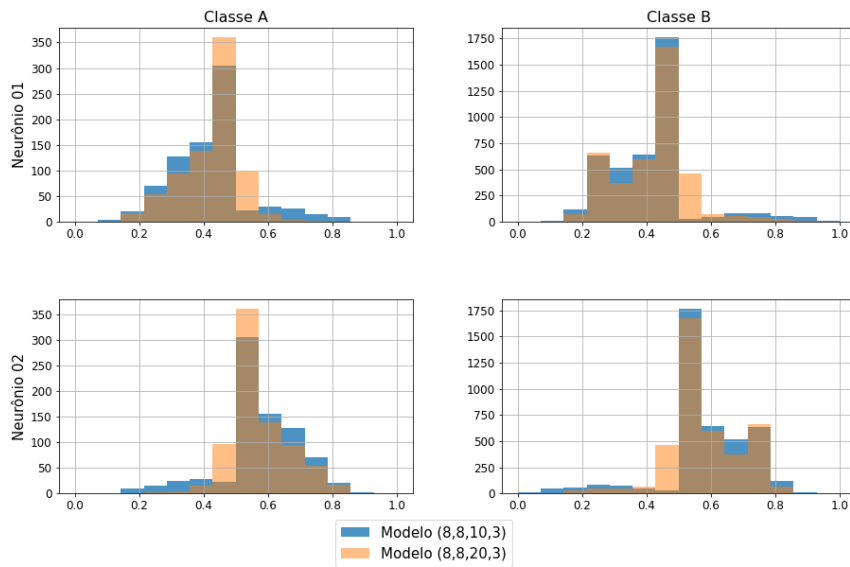
Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 8



(b) *fold-8*

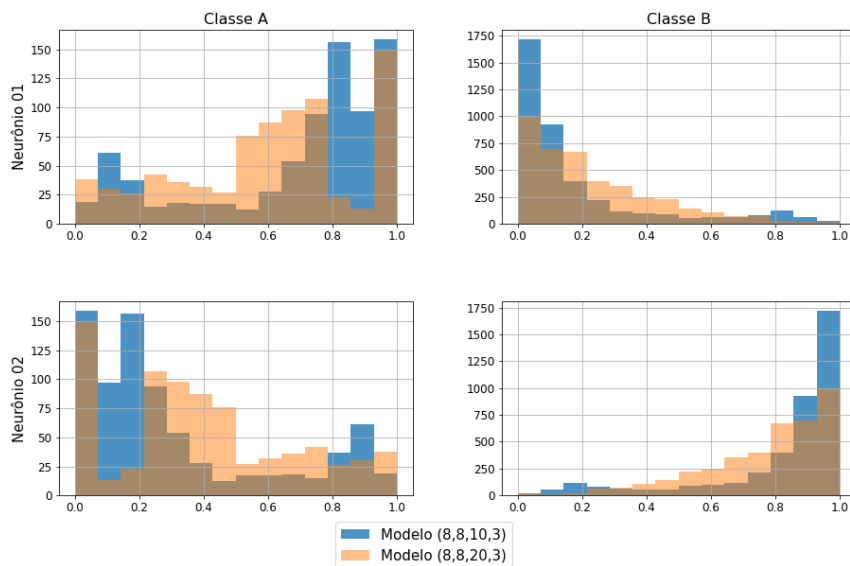
Figura C.92: Histogramas da saída dos modelos temporais CNN (*folders* 7 e 8).

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 9



(a) *fold-9*

Histogramas da saída da rede CNN (8,8,10,3) vs CNN (8,8,20,3) (Tempo) - Fold 10



(b) *fold-10*

Figura C.93: Histogramas da saída dos modelos temporais CNN (*folds* 9 e 10).

As Figuras de C.94 a C.98 mostram os histogramas da saída dos 10 *fold*s do modelo temporal LSTM com 50 células na camada oculta.

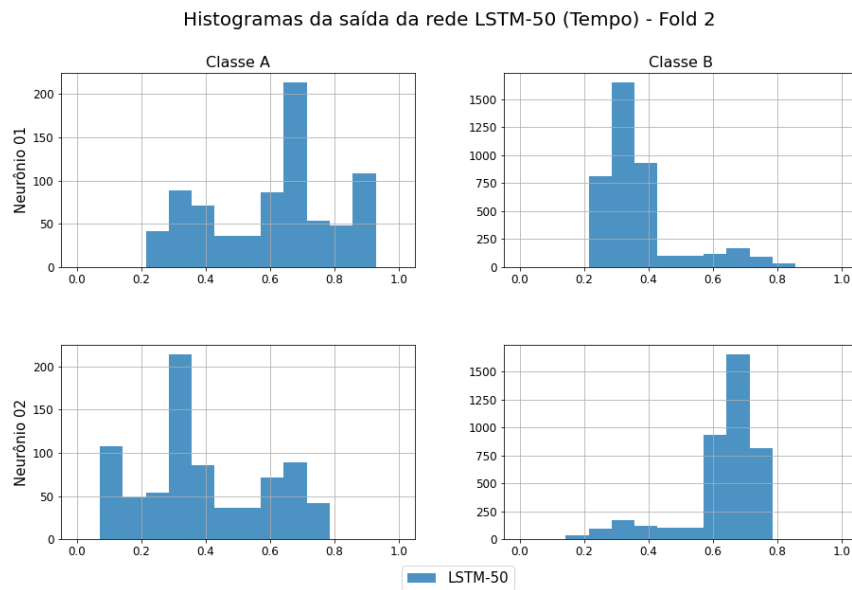
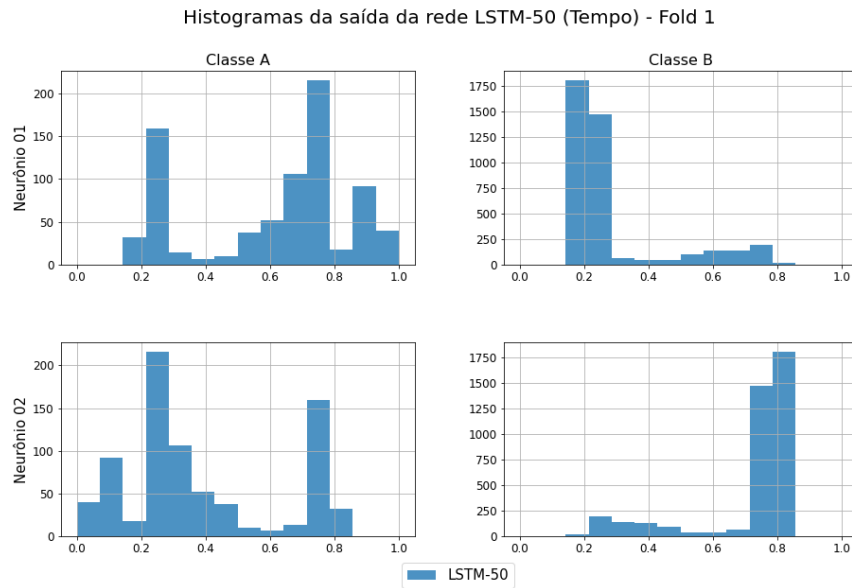
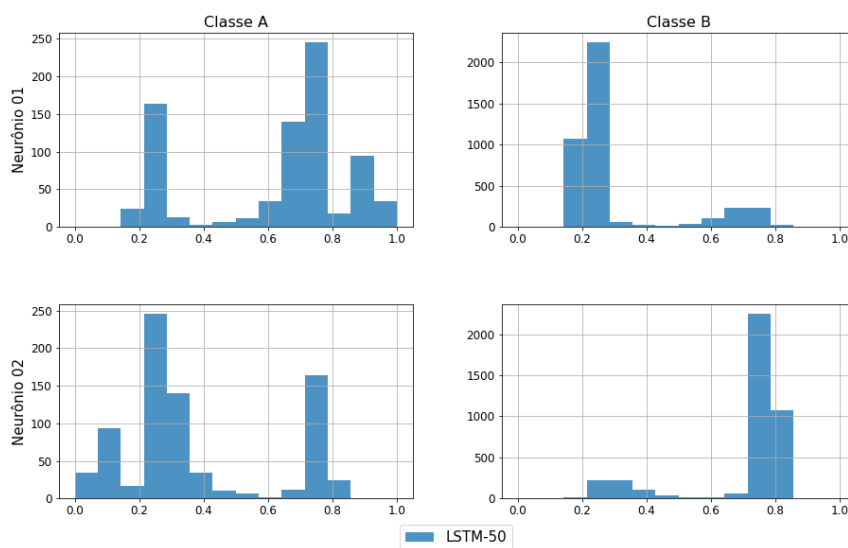


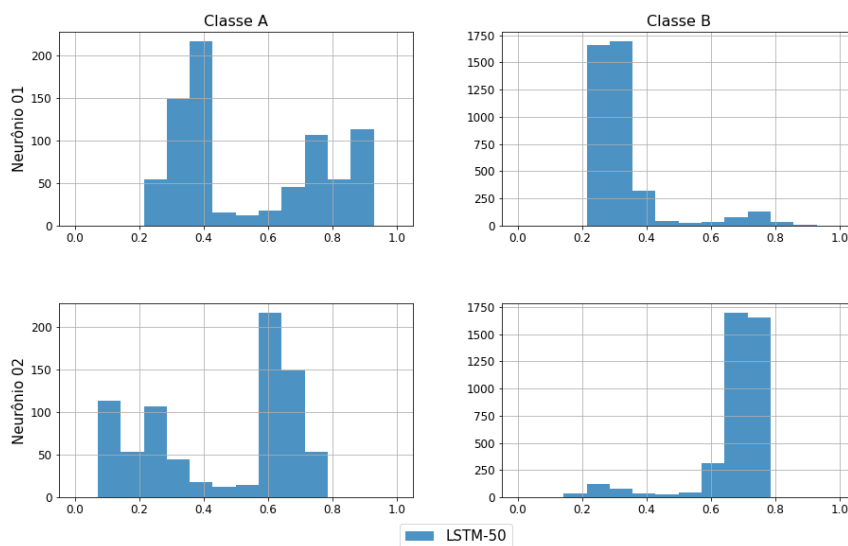
Figura C.94: Histogramas da saída dos modelos temporais LSTM (*fold*s 1 e 2).

Histogramas da saída da rede LSTM-50 (Tempo) - Fold 3



(a) *fold-3*

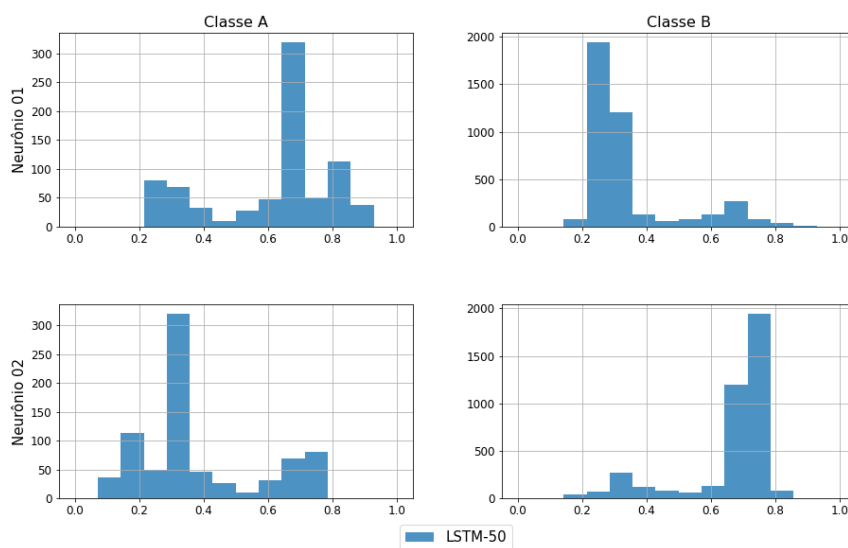
Histogramas da saída da rede LSTM-50 (Tempo) - Fold 4



(b) *fold-4*

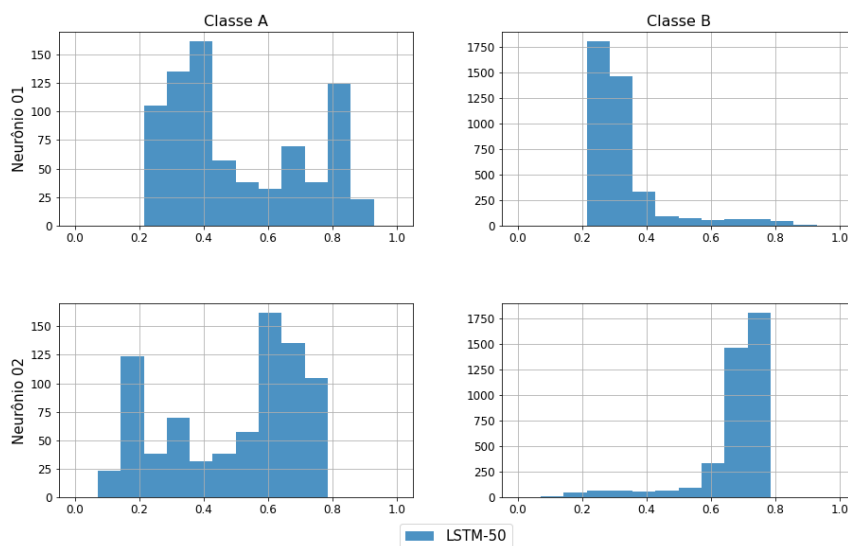
Figura C.95: Histogramas da saída dos modelos temporais LSTM (*folders* 3 e 4).

Histogramas da saída da rede LSTM-50 (Tempo) - Fold 5



(a) *fold-5*

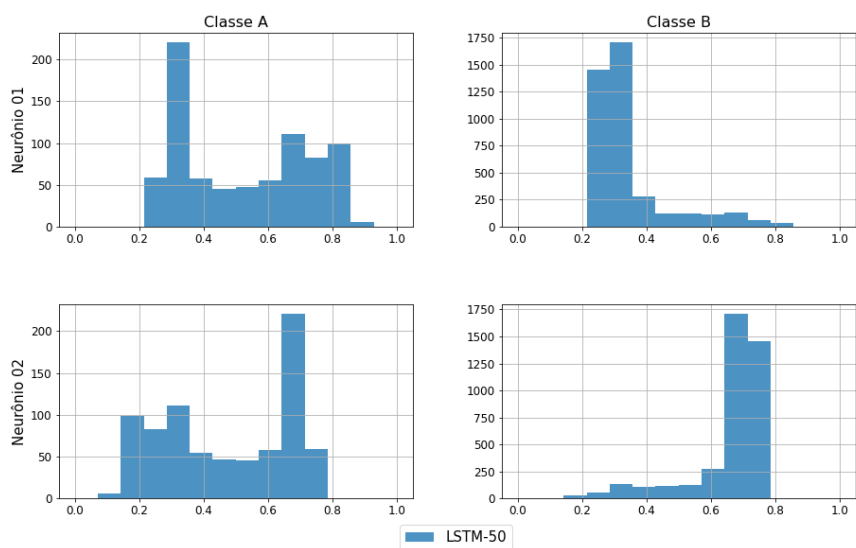
Histogramas da saída da rede LSTM-50 (Tempo) - Fold 6



(b) *fold-6*

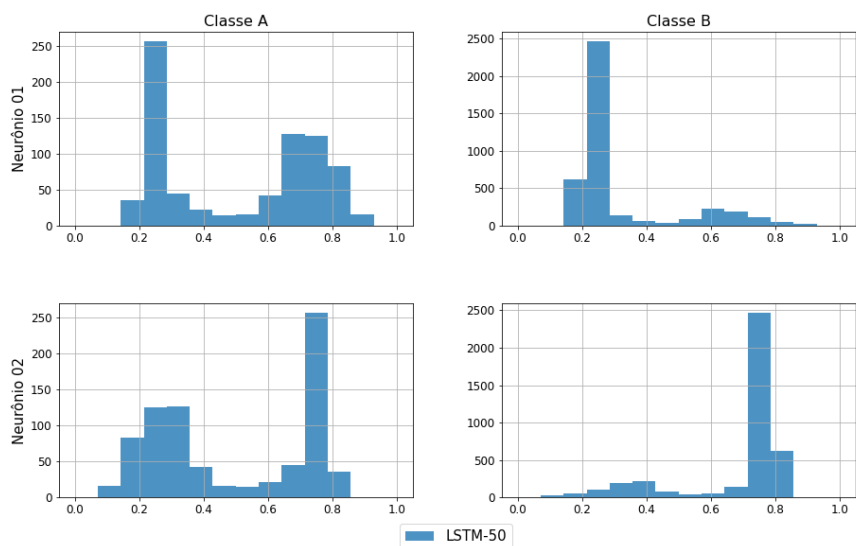
Figura C.96: Histogramas da saída dos modelos temporais LSTM (*folds* 5 e 6).

Histogramas da saída da rede LSTM-50 (Tempo) - Fold 7



(a) *fold-7*

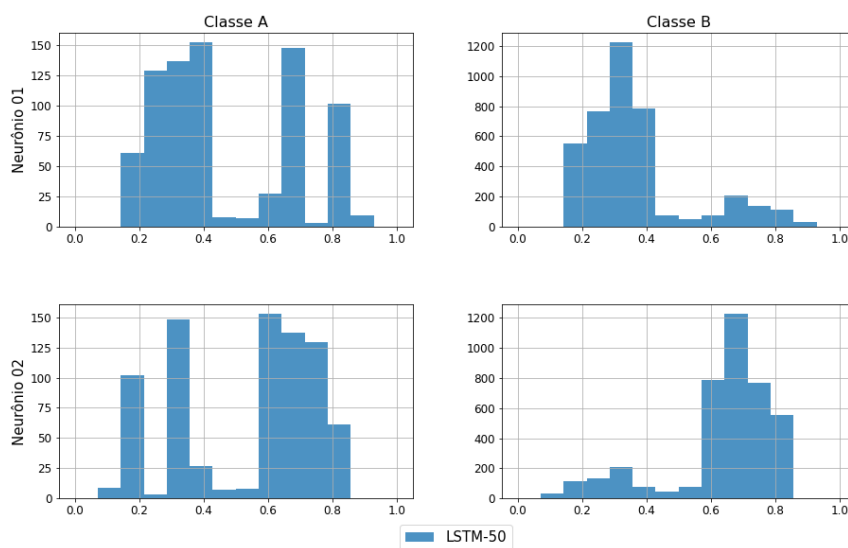
Histogramas da saída da rede LSTM-50 (Tempo) - Fold 8



(b) *fold-8*

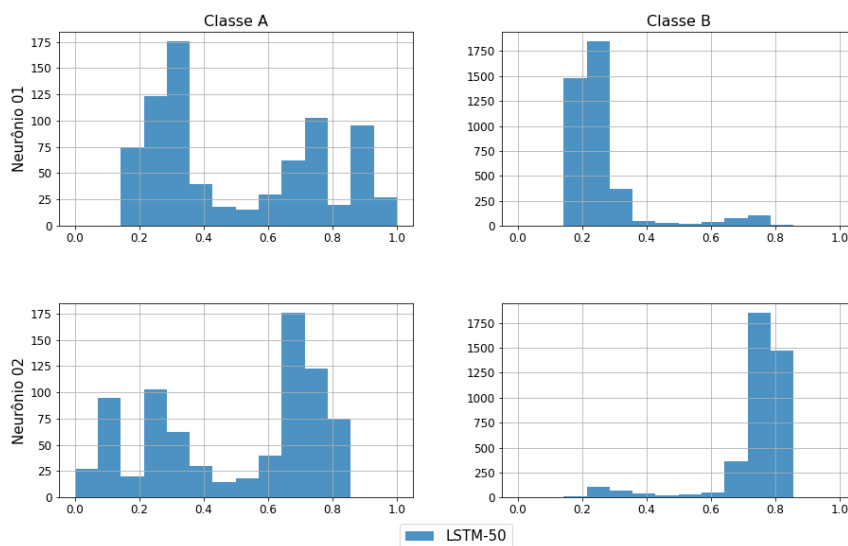
Figura C.97: Histogramas da saída dos modelos temporais LSTM (*folds* 7 e 8).

Histogramas da saída da rede LSTM-50 (Tempo) - Fold 9



(a) *fold-9*

Histogramas da saída da rede LSTM-50 (Tempo) - Fold 10

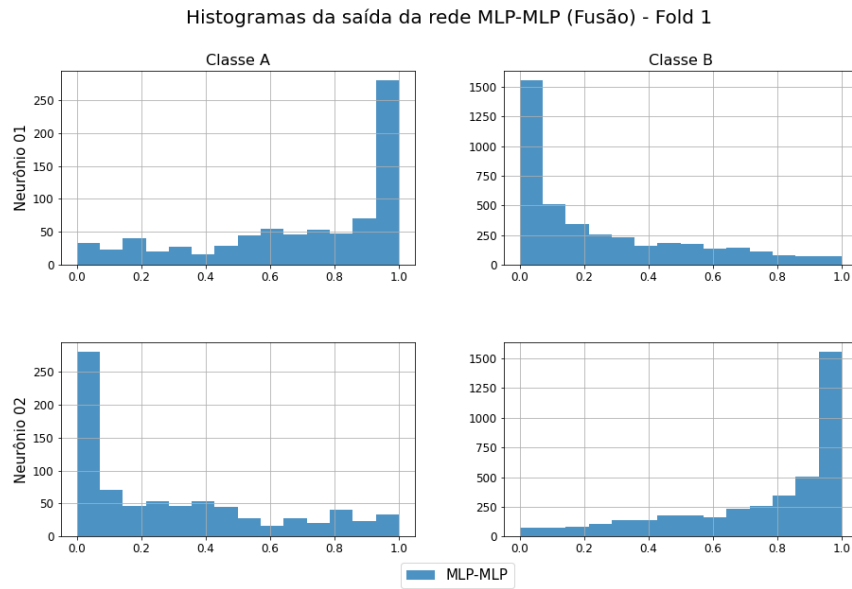


(b) *fold-10*

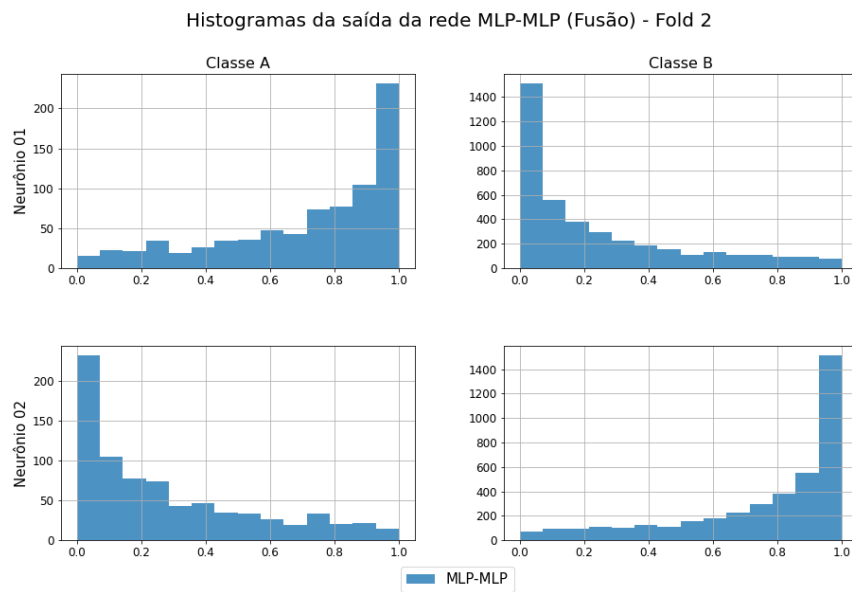
Figura C.98: Histogramas da saída dos modelos temporais LSTM (*fold*s 9 e 10).

Fusão dos modelos

As Figuras de C.99 a C.103 mostram os histogramas da saída dos 10 *folds* da fusão dos modelos MLP.



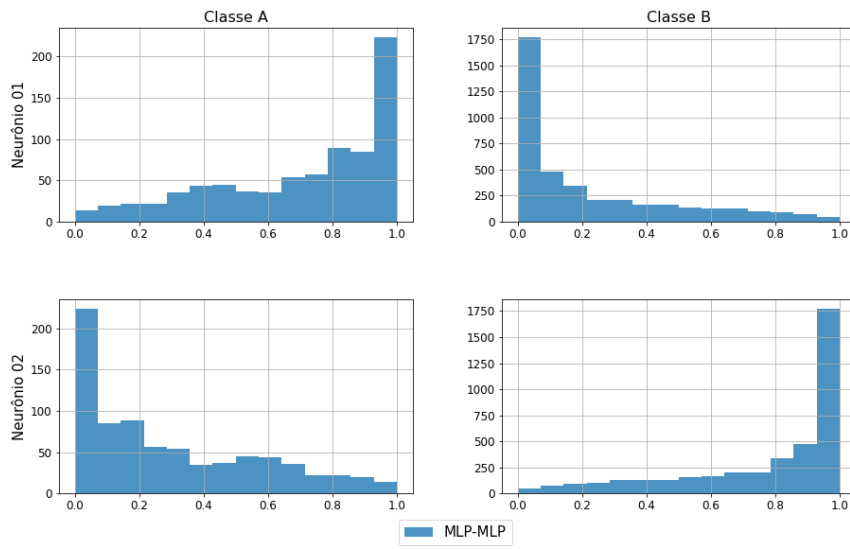
(a) *fold-1*



(b) *fold-2*

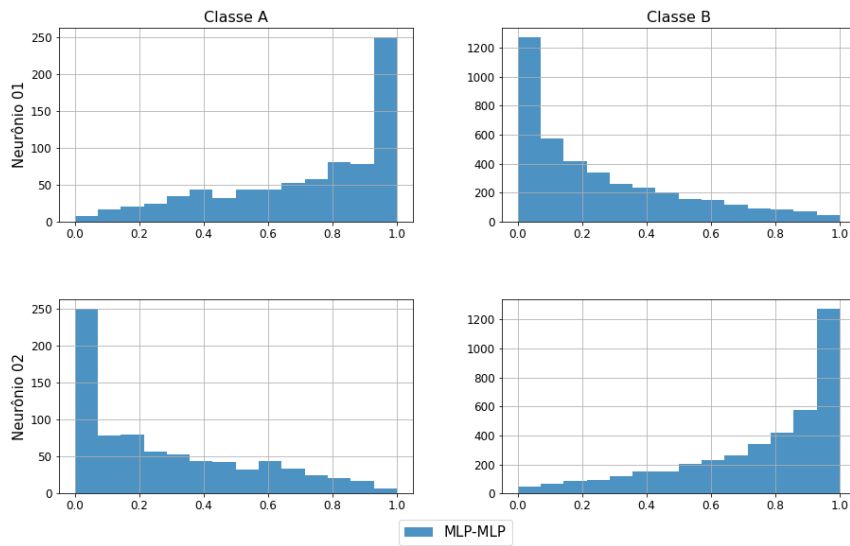
Figura C.99: Histogramas da saída da fusão dos modelos MLP (*folds* 1 e 2).

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 3



(a) *fold-3*

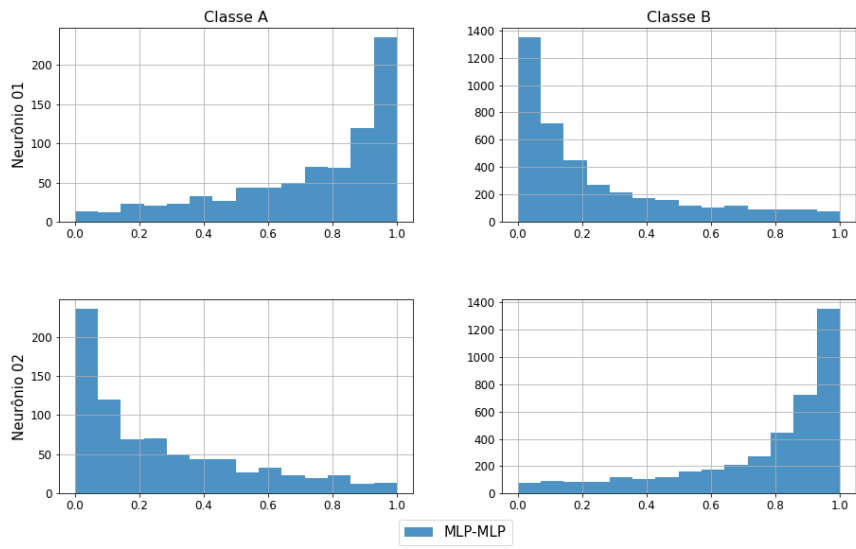
Histogramas da saída da rede MLP-MLP (Fusão) - Fold 4



(b) *fold-4*

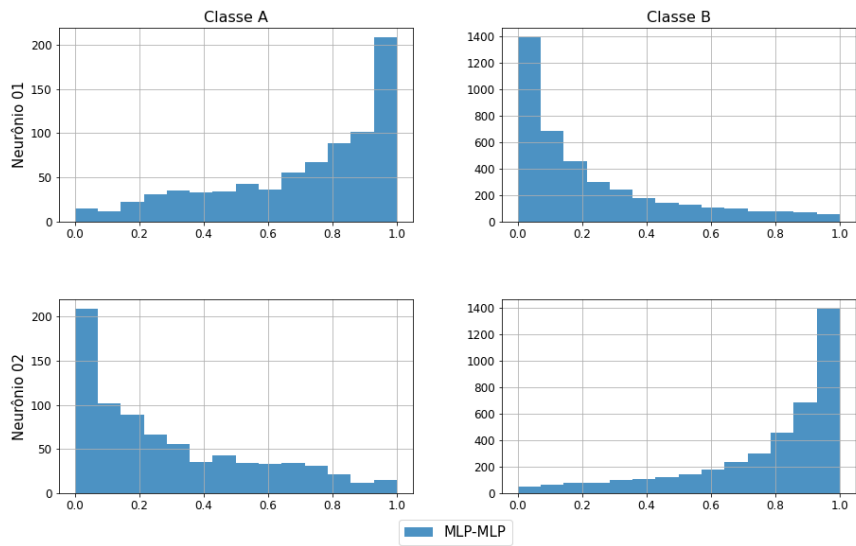
Figura C.100: Histogramas da saída da fusão dos modelos MLP (*folds* 3 e 4).

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 5



(a) *fold-5*

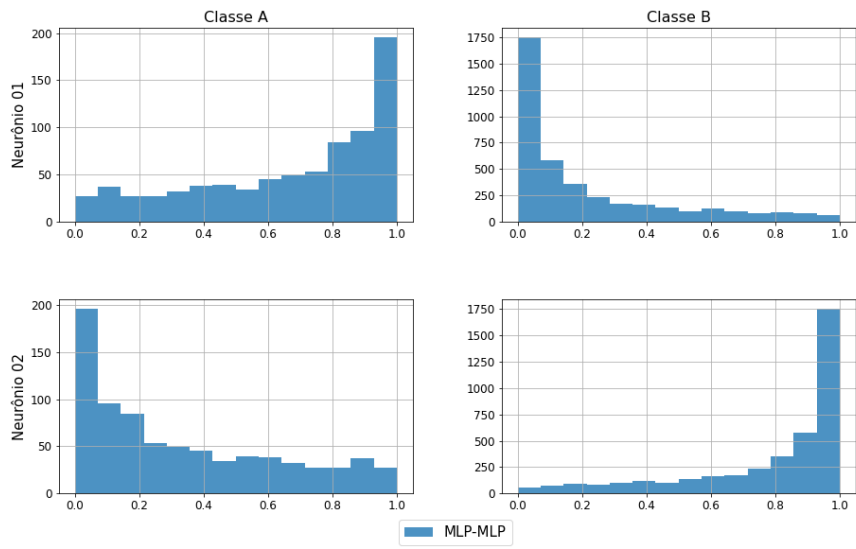
Histogramas da saída da rede MLP-MLP (Fusão) - Fold 6



(b) *fold-6*

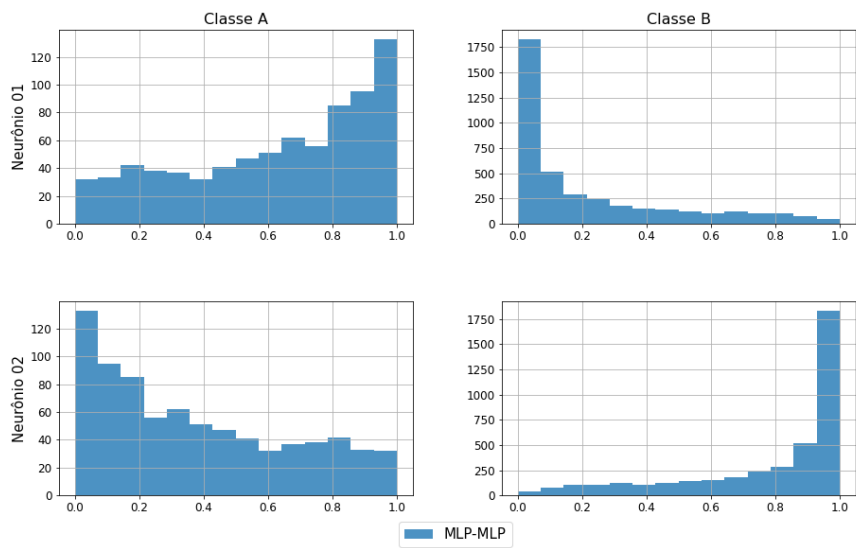
Figura C.101: Histogramas da saída da fusão dos modelos MLP (*fold*s 5 e 6).

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 7



(a) *fold-7*

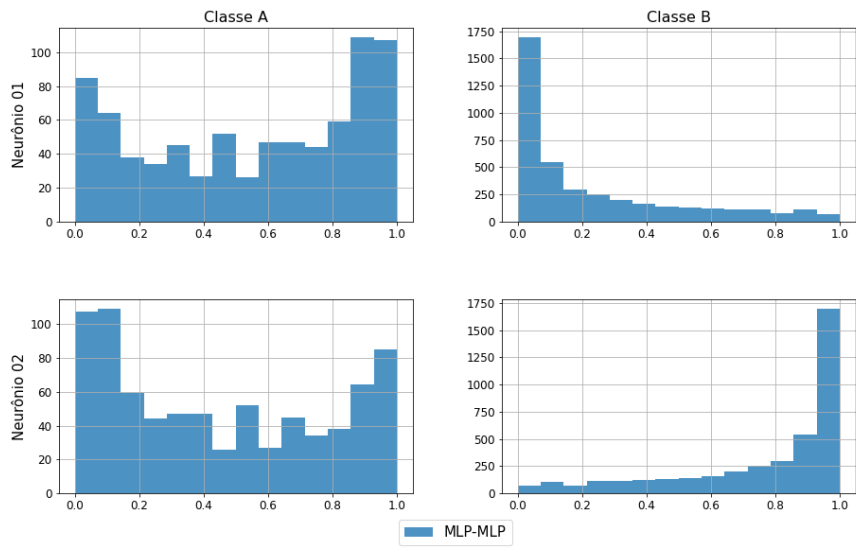
Histogramas da saída da rede MLP-MLP (Fusão) - Fold 8



(b) *fold-8*

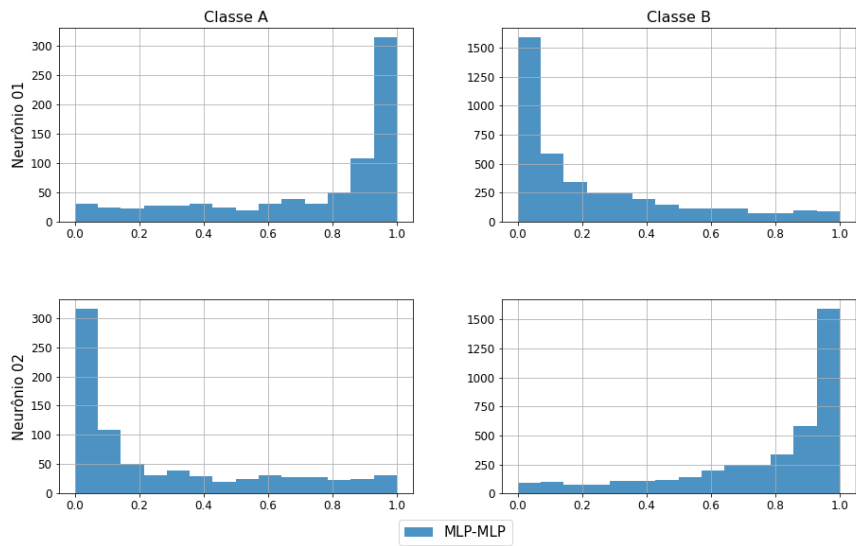
Figura C.102: Histogramas da saída da fusão dos modelos MLP (*fold*s 7 e 8).

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 9



(a) *fold-9*

Histogramas da saída da rede MLP-MLP (Fusão) - Fold 10



(b) *fold-10*

Figura C.103: Histogramas da saída da fusão dos modelos MLP (*folds* 9 e 10).

As Figuras de C.104 a C.108 mostram os histogramas da saída dos 10 *fold*s da fusão dos modelos CNN com e sem a rede de fusão.

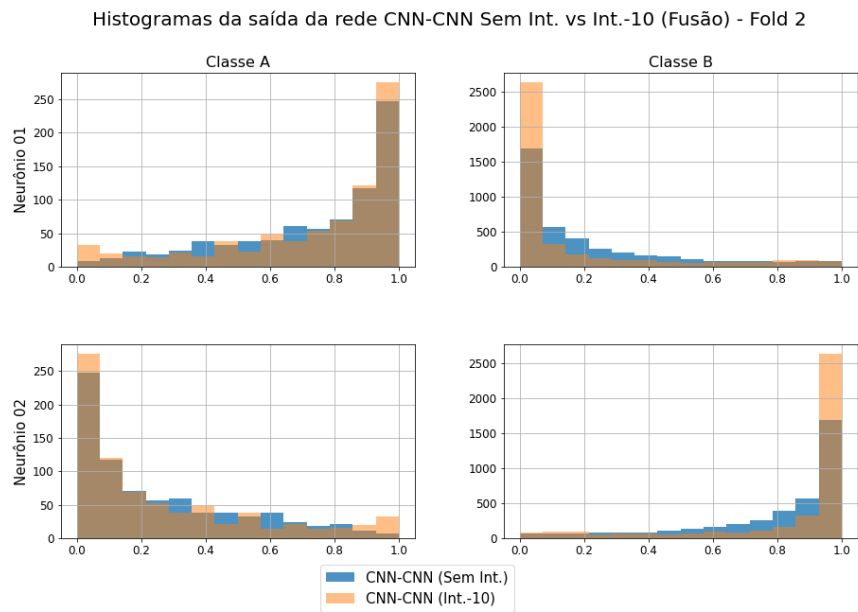
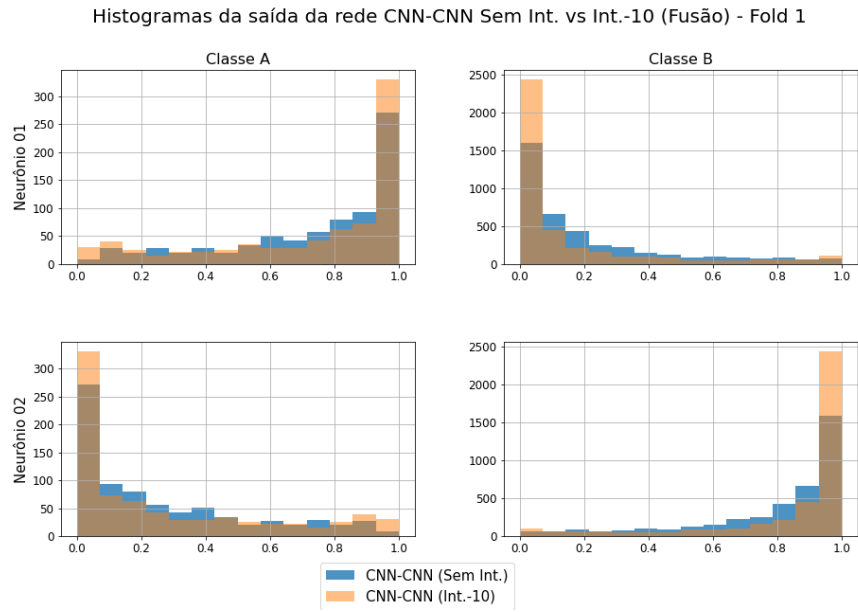
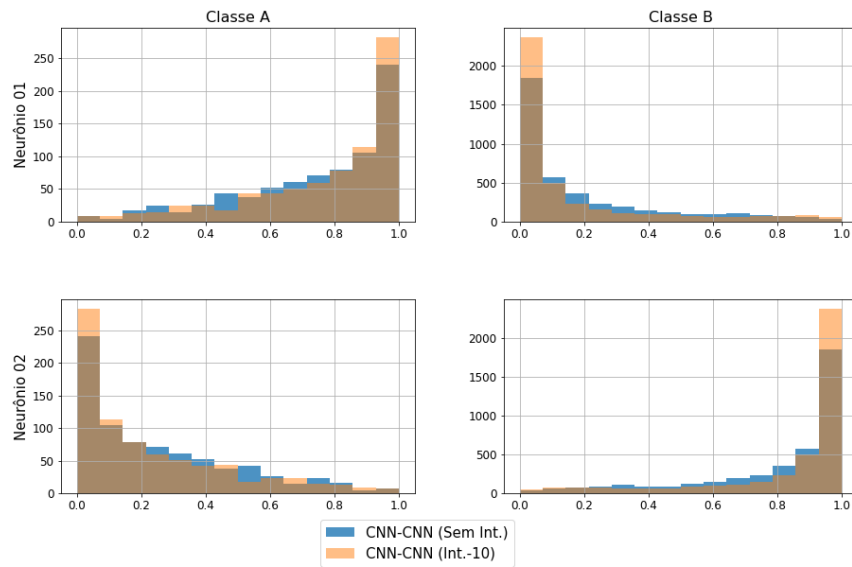


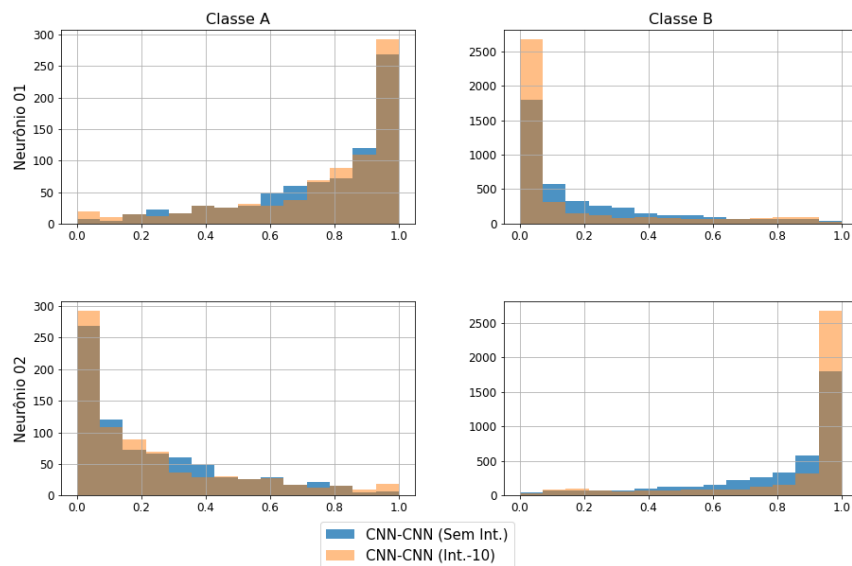
Figura C.104: Histogramas da saída do modelo CNN da fusão (*fold*s 1 e 2).

Histogramas da saída da rede CNN-CNN Sem Int. vs Int.-10 (Fusão) - Fold 3



(a) *fold-3*

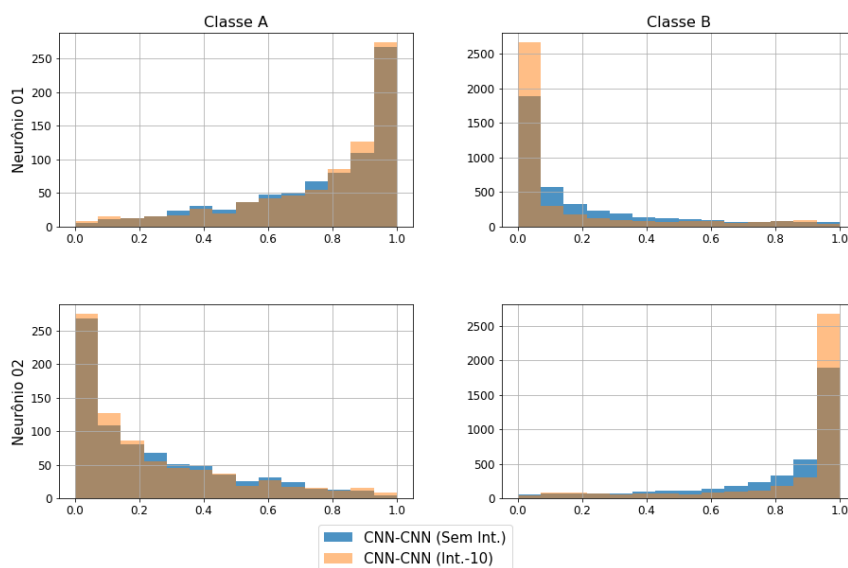
Histogramas da saída da rede CNN-CNN Sem Int. vs Int.-10 (Fusão) - Fold 4



(b) *fold-4*

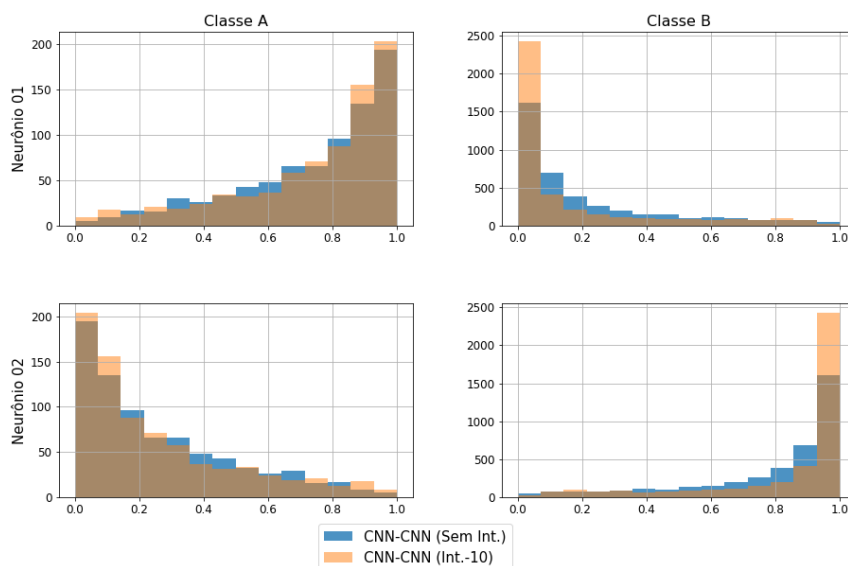
Figura C.105: Histogramas da saída do modelo CNN da fusão (*folds* 3 e 4).

Histogramas da saída da rede CNN-CNN Sem Int. vs Int.-10 (Fusão) - Fold 5



(a) *fold-5*

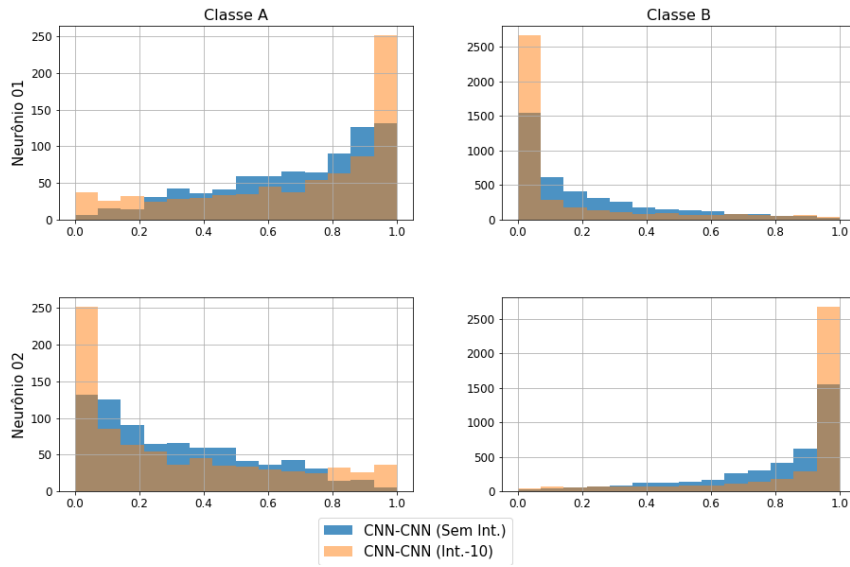
Histogramas da saída da rede CNN-CNN Sem Int. vs Int.-10 (Fusão) - Fold 6



(b) *fold-6*

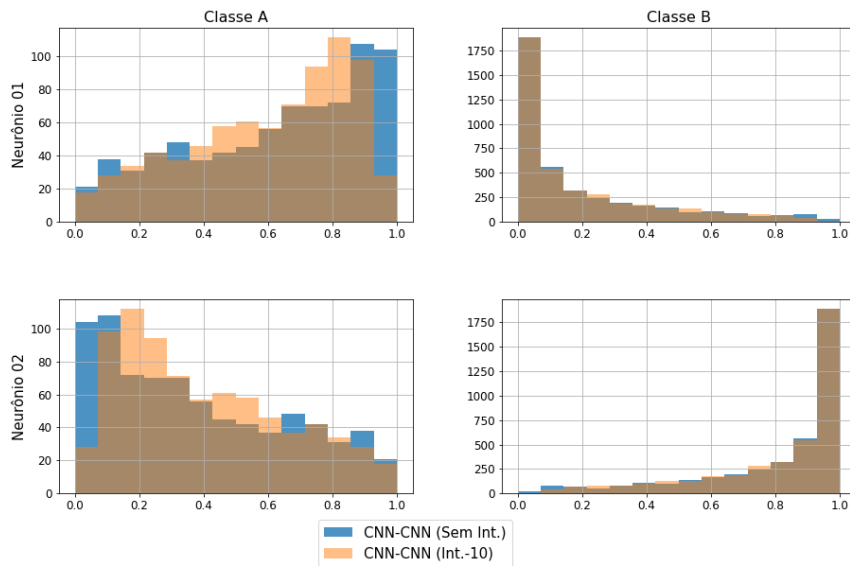
Figura C.106: Histogramas da saída do modelo CNN da fusão (*folds* 5 e 6).

Histogramas da saída da rede CNN-CNN Sem Int. vs Int.-10 (Fusão) - Fold 7



(a) *fold-7*

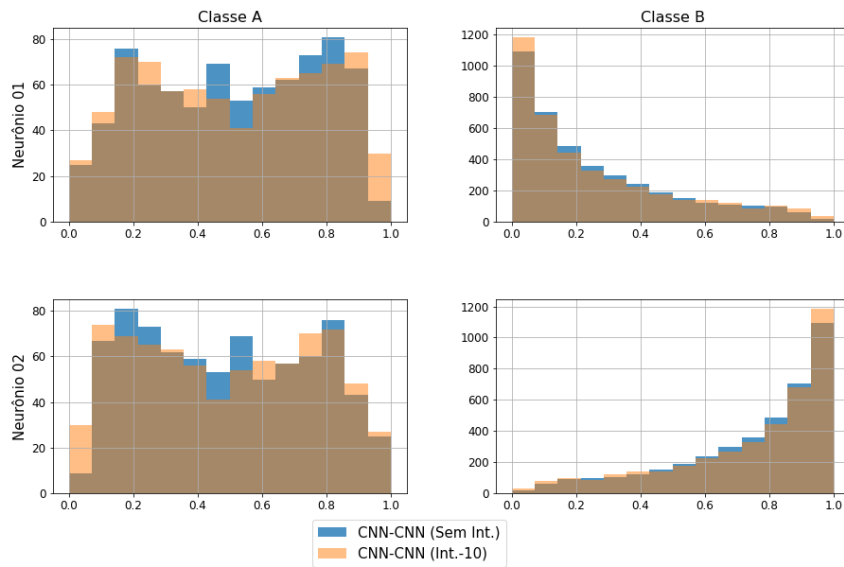
Histogramas da saída da rede CNN-CNN Sem Int. vs Int.-10 (Fusão) - Fold 8



(b) *fold-8*

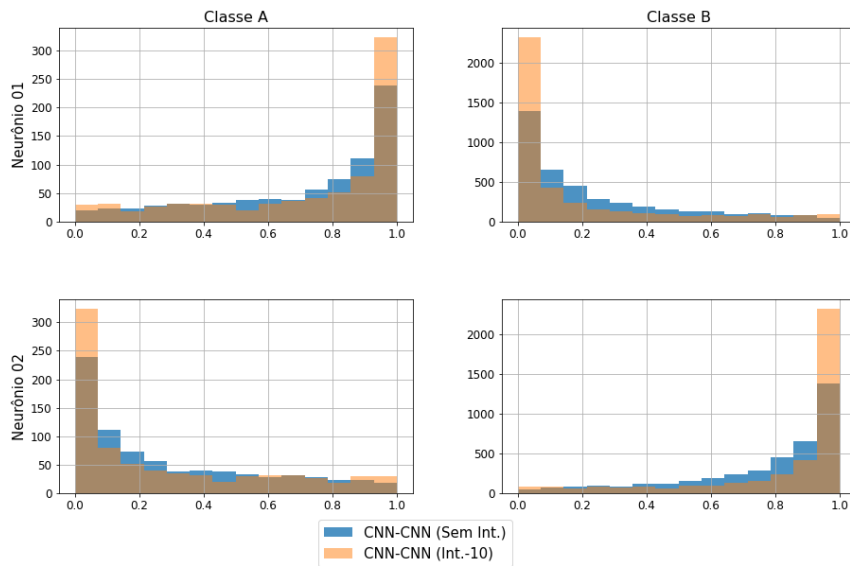
Figura C.107: Histogramas da saída do modelo CNN da fusão (*folds* 7 e 8).

Histogramas da saída da rede CNN-CNN Sem Int. vs Int.-10 (Fusão) - Fold 9



(a) *fold-9*

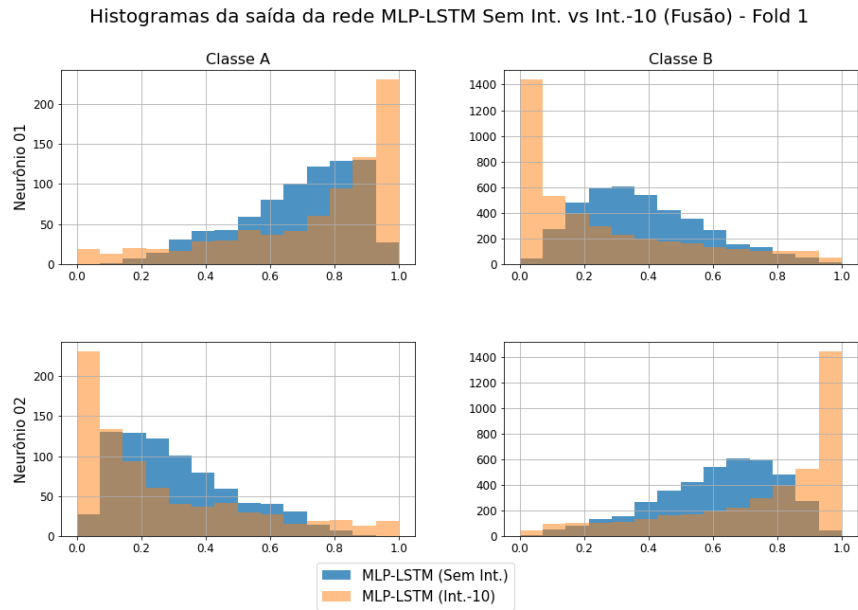
Histogramas da saída da rede CNN-CNN Sem Int. vs Int.-10 (Fusão) - Fold 10



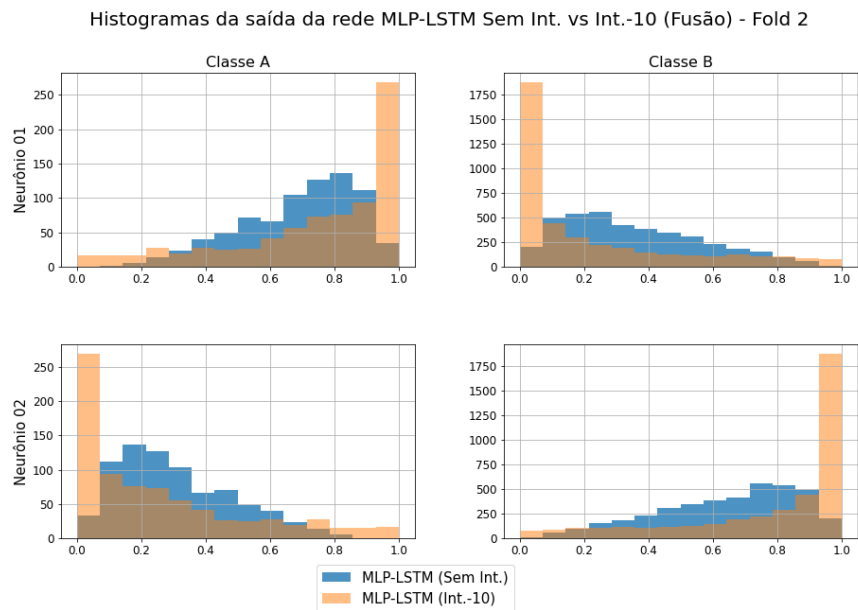
(b) *fold-10*

Figura C.108: Histogramas da saída do modelo CNN da fusão (*folds* 9 e 10).

As Figuras de C.109 a C.113 mostram os histogramas da saída dos 10 *fold*s da fusão dos modelos MLP e LSTM com e sem a rede de fusão.



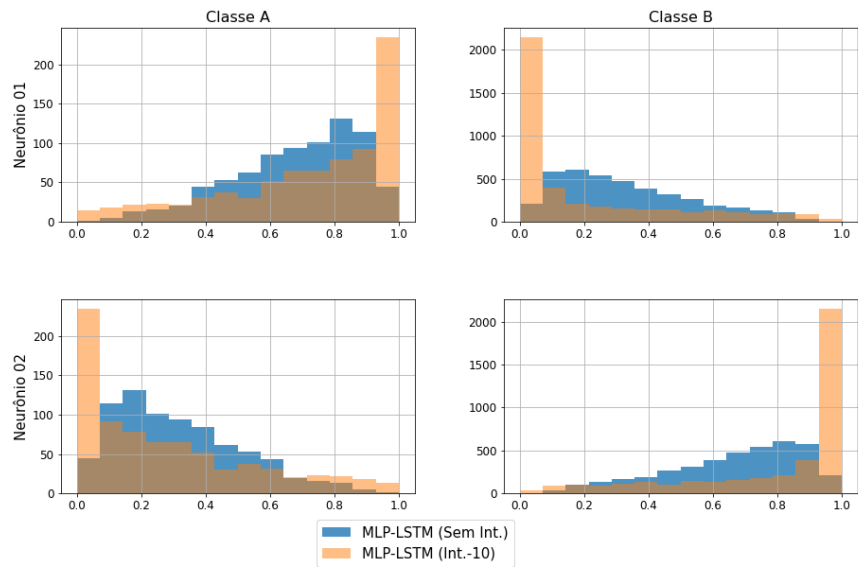
(a) *fold-1*



(b) *fold-2*

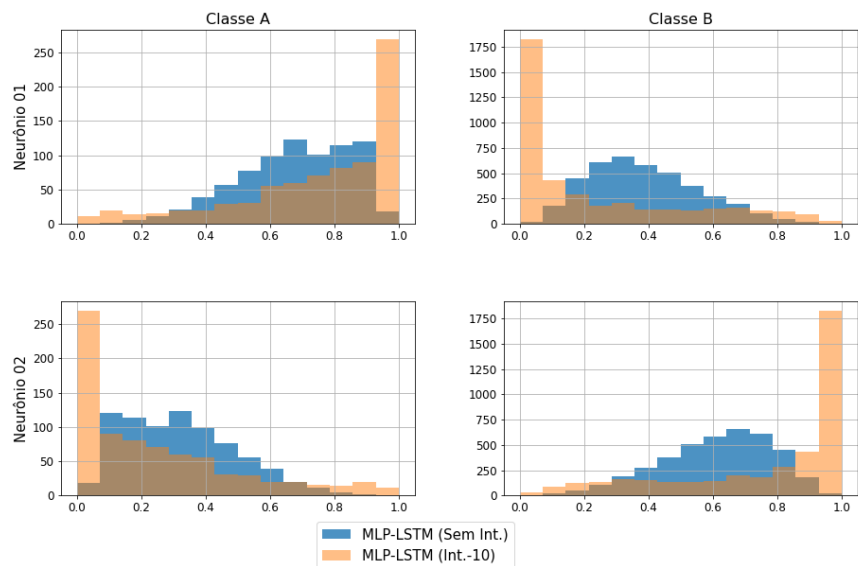
Figura C.109: Histogramas da saída da fusão do modelo MLP com LSTM (*fold*s 1 e 2).

Histogramas da saída da rede MLP-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 3



(a) *fold-3*

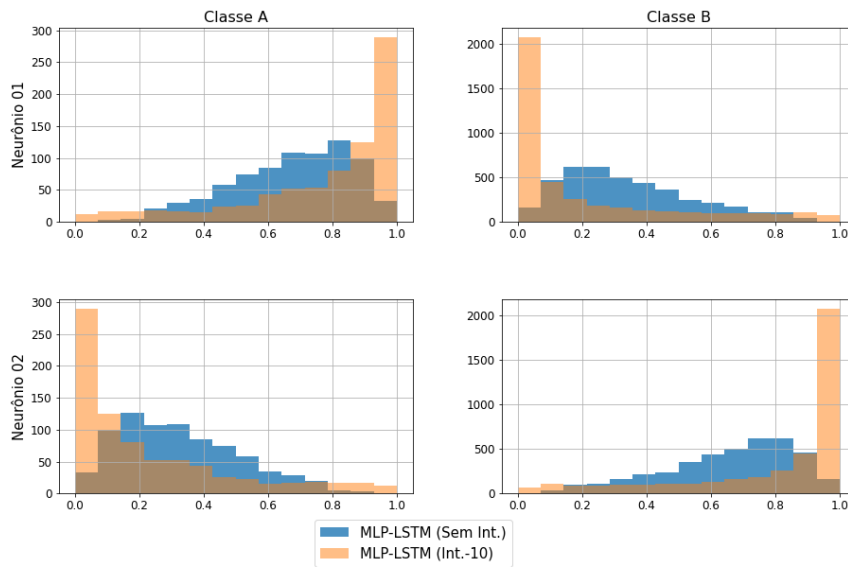
Histogramas da saída da rede MLP-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 4



(b) *fold-4*

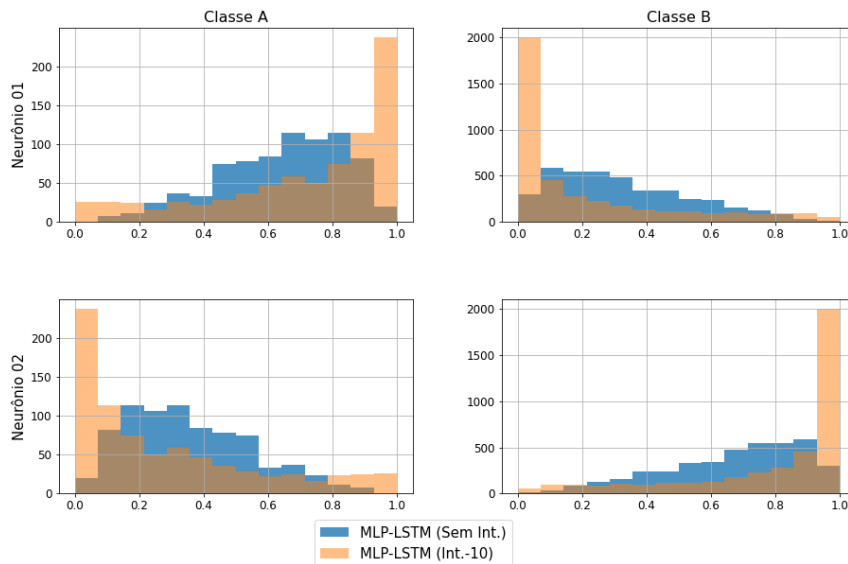
Figura C.110: Histogramas da saída da fusão do modelo MLP com LSTM (*folds* 3 e 4).

Histogramas da saída da rede MLP-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 5



(a) *fold-5*

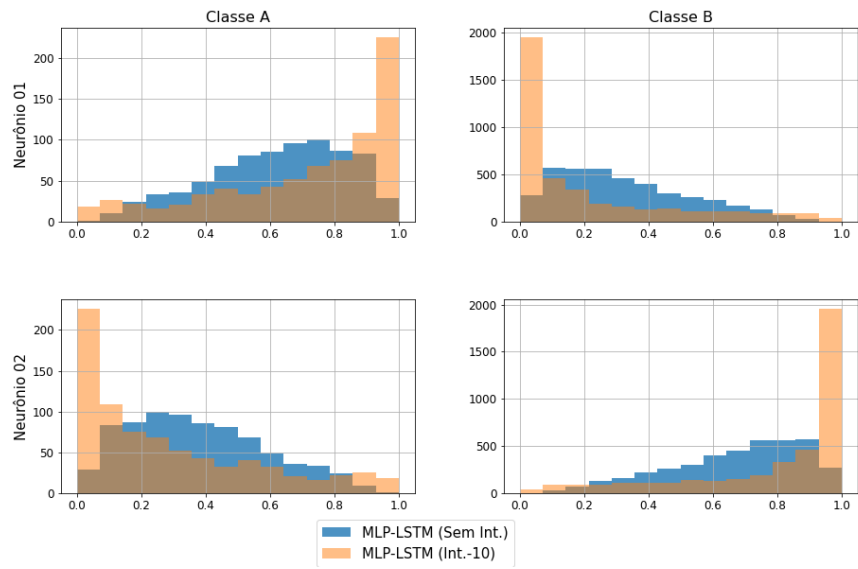
Histogramas da saída da rede MLP-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 6



(b) *fold-6*

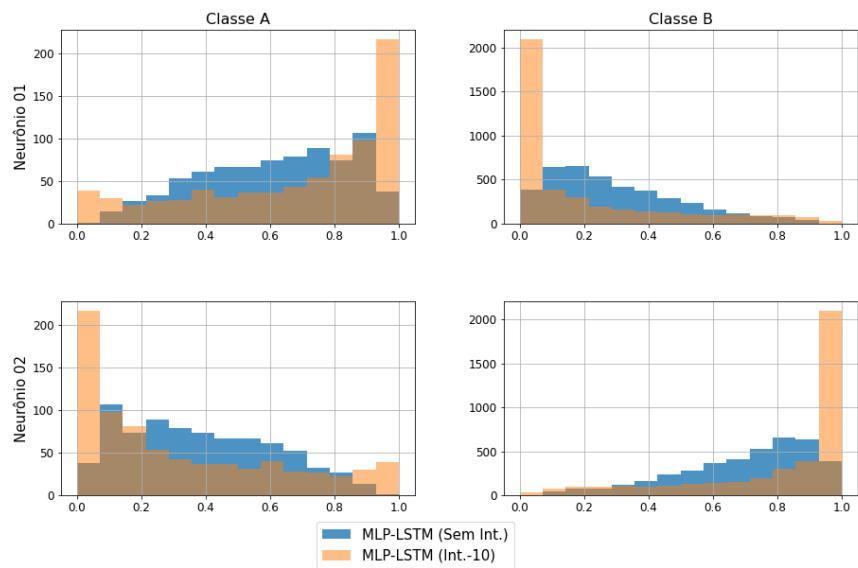
Figura C.111: Histogramas da saída da fusão do modelo MLP com LSTM (*folds* 5 e 6).

Histogramas da saída da rede MLP-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 7



(a) *fold-7*

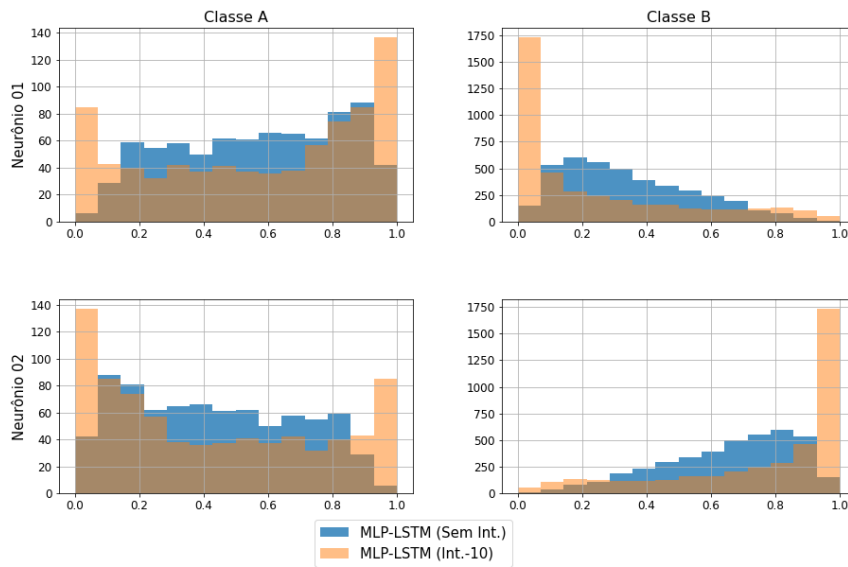
Histogramas da saída da rede MLP-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 8



(b) *fold-8*

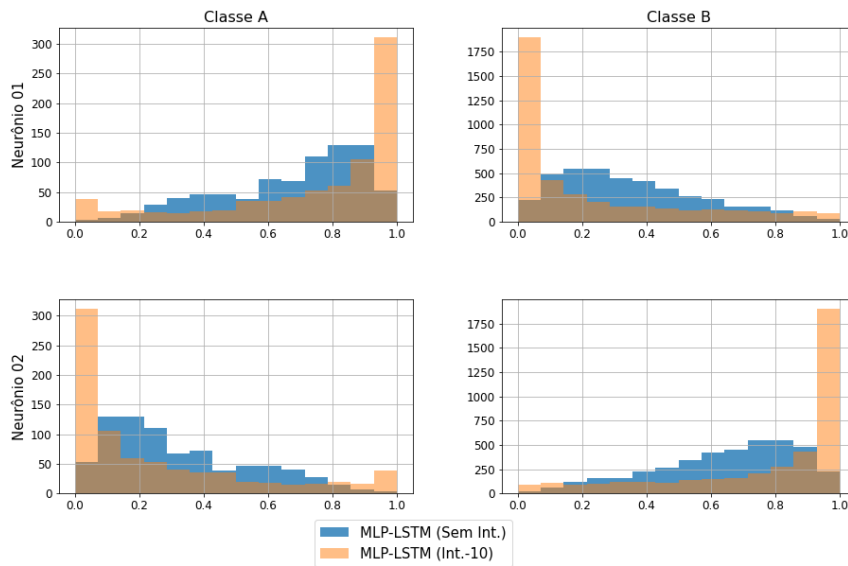
Figura C.112: Histogramas da saída da fusão do modelo MLP com LSTM (*folds* 7 e 8).

Histogramas da saída da rede MLP-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 9



(a) *fold-9*

Histogramas da saída da rede MLP-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 10

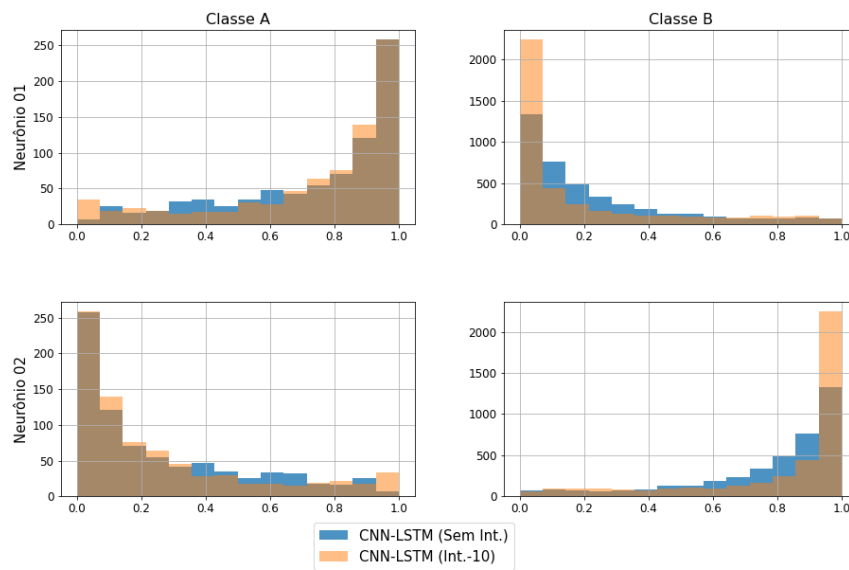


(b) *fold-10*

Figura C.113: Histogramas da saída da fusão do modelo MLP com LSTM (*folders* 9 e 10).

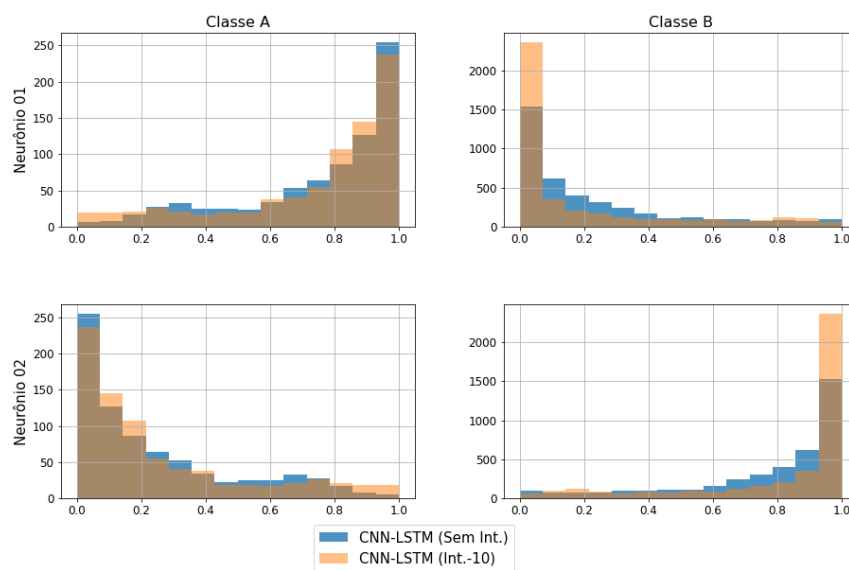
As Figuras de C.114 a C.118 mostram os histogramas da saída dos 10 *fold*s da fusão dos modelos CNN e LSTM com e sem a rede de fusão.

Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 1



(a) *fold-1*

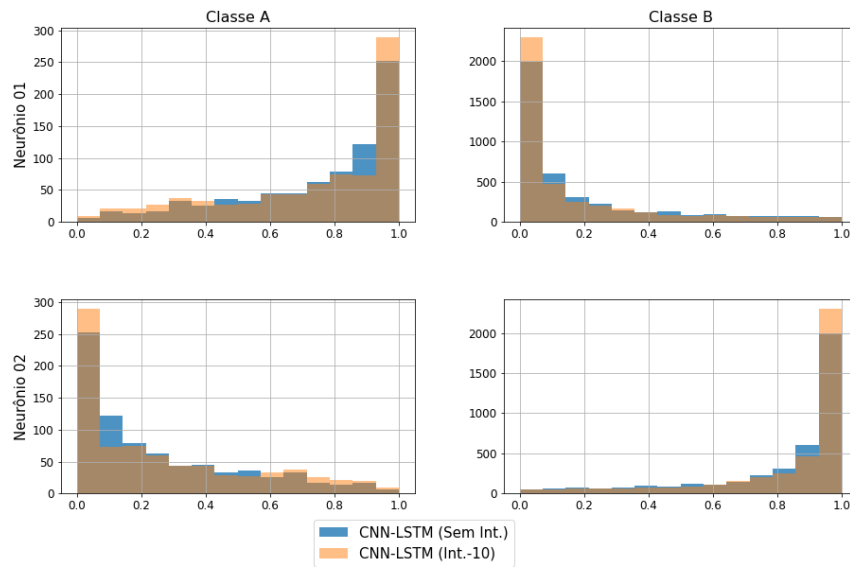
Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 2



(b) *fold-2*

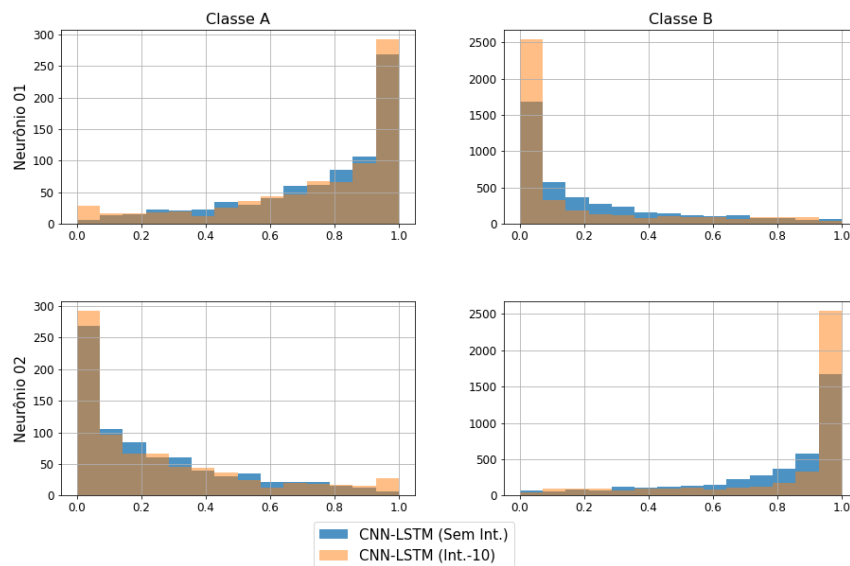
Figura C.114: Histogramas da saída da fusão do modelo CNN com LSTM (*fold*s 1 e 2).

Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 3



(a) *fold-3*

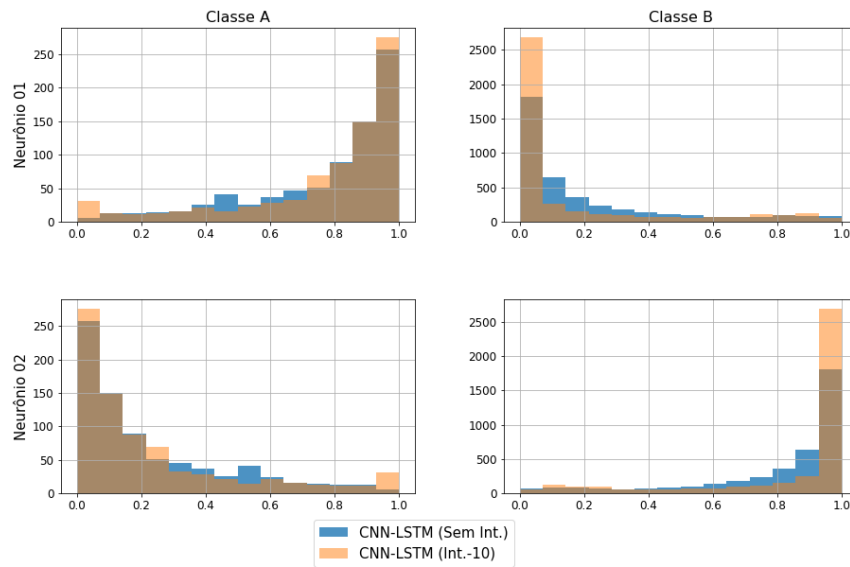
Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 4



(b) *fold-4*

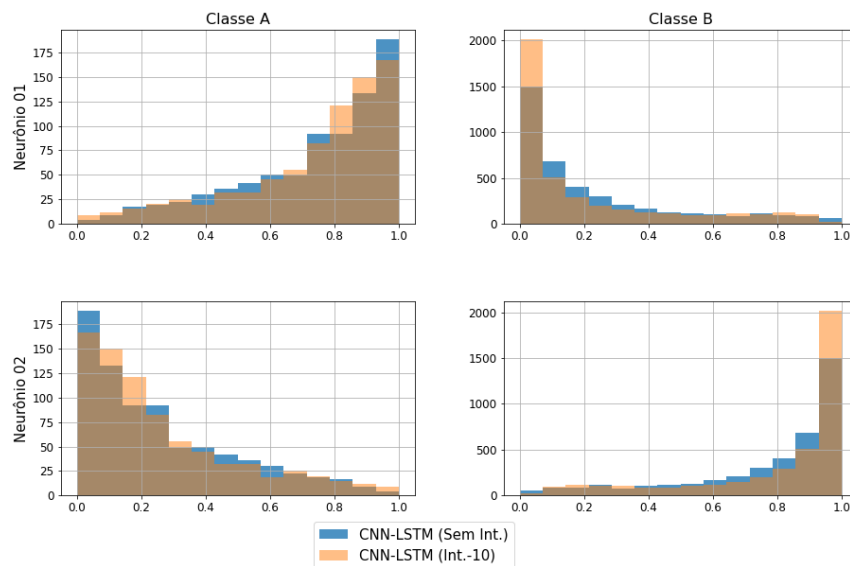
Figura C.115: Histogramas da saída da fusão do modelo CNN com LSTM (*folds* 3 e 4).

Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 5



(a) *fold-5*

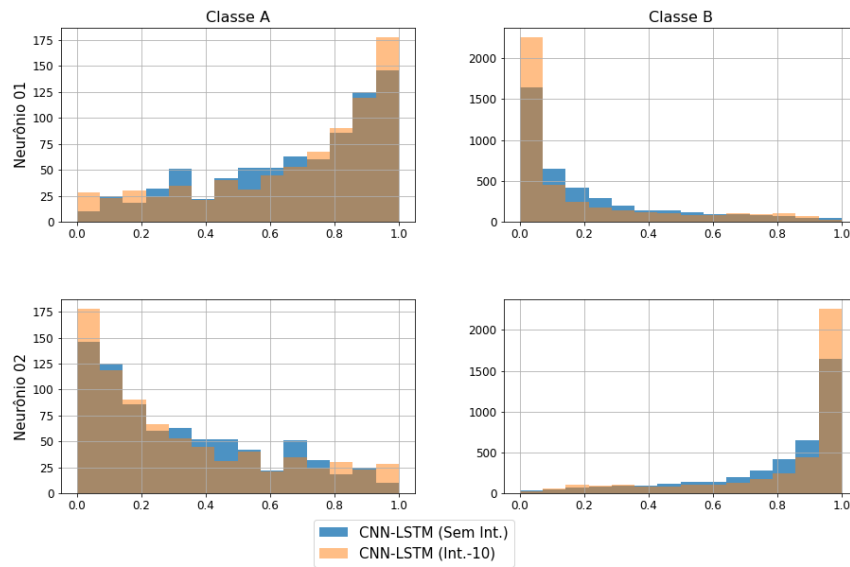
Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 6



(b) *fold-6*

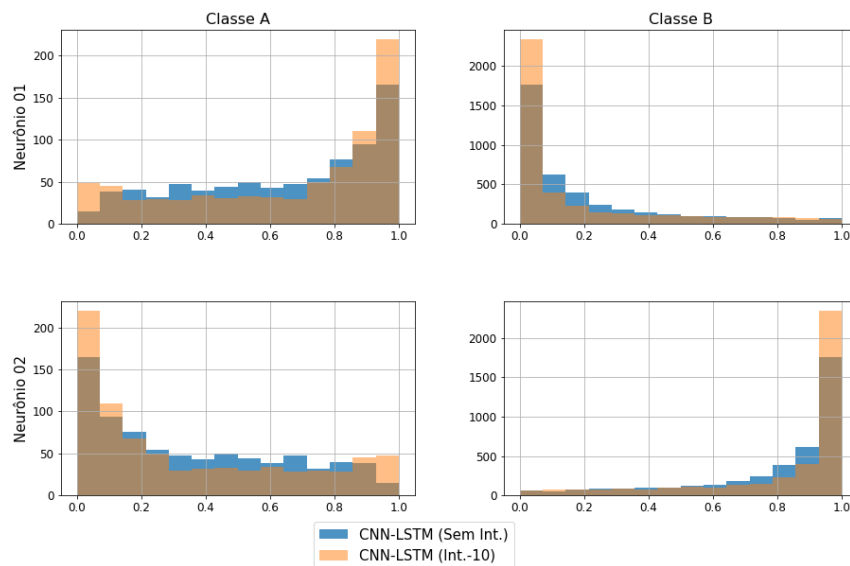
Figura C.116: Histogramas da saída da fusão do modelo CNN com LSTM (*folds* 5 e 6).

Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 7



(a) *fold-7*

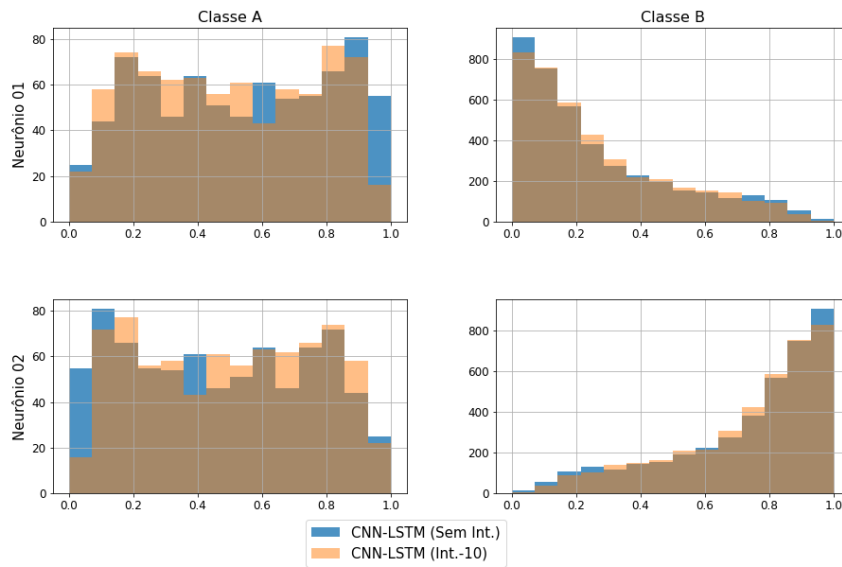
Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 8



(b) *fold-8*

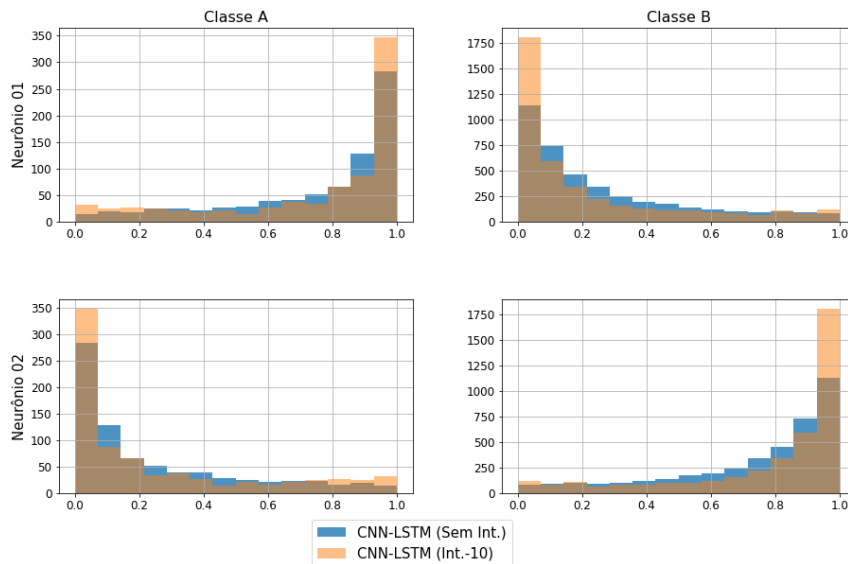
Figura C.117: Histogramas da saída da fusão do modelo CNN com LSTM (*folds* 7 e 8).

Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 9



(a) *fold-9*

Histogramas da saída da rede CNN-LSTM Sem Int. vs Int.-10 (Fusão) - Fold 10



(b) *fold-10*

Figura C.118: Histogramas da saída da fusão do modelo CNN com LSTM (*folds* 9 e 10).

Apêndice D

Publicações

Segue uma descrição dos trabalhos aceitos ou em fase de aceitação para publicação.

D.1 Artigos em Congressos

1. **Título:** Classificação de Sinais Temporais de Sonar Passivo utilizando Redes Neurais Convolucionais.

Autores: Marlon Souza, Natanael Júnior e José Seixas.

Congresso: XV Congresso Brasileiro de Inteligência Computacional, em formato online, Brasil, 2021.

Descrição: O oceano é atualmente explorado em vários níveis. Devido ao interesse crescente pelo mundo no ambiente marinho, surge a necessidade de compreendê-lo melhor através da acústica submarina. Sinais irradiados no meio submarino podem ser detectados e classificados por diversos sistemas. As Redes Neurais Artificiais, em especial as chamadas Redes profundas são ótimas ferramentas de apoio para o estudo desses sinais. Neste artigo é proposto dois modelos de Redes Neurais Artificiais, Multilayer Perceptron e Convolutional Neural Network, a fim de classificar sinais de navios provenientes de um sistema de Sonar Passivo fornecidos pela Marinha do Brasil. Outros trabalhos desenvolvidos com esta base de dados analisam o sinal no domínio da frequência, entretanto este artigo visa uma análise no domínio do Tempo. Os resultados obtidos serão analisados utilizando algumas figuras de méritos já estabelecidas no meio acadêmico. Através delas é possível constatar que o classificador baseado em modelos profundos obteve um melhor desempenho de classificação dos sinais se comparado com o classificador de rede rasas.

2. **Título:** Passive Sonar Classification Using Time-Domain Information and Recurrent Neural Networks.

Autores: Marlon Souza, Natanael Júnior e José Seixas.

Congresso: 8th IEEE Latin American Conference on Computational Intelligence (IEEE LA-CCI 2022), Montevideu-Uruguai, 2022.

Descrição: Sonar systems have widely been used in both military and civilian applications. In particular, passive sonar systems play an important role in submarine operations in any nation's Navy. Usually, passive sonar signal processing is performed in frequency domain for target detection and identification. Alternatively, in this work, a classifier based on recurrent neural networks and fed from the time-domain information is proposed. The proposed model employs Long Short-Term Memory (LSTM) networks aiming at classifying signals coming from 24 classes of military ships, which were organized into 4 super-classes based on expert knowledge. The model achieved an accuracy of $86.03\% \pm 3.08\%$ outperforming a multilayer perceptron network (MLP) baseline model that was fed from frequency-domain data and obtained from Short-Time Fourier transformation.