



UNIVERSIDADE ESTADUAL DE CAMPINAS
Faculdade de Engenharia Elétrica e de Computação

Guilherme Rosa

Aplicação de Modelos Generativos para Síntese de Sinais de EEG em Sistemas BCI-SSVEP

Campinas

2024

Guilherme Rosa

Aplicação de Modelos Generativos para Síntese de Sinais de EEG em Sistemas BCI-SSVEP

Dissertação apresentada à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica, na Área de Engenharia de Computação.

Orientador: Prof. Dr. Levy Boccato

Este exemplar corresponde à versão final da dissertação defendida pelo aluno Guilherme Rosa, e orientada pelo Prof. Dr. Levy Boccato

Campinas

2024

Ficha catalográfica
Universidade Estadual de Campinas (UNICAMP)
Biblioteca da Área de Engenharia e Arquitetura
Rose Meire da Silva - CRB 8/5974

R71a Rosa, Guilherme, 1995-
Aplicação de modelos generativos para síntese de Sinais de EEG em sistemas BCI-SSVEP / Guilherme Rosa. – Campinas, SP : [s.n.], 2024.

Orientador(es): Levy Boccato.
Dissertação (mestrado) – Universidade Estadual de Campinas (UNICAMP), Faculdade de Engenharia Elétrica e de Computação.

1. Interfaces cérebro-computador. 2. Potenciais evocados. 3. Aprendizado profundo. 4. Eletroencefalografia. 5. Redes neurais convolucionais. I. Boccato, Levy, 1986-. II. Universidade Estadual de Campinas (UNICAMP). Faculdade de Engenharia Elétrica e de Computação. III. Título.

Informações complementares

Título em outro idioma: Application of generative models for synthesizing EEG signals in BCI-SSVEP systems

Palavras-chave em inglês:

Brain-computer interface

Evoked potentials

Deep learning

Electroencephalography

Convolutional neural networks

Área de concentração: Engenharia de Computação

Titulação: Mestre em Engenharia Elétrica

Banca examinadora:

Levy Boccato [Orientador]

Romis Ribeiro de Faissol Attux

Thiago Bulhões da Silva Costa

Data de defesa: 27-09-2024

Programa de Pós-Graduação: Engenharia Elétrica

Identificação e informações acadêmicas do(a) aluno(a)

- ORCID do autor: <https://orcid.org/0009-0002-6611-4409>

- Currículo Lattes do autor: <http://lattes.cnpq.br/5880103814874539>

Comissão Julgadora - Dissertação de Mestrado

Candidato: Guilherme Rosa RA: 157955

Data da Defesa: 27 de setembro de 2024

Título da Tese: “Aplicação de Modelos Generativos para Síntese de Sinais de EEG em Sistemas BCI-SSVEP”.

Prof. Dr. Levy Boccato (Presidente, FEEC/UNICAMP)

Prof. Dr. Romis Ribeiro de Faissol Attux (FEEC/UNICAMP)

Prof. Dr. Thiago Bulhões da Silva Costa (UNIFESP)

A ata de defesa, com as respectivas assinaturas dos membros da Comissão Julgadora, encontra-se no SIGA (Sistema de Fluxo de Dissertação/Tese) e na Secretaria de Pós Graduação da Faculdade de Engenharia Elétrica e de Computação.

Agradecimentos

Esta dissertação de mestrado resume toda uma trajetória de estudos cujo apoio e ajuda de várias pessoas foram fundamentais para o desenvolvimento e finalização deste trabalho. Em primeiro lugar, não posso deixar de agradecer ao Professor Doutor Levy Boccato, por ter aceitado me orientar nesse desafio, por toda a paciência e pelo empenho durante esse tempo, tanto no aspecto profissional com o direcionamento do trabalho, quanto no aspecto pessoal, sempre trazendo bons conselhos, motivação e novas perspectivas nos momentos de dificuldade. Também agradeço aos professores doutores Romis Ribeiro de Faissol Attux, Fernando José von Zuben e novamente ao Professor Levy Boccato pelas disciplinas ministradas no tema de inteligência artificial, pois a excelente qualidade do conteúdo e didática foram o que despertaram meu interesse pela área. Gostaria de agradecer a duas pessoas que tiveram grande importância nessa jornada, o colega Thiago Laitz, que iniciou junto comigo o estudo sobre o tema de estudo deste trabalho, e o colega Guilherme Vettorazzi, que nos forneceu um grande apoio durante o processo de submissão do artigo desenvolvido. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

Quero agradecer à minha família pelo apoio incondicional que me deram, especialmente aos meus pais Ulisses e Eliana e ao meu tio Wilson, que desde cedo me ensinaram a importância dos estudos e por nunca medirem esforços para me proporcionar sempre o melhor. Também gostaria de agradecer aos meus irmãos, Gabriel e Flávia, e a todos meus amigos, não apenas pelo apoio mas principalmente pela compreensão dos muitos momentos que precisei me ausentar. Obrigado novamente a todos, pois sem esses apoios nenhuma conquista seria possível.

Resumo

Uma das principais dificuldades no contexto das interfaces cérebro-computador está em obter classificadores de alto desempenho, devido tanto à escassez de dados disponíveis para treinamento quanto à diversidade de comportamentos que o cérebro humano pode apresentar. Diante deste cenário, técnicas de aumento de dados (*data augmentation*) se mostram como um caminho promissor, uma vez que podem sintetizar amostras artificiais úteis para o treinamento de modelos, amenizando o problema de falta de dados. Embora os trabalhos na literatura apresentem resultados importantes com o uso dos dados artificiais, verificamos que todos os trabalhos desenvolvidos até o momento para BCIs-SSVEP tratam exclusivamente da geração e uso dos sinais brutos de EEGs no domínio do tempo, explorando o domínio da frequência apenas para verificar a qualidade dos EEGs sintetizados. Contudo, como o paradigma SSVEP apresenta um apelo ao uso da informação espectral dos eletroencefalogramas (EEGs), já que a estimulação visual induz atividades cerebrais que oscilam nas mesmas frequências dos estímulos, a abordagem do problema no domínio da frequência parece ser mais adequada.

Neste trabalho estendemos a aplicação de modelos generativos para sintetizar não apenas EEGs-SSVEP no domínio do tempo, mas também diretamente no domínio da frequência, produzindo os espectros que serão utilizados como entrada para os classificadores da BCI. Com isso, realizamos um extensivo estudo sobre o problema de classificação que envolve identificar a frequência de evocação dos EEGs-SSVEP abordando: (i) domínio do tempo sem *data augmentation* (desempenho-base), (ii) domínio do tempo com *data augmentation*, (iii) domínio da frequência sem *data augmentation*, (iv) domínio da frequência com *data augmentation* utilizando EEGs-SSVEP sintetizados na frequência e, por fim, (v) domínio da frequência com *data augmentation* utilizando EEGs-SSVEP sintetizados no domínio do tempo e depois convertidos para o domínio da frequência.

Os resultados indicam que a classificação no domínio da frequência tende a apresentar melhores desempenhos do que no domínio do tempo, especialmente quando os espectros de EEGs são bem comportados e apresentam picos nítidos nas frequências de estímulo. Além disso, o uso de dados artificiais nas abordagens com *data augmentation* se mostrou eficaz, promovendo ganhos de desempenho em 80,7% dos cenários investigados, com os maiores desempenhos de validação sendo alcançados no domínio da frequência pelos classificadores convolucionais, CNN-1D, treinados com bases de dados aumentadas formadas por espectros dos EEGs reais de 4 segundos de duração e por espectros dos EEGs-SSVEP de 4 segundos de comprimento sintetizados no domínio do tempo pelas técnicas investigadas.

Palavras-chaves: Interface Cérebro-Computador; SSVEP; Aprendizado Profundo; Modelos Generativos; Data Augmentation.

Abstract

One of the main challenges in the context of brain-computer interfaces lies in obtaining high-performance classifiers, due both to the scarcity of available data for training and the diversity of behaviors that the human brain can exhibit. In this scenario, data augmentation techniques have proven to be a promising path, as they can synthesize artificial samples useful for model training, mitigating the issue of data shortage. Although studies in the literature present significant results with the use of artificial data, we have found that all the work developed so far for BCIs based on the SSVEP paradigm exclusively deals with the generation and use of raw EEG signals in the time domain, exploring the frequency domain only to verify the quality of the synthesized EEGs. However, since the SSVEP paradigm encourages the use of spectral information from electroencephalograms (EEGs), as visual stimulation induces brain activity oscillating at the same frequencies as the stimuli, approaching the problem in the frequency domain seems more appropriate.

In this work, we extend the application of generative models to synthesize not only SSVEP-EEGs in the time domain but also SSVEP-EEGs directly in the frequency domain, producing spectra that will be used as input for the BCI classifiers. Thus, we conduct an extensive study on the classification problem that involves identifying the evoked frequency of SSVEP-EEGs by addressing: (i) the time domain without data augmentation (baseline performance), (ii) the time domain with data augmentation, (iii) the frequency domain without data augmentation, (iv) the frequency domain with data augmentation using SSVEP-EEGs synthesized in the frequency domain, and finally, (v) the frequency domain with data augmentation using SSVEP-EEGs synthesized in the time domain and then converted to the frequency domain.

The results indicate that classification in the frequency domain tends to perform better than that in the time domain, especially when the EEG spectra are well-behaved and show clear peaks at the stimulus frequencies. Additionally, the use of artificial data in data augmentation approaches proved effective, promoting performance gains in 80.7% of the scenarios investigated, with the highest validation performances being achieved in the frequency domain by convolutional classifiers, CNN-1D, trained with augmented datasets formed by spectra of real 4-second EEGs and spectra of 4-second SSVEP-EEGs synthesized in the time domain by the investigated techniques.

Keywords: Brain-computer Interfaces; SSVEP; Deep Learning; Generative Models; Data Augmentation.

Lista de ilustrações

Figura 2.1 – Diagrama geral de uma interface cérebro-computador.	21
Figura 2.2 – Disposição dos eletrodos no escalpo segundo o padrão internacional 10-10. Adaptado de (ASANZA <i>et al.</i> , 2022).	24
Figura 2.3 – Espectro de magnitude de um EEG no contexto de um sistema BCI-SSVEP, no qual o paciente foi exposto a um estímulo visual com frequência de 12 Hz.	25
Figura 2.4 – Figura esquemática representando o monitor utilizado na etapa experimental contendo os padrões xadrez que oscilavam com as frequências de evocação 6 Hz, 10 Hz, 12 Hz e 15 Hz.	26
Figura 2.5 – Exemplos de espectros de EEGs referentes a um paciente exposto a estímulos visuais com frequências de oscilação iguais a: (a) 6 Hz, (b) 10 Hz, (c) 12 Hz e (d) 15 Hz.	26
Figura 2.6 – Exemplos de espectros antes e após a aplicação do filtro CAR em sinais de EEG-SSVEP amostrados durante a etapa de experimentação após submeter um paciente aos estímulos visuais com frequências de oscilação iguais a: (a) 6 Hz, (b) 10 Hz, (c) 12 Hz e (d) 15 Hz.	28
Figura 2.7 – Hiperplano de máxima margem e seus vetores-suporte obtidos com SVM.	31
Figura 2.8 – Diagrama representando a ideia da extensão da SVM para tratar problemas não-lineares no espaço de atributos, mas que se tornam lineares no espaço de características.	32
Figura 2.9 – Diagrama representando um classificador baseado em uma rede neural convolucional 1D.	33
Figura 2.10 – Operação de um kernel aplicando convolução numa amostra de entrada.	34
Figura 2.11 – Diagrama representando o processando realizado pela operação de <i>pooling</i>	35
Figura 2.12 – Diagrama representando uma rede neural com camadas convolucionais transpostas 1D.	36
Figura 2.13 – Operação de um <i>kernel</i> aplicando convolução transposta em uma amostra de entrada.	37
Figura 3.1 – Versão simplificada do <i>autoencoder</i> variacional.	38
Figura 3.2 – Versão simplificada final do <i>autoencoder</i> variacional com a formulação proposta no problema.	39
Figura 3.3 – Versão final do <i>autoencoder</i> variacional com o truque da reparametrização.	40

Figura 3.4 – Modelo generativo obtido com o VAE.	41
Figura 3.5 – Diagrama representando esquematicamente o treinamento de uma GAN.	41
Figura 3.6 – Exemplo de estruturas das redes generativa e discriminativa de uma arquitetura do tipo DCGAN.	44
Figura 3.7 – Diagrama de uma rede do tipo AC-GAN.	46
Figura 3.8 – Diagrama da estrutura geral de um <i>autoencoder</i> adversário.	47
Figura 3.9 – Uso da camada de <i>embedding</i> para incorporar a informação do rótulo da classe.	48
Figura 4.1 – Arquitetura do modelo de classificação com bloco SCU para processamento de EEGs-SSVEP brutos.	56
Figura 4.2 – Diagrama esquemático dos treinamentos realizados no segundo experimento do estudo.	57
Figura 4.3 – Diagrama esquemático dos treinamentos realizados no terceiro experimento do estudo.	58
Figura 4.4 – Diagrama do modelo proposto SIS-GAN.	61
Figura 4.5 – Diagrama esquemático dos treinamentos realizados no primeiro experimento do estudo.	62
Figura 4.6 – Diagrama esquemático dos treinamentos realizados no segundo e terceiro experimento do estudo. No caso do terceiro experimento, os EEGs de todos os usuários S0X foram utilizados para obter os classificadores, enquanto os usuários T0X foram utilizados para validação.	63
Figura 4.7 – Protocolo experimental adotado em (KWON; IM, 2022) para coleta dos EEGs de repouso e os EEGs-SSVEP para uma frequência de evocação durante uma sessão.	65
Figura 4.8 – Diagrama esquemático das redes que compõem a arquitetura S2S-StarGAN.	66
Figura 4.9 – Diagrama esquemático das duas formas de operação do modelo generativo da S2S-StarGAN após sua obtenção.	66
Figura 5.1 – Exemplos de EEGs no domínio do tempo e no domínio da frequência para as frequências de evocação 12 Hz e 15 Hz capturadas pelo eletrodo Oz do usuário S01. As imagens da coluna à esquerda apresentam os EEGs brutos, sem qualquer pré-processamento digital, e as imagens da coluna à direita apresentam os mesmos EEGs após o pré-processamento.	70
Figura 5.2 – Diagrama com as dimensões dos conjuntos de dados, desde a base de dados completa até os conjuntos de treinamento e de validação após cada configuração de particionamento.	71

Figura 5.3 – EEG no domínio do tempo e no domínio da frequência para a frequência de evocação de 12 Hz capturada pelo eletrodo Oz do usuário S01. As figuras (a) e (b) apresentam o EEG bruto de 12 s após o pré-processamento, as figuras (c) e (d) apresentam um exemplo de segmento de 2 s do mesmo EEG, as figuras (e) e (f) apresentam um exemplo de segmento de 3 s do mesmo EEG e as figuras (g) e (h) apresentam um exemplo de segmento de 4 s do mesmo EEG.	72
Figura 5.4 – EEG no domínio do tempo e no domínio da frequência para a frequência de evocação de 15 Hz capturada pelo eletrodo Oz do usuário S01. As figuras (a) e (b) apresentam o EEG bruto de 12 s após o pré-processamento, as figuras (c) e (d) apresentam um exemplo de segmento de 2 s do mesmo EEG, as figuras (e) e (f) apresentam um exemplo de segmento de 3 s do mesmo EEG e as figuras (g) e (h) apresentam um exemplo de segmento de 4 s do mesmo EEG.	73
Figura 5.5 – Diagrama esquemático dos treinamentos sem <i>data augmentation</i> dos classificadores SVM linear, SVM não-linear e CNN-1D nos domínios do tempo e da frequência com EEGs-SSVEP de 4 segundos de duração do participante S01.	90
Figura 5.6 – Diagrama esquemático dos treinamentos com <i>data augmentation</i> dos classificadores SVM linear, SVM não-linear e CNN-1D nos domínios do tempo e da frequência com EEGs-SSVEP de 4 segundos de duração do participante S01. Neste diagrama utilizamos como exemplo os modelos generativos obtidos a partir do treinamento da DC-GAN.	91
Figura 6.1 – Mapa de calor com os ganhos e as perdas de desempenho quando comparamos às performances dos classificadores no domínio da frequência com seus respectivos desempenhos base no domínio do tempo.	97
Figura 6.2 – Média dos espectros dos EEGs-SSVEP de 12 Hz e 15 Hz do conjunto de treinamento dos usuários S01, S02, S05, S06 e S10	98
Figura 6.3 – Número de vezes que cada um dos classificadores treinados sem <i>data augmentation</i> alcançou os maiores desempenhos de validação para cada cenário abordado da tarefa de identificar a frequência de evocação dos EEGs-SSVEP.	98
Figura 6.4 – Quantidade de vezes que cada comprimento de EEG investigado levou os classificadores treinados sem <i>data augmentation</i> a alcançar os maiores desempenhos de validação nos domínios do tempo e da frequência para a tarefa de identificar a frequência de evocação dos EEGs-SSVEP.	99
Figura 6.5 – Ganhos de desempenho no domínio do tempo promovidos pelo uso de dados artificiais para obtenção dos classificadores da BCI.	101

Figura 6.6 – Quantidade de cenários que o uso dos EEGs-SSVEP artificiais sintetizados por cada modelo generativo investigado levou a ganhos de desempenho para os classificadores SVM linear, SVM não-linear e CNN-1D no domínio do tempo.	102
Figura 6.7 – Ganhos de desempenho no domínio da frequência promovidos pelo uso dos espectros de EEGs sintetizados diretamente no domínio da frequência para obtenção dos classificadores da BCI.	104
Figura 6.8 – Quantidade de cenários que o uso dos espectros dos EEGs-SSVEP artificiais sintetizados no domínio da frequência por cada modelo generativo investigado levou a ganhos de desempenho para os classificadores SVM linear, SVM não-linear e CNN-1D.	104
Figura 6.9 – Ganhos de desempenho no domínio da frequência promovidos pelo uso dos espectros de EEGs sintetizados no domínio do tempo para obtenção dos classificadores da BCI.	106
Figura 6.10–Quantidade de cenários que o uso dos espectros dos EEGs-SSVEP artificiais sintetizados no domínio do tempo por cada modelo generativo investigado levou a ganhos de desempenho para os classificadores SVM linear, SVM não-linear e CNN-1D.	107
Figura 6.11–Ganhos de desempenho proporcionados pelo uso dos EEGs de 2 segundos sintetizados por cada modelo generativo quando comparamos aos maiores desempenhos sem <i>data augmentation</i>	108
Figura 6.12–Ganhos de desempenho proporcionados pelo uso dos EEGs de 3 segundos sintetizados por cada modelo generativo quando comparamos aos maiores desempenhos sem <i>data augmentation</i>	109
Figura 6.13–Ganhos de desempenho proporcionados pelo uso dos EEGs de 4 segundos sintetizados por cada modelo generativo quando comparamos aos maiores desempenhos sem <i>data augmentation</i>	110
Figura 6.14–Comparativo dos desempenhos médios dos classificadores SVM linear, SVM não-linear e CNN-1D sobre os 10 participantes da base de dados em função das abordagens sem e com <i>data augmentation</i> investigadas.	111
Figura 6.15–Exemplos de amostras reais do paciente S01 nos domínios do tempo e da frequência para as frequências de evocação de 12 Hz e 15 Hz.	112
Figura 6.16–Exemplos de amostras artificiais do paciente S01 nos domínios do tempo e da frequência para a frequência de evocação de 12 Hz.	113
Figura 6.17–Exemplos de amostras artificiais do paciente S01 nos domínios do tempo e da frequência para a frequência de evocação de 15 Hz.	114
Figura 6.18–Exemplos de amostras reais do paciente S10 nos domínios do tempo e da frequência para as frequências de evocação de 12 Hz e 15 Hz.	116

Figura 6.19–Exemplos de amostras artificiais do paciente S10 nos domínios do tempo e da frequência para a frequência de evocação de 15 Hz.	117
Figura 6.20–Exemplos de amostras artificiais do paciente S10 nos domínios do tempo e da frequência para a frequência de evocação de 12 Hz.	118

Lista de tabelas

Tabela 4.1 – Quadro resumo da distribuição dos 53 trabalhos compilados em (LASH-GARI <i>et al.</i> , 2020) com relação ao tipo de tarefa de classificação da BCI, arquitetura de classificador, formato do dado de entrada e presença ou ausência de pré-processamento.	52
Tabela 4.2 – Quadro resumo da distribuição dos 53 trabalhos compilados em (LASH-GARI <i>et al.</i> , 2020) em função do tipo de técnica de <i>data augmentation</i> utilizada.	53
Tabela 4.3 – Tabela resumo com a estrutura das redes de cada uma das arquiteturas de modelo generativo - DCGAN, WGAN e VAE.	56
Tabela 4.4 – Tabela resumo com as estruturas das redes generativas, discriminativas e classificadoras das redes DCGAN, AC-GAN e SIS-GAN.	60
Tabela 5.1 – Conjunto de hiperparâmetros utilizados durante o processo de treinamento e obtenção dos modelos de classificação baseados em SVM Linear e SVM Não-Linear.	74
Tabela 5.2 – Procedimento para obtenção dos modelos de classificação baseados em SVM com EEGs no domínio do tempo para um usuário.	75
Tabela 5.3 – Estrutura e hiperparâmetros do modelo de classificação CNN-1D utilizado no experimento. Alguns hiperparâmetros variam de acordo com o domínio investigado e com a duração do EEG.	76
Tabela 5.4 – Procedimento geral para obtenção dos classificadores CNN-1D para um usuário, duração e domínio do EEG.	77
Tabela 5.5 – Estrutura e hiperparâmetros do modelo generativo DC-GAN utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos. . .	78
Tabela 5.6 – Procedimento para treinamento de modelos DCGAN para um usuário utilizando EEGs com duração e domínio únicos.	79
Tabela 5.7 – Estrutura e hiperparâmetros do modelo generativo WGAN utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos. . .	80
Tabela 5.8 – Procedimento para treinamento de modelos WGAN para um usuário utilizando EEGs com duração e domínio únicos.	81
Tabela 5.9 – Estrutura e hiperparâmetros do modelo generativo VAE utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos. . .	82

Tabela 5.10–Procedimento para treinamento de modelos VAE para um usuário utilizando EEGs com duração e domínio únicos.	83
Tabela 5.11–Estrutura e hiperparâmetros do modelo generativo AAE utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos. . .	84
Tabela 5.12–Procedimento para treinamento de modelos AAE para um usuário utilizando EEGs com duração e domínio únicos.	85
Tabela 5.13–Estrutura e hiperparâmetros do modelo generativo AC-GAN utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos. . .	86
Tabela 5.14–Alterações na estrutura dos modelos generativos para incorporação da informação da classe para o treinamento dos modelos condicionados cDCGAN, cWGAN, cVAE e cAAE.	87
Tabela 5.15–Procedimento para treinamento de modelos AC-GAN para um usuário utilizando EEGs com duração e domínio únicos.	88
Tabela 6.1 – Desempenhos dos classificadores treinados somente com EEGs-SSVEP reais no domínio do tempo quando avaliados utilizando os dados de validação.	95
Tabela 6.2 – Desempenhos dos classificadores treinados somente com espectros dos EEGs-SSVEP reais no domínio da frequência quando avaliados utilizando os espectros dos EEGs-SSVEP do conjunto de validação.	96
Tabela 6.3 – Maiores desempenhos para os modelos de classificação obtidos com o uso de EEGs-SSVEP artificiais sintetizados no domínio do tempo para o problema de classificar os sinais no domínio do tempo.	100
Tabela 6.4 – Maiores desempenhos para os modelos de classificação obtidos com o uso de EEGs-SSVEP artificiais sintetizados no domínio da frequência para o problema de classificar os sinais no domínio da frequência.	103
Tabela 6.5 – Maiores desempenhos para os modelos de classificação obtidos com o uso de EEGs-SSVEP artificiais sintetizados no domínio do tempo para o problema de classificar os sinais no domínio da frequência.	105

Sumário

1	Introdução	17
1.1	Organização do Trabalho	19
2	Fundamentos de interfaces cérebro-computador	20
2.1	Funcionamento geral das BCIs	20
2.2	Métodos de aquisição dos sinais cerebrais	21
2.3	Paradigmas de sistemas BCI	24
2.4	Módulo de processamento de sinais	27
2.4.1	Pré-processamento	27
2.4.2	Extração de características e seleção de atributos	29
2.4.3	Classificação	30
2.5	Fechamento	37
3	Fundamentos dos modelos generativos	38
3.1	Autoencoder Variacional (VAE)	38
3.2	Redes Adversárias Generativas (GAN)	41
3.2.1	<i>Deep Convolutional GAN</i> (DCGAN)	42
3.2.2	<i>Wasserstein GAN</i> (WGAN)	44
3.2.3	<i>Auxiliary Classifier GAN</i> (AC-GAN)	45
3.2.4	<i>Adversarial Autoencoder</i> (AAE)	46
3.3	Modelos generativos condicionados	47
3.4	Fechamento	49
4	<i>Data augmentation</i> aplicado a BCIs-SSVEP	50
4.1	<i>Data augmentation</i> em BCIs com EEGs	52
4.2	<i>Data augmentation</i> em BCIs-SSVEP com EEGs	54
4.2.1	Trabalho de (AZNAN <i>et al.</i> , 2019)	54
4.2.2	Artigo de (AZNAN <i>et al.</i> , 2020)	58
4.2.3	Artigo de (KWON; IM, 2022)	64
4.3	Proposta de Trabalho	67
5	Metodologia	69
5.1	Base de dados	69
5.2	Estruturas e hiperparâmetros dos modelos de classificação	74
5.3	Estruturas e hiperparâmetros dos modelos generativos	77
5.4	Procedimento experimental	89
5.5	Fechamento	93
6	Resultados e discussões	94

6.1	Resultados dos classificadores treinados somente com EEGs-SSVEP reais	95
6.2	Resultados dos classificadores treinados com bases de dados aumentadas artificialmente no domínio do tempo	100
6.3	Resultados dos classificadores treinados com bases de dados aumentadas artificialmente no domínio da frequência com EEGs sintetizados no domínio da frequência	102
6.4	Resultados dos classificadores treinados com bases de dados aumentadas artificialmente no domínio da frequência com EEGs sintetizados no domínio do tempo	105
6.5	Resultados finais das abordagens com <i>data augmentation</i> nos domínios do tempo e da frequência	107
6.6	Exemplos de amostras de EEGs-SSVEP artificiais	111
7	Conclusão	120
	Referências	122

1 Introdução

Uma interface cérebro-computador (BCI, do inglês *Brain Computer Interface*) é um sistema de comunicação no qual as intenções do usuário são transmitidas para o computador a partir da extração e do processamento dos registros de sua atividade cerebral, e não a partir das vias convencionais de saída do sistema nervoso, como os nervos periféricos e músculos (WOLPAW, 2002). Tipicamente, uma BCI é formada por um módulo de monitoramento constituído por um conjunto de sensores capazes de capturar a atividade cerebral do usuário, por um módulo de conversão analógico-digital encarregado de digitalizar os registros da atividade cerebral monitorada e por um módulo de processamento responsável por aplicar uma sequência de transformações sobre os sinais capturados com o objetivo de extrair a intenção do usuário da BCI e, conseqüentemente, tomar a ação desejada.

A pesquisa na área de sistemas BCI teve um grande desenvolvimento devido à motivação de parte da comunidade científica em desenvolver aplicações biomédicas assistivas, pois a comunicação direta entre cérebro e máquina vislumbrada por esses sistemas possibilitaria que pessoas com alguma forma de deficiência fossem capazes de controlar dispositivos sem exercer qualquer ação motora como, por exemplo, na operação de uma cadeira de rodas, na movimentação de próteses mecânicas ou no controle de qualquer outro dispositivo capaz de promover maior acessibilidade, autonomia e qualidade de vida para o usuário (LIMCHESING *et al.*, 2021).

Além da motivação social, diversos fatores contribuíram para o desenvolvimento da pesquisa na área de BCIs, tais como: (i) o uso da eletroencefalografia para captura da atividade cerebral do usuário; (ii) a evolução do processamento através de técnicas de filtragem; (iii) o avanço da inteligência artificial, em especial dos modelos de aprendizado de máquina e de aprendizado profundo para tomada de decisão; (iv) o aumento da disponibilidade e acessibilidade a recursos computacionais.

Atualmente um dos principais obstáculos para o desenvolvimento de BCIs é a relativa escassez de dados para o treinamento de modelos de classificação. Tipicamente, os protocolos para coleta de sinais cerebrais preveem a realização de múltiplas sessões com diferentes usuários, nas quais cada pessoa cumpre uma sequência de etapas para que se monitore sua atividade cerebral nos casos de interesse daquele sistema BCI. Por exemplo, no contexto de SSVEP, o usuário deve concentrar o olhar para cada estímulo visual repetidas vezes (se possível, em sessões realizadas em momentos distintos). Por conta disso, o processo completo de coleta é relativamente lento e custoso, o que explica

a limitação na quantidade de amostras obtidas.

A diversidade de comportamentos do cérebro humano, que pode ser causada por mudanças fisiológicas, de ambiente ou por alterações no estado físico, mental e emocional do usuário, impõe um desafio adicional ao projeto de uma BCI, pois a significativa variação nas características dos sinais – para um mesmo usuário e entre usuários – leva a uma degradação da capacidade de generalização do classificador empregado na BCI. Assim, uma BCI obtida para um usuário em um determinado ambiente e momento pode não apresentar desempenho satisfatório em outros ambientes ou até mesmo em outros momentos no mesmo ambiente, tornando necessária a aquisição de novas amostras e subsequente retreinamento do modelo para cada ambiente no qual a BCI é utilizada.

No entanto, a frequente necessidade de novas amostras é limitada pelo processo de captura dos sinais cerebrais, o qual, conforme já destacado, é oneroso, demorado e cansativo para o usuário, pois envolve a preparação do usuário, caso não seja experiente no uso desse tipo de dispositivo, algumas repetições do experimento para que uma quantidade maior de amostras seja atingida, a fim de que o treinamento do modelo seja efetivo, e, finalmente, a calibração da BCI.

Para diminuir o problema da escassez de dados de treinamento, diversas abordagens têm sido desenvolvidas e aplicadas ao contexto de BCIs. Técnicas de *feature engineering*, por exemplo, são amplamente empregadas para extrair e selecionar os atributos mais relevantes e discriminativos dos sinais, de modo que a dimensão dos dados seja reduzida e, conseqüentemente, a necessidade por grandes quantidades de amostras seja amenizada. Nesta abordagem podem ser utilizadas técnicas como a transformada de Fourier, transformada de Wavelet, espectrogramas, bancos de filtros, decomposição com PCA (*Principal Component Analysis*) e também técnicas que envolvem, por exemplo, o cálculo de parâmetros estatísticos ou de coeficientes autorregressivos (OPPENHEIM *et al.*, 1999) (DUDA *et al.*, 2001), (BISHOP, 2007), (CARVALHO, 2016).

Nos últimos anos, esse problema tem sido abordado com o auxílio de técnicas de aumento de dados (*data augmentation*), que aumentam artificialmente a quantidade de amostras: a partir de dados reais obtidos em sessões de experimentação, essas técnicas criam amostras sintéticas que apresentam características e comportamento similares aos dos dados originais, sendo possível incorporá-las ao conjunto de treinamento de classificadores da BCI e, assim, potencialmente melhorar os seus desempenhos. Diferentes técnicas têm sido empregadas para esta finalidade, desde as mais simples, como, por exemplo, inserção de ruídos e perturbações aleatórias, até o uso de modelos complexos de aprendizado profundo que se mostraram efetivos em processamento de imagens, como os *autoencoders* variacionais (VAE, do inglês *Variational Autoencoder*) (KINGMA; WELLING, 2014) e as redes adversárias generativas (GAN, do inglês *Generative Adversarial Network*) (GOOD-

FELLOW *et al.*, 2014).

Neste trabalho investigamos o uso de modelos generativos de aprendizado profundo para síntese de EEGs artificiais e como a incorporação desses sinais artificiais impacta o treinamento e o desempenho de classificadores em BCIs baseadas no paradigma SSVEP (do inglês, *Steady-State Visual Evoked Potential*), paradigma no qual cada ação/intenção é associada a um estímulo visual que oscila em uma frequência pré-determinada e que, ao ser observado pelo usuário, induz a produção de respostas cerebrais que oscilam em frequências múltiplas e sub-múltiplas do estímulo visual observado.

Embora existam muitos trabalhos sobre síntese de sinais artificiais para BCIs com técnicas de *data augmentation* (LASHGARI *et al.*, 2020), esse número é reduzido para BCIs baseadas no paradigma SSVEP, se resumindo aos trabalhos (AZNAN *et al.*, 2019), (AZNAN *et al.*, 2020) e (KWON; IM, 2022), em que os modelos generativos investigados e propostos são treinados para sintetizar EEGs no domínio do tempo. No entanto, esses trabalhos não abordam a síntese de EEGs no domínio da frequência, que parece ser o domínio mais adequado em BCIs-SSVEP devido à forma com que a atividade cerebral de interesse é induzida. Por esse motivo, investigamos a qualidade e o impacto dos EEGs sintetizados não apenas no domínio do tempo, mas também no domínio da frequência, treinando modelos generativos capazes de produzir diretamente os espectros dos EEGs.

Com este trabalho buscamos contribuir para o desenvolvimento de BCIs com maior desempenho e agregar ao estudo sobre *data augmentation* em BCIs com paradigma SSVEP, em especial quando a geração de sinais sintéticos se dá no domínio da frequência.

1.1 Organização do Trabalho

Este trabalho está estruturado em sete capítulos. O Capítulo 2 aborda os fundamentos de interfaces cérebro-computador, incluindo o funcionamento geral de uma BCI, os tipos de técnicas de aquisição de sinais cerebrais, os diferentes paradigmas e as características de cada etapa de processamento de sinais. No Capítulo 3 apresentamos os modelos generativos utilizados no trabalho, a saber: VAE, DC-GAN, AC-GAN, WGAN e AAE. No Capítulo 4 realizamos uma revisão da literatura sobre técnicas de *data augmentation* aplicadas a sistemas BCI, destacando os trabalhos (AZNAN *et al.*, 2019), (AZNAN *et al.*, 2020) e (KWON; IM, 2022) que exploram a síntese e o uso de EEGs artificiais em BCIs-SSVEP. Finalizados os capítulos de caráter mais teórico, o Capítulo 5 apresenta a base de dados, as estruturas dos modelos de classificação da BCI e dos modelos generativos para síntese dos EEGs, junto com a sequência de experimentos realizados. Os resultados obtidos e as discussões são reportados no Capítulo 6. Por fim, o Capítulo 7 traz as conclusões gerais do trabalho, bem como as perspectivas futuras.

2 Fundamentos de interfaces cérebro-computador

Neste capítulo, são apresentados os conceitos fundamentais que formam a base teórica para a compreensão e o desenvolvimento de interfaces cérebro-computador. A primeira seção aborda a definição e o funcionamento geral das BCIs, apresentando suas principais etapas e a função de cada uma delas. As duas seções seguintes trazem, respectivamente, os métodos de aquisição de sinais cerebrais e os paradigmas de sistemas BCI mais encontrados na literatura, destacando a eletroencefalografia e o paradigma SSVEP, ambos utilizados neste trabalho. Por fim, a quarta e última seção apresenta os conceitos teóricos por trás das técnicas computacionais empregadas no módulo de processamento de sinais, responsável por extrair as informações de interesse da atividade cerebral monitorada.

2.1 Funcionamento geral das BCIs

Uma interface cérebro-computador é um sistema de comunicação em que a intenção de um indivíduo é transmitida para a máquina por meio do monitoramento de sua atividade cerebral seguido do processamento dos sinais registrados, e não a partir do uso das vias convencionais de saída do sistema nervoso, como os nervos periféricos e músculos (WOLPAW, 2002). O funcionamento de uma BCI envolve: *(i)* codificar a intenção do usuário em sua atividade cerebral por meio de algum fenômeno neurofisiológico, *(ii)* observar essa atividade cerebral registrando os sinais de interesse e *(iii)* aplicar uma sequência de processamentos aos sinais capturados de modo que o resultado seja um comando ou saída que reflita a intenção inicial do usuário.

A Figura 2.1 apresenta as etapas típicas de uma BCI. A operação inicia com a captura da atividade cerebral produzida pelo usuário, que pode ser gerada de forma endógena ou exógena, como consequência de um estímulo externo. Os sinais monitorados passam por etapas de pré-processamento, ainda no domínio analógico, que visam melhorar a qualidade desses sinais e, em seguida, são digitalizados e submetidos ao módulo de processamento de sinais. Neste módulo ocorrem, além da aplicação de uma ou mais técnicas de pré-processamento digital, três grandes etapas (CARVALHO, 2016; VARGAS, 2021): a extração de características, a seleção de atributos e a classificação. Na etapa de extração de características, são aplicadas técnicas que visam representar os sinais amostrados da melhor forma possível, extraindo as informações relevantes do dado. Na etapa de seleção

de atributos, são empregadas técnicas que ordenam e selecionam o conjunto de atributos mais discriminativos com o objetivo de facilitar o treinamento do modelo de tomada de decisão da etapa de classificação, que é responsável por gerar a saída correspondente à intenção do usuário.

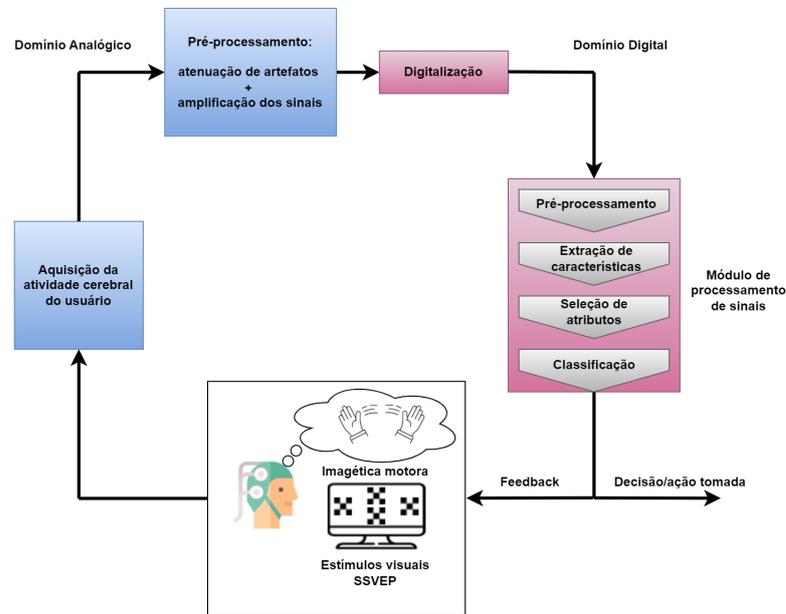


Figura 2.1 – Diagrama geral de uma interface cérebro-computador.

A escolha da técnica de aquisição dos sinais cerebrais e a escolha do paradigma empregado para definir a interação do usuário com a interface são etapas cruciais no projeto desse sistema. Tais escolhas devem considerar a aplicação da BCI, as características do usuário e os recursos técnicos e computacionais disponíveis. Além disso, o paradigma escolhido influencia significativamente no módulo de processamento, pois os sinais gerados por cada paradigma podem ser tratados de maneira mais assertiva explorando técnicas diferentes.

As próximas seções apresentam os principais métodos de aquisição de sinais cerebrais, os tipos de paradigmas de BCI empregados e os componentes típicos do módulo de processamento, destacando as técnicas utilizadas neste trabalho.

2.2 Métodos de aquisição dos sinais cerebrais

Os métodos de aquisição de sinais cerebrais são técnicas de monitoramento que utilizam sensores para registrar a atividade cerebral de um indivíduo. No contexto de BCIs, é desejado que o método de aquisição apresente (i) boa relação sinal-ruído, capturando sinais de melhor qualidade para facilitar a etapa de processamento, (ii) boa resolução temporal, permitindo uma resposta rápida do sistema, (iii) boa resolução espacial, para que o monitoramento seja feito simultaneamente em diferentes regiões do

cérebro, (iv) alta flexibilidade, para permitir diferentes experimentações e disposições dos sensores e (v) baixo custo, para viabilizar o desenvolvimento e o acesso à BCI (CARVALHO, 2016).

As opções existentes para a aquisição dos sinais se dividem em métodos invasivos e não-invasivos, e oferecem compromissos distintos entre os requisitos apontados anteriormente. Os métodos invasivos são os que exigem intervenção cirúrgica para implantação dos sensores diretamente no córtex cerebral e, como consequência, apresentam uma ótima relação sinal-ruído, alta resolução espacial e alta resolução temporal (NICOLAS-ALONSO; GOMEZ-GIL, 2012; RAMADAN; VASILAKOS, 2016). Embora possuam características desejáveis, a necessidade de intervenção cirúrgica faz essa classe de técnicas esbarrar em considerações éticas e financeiras que limitam drasticamente o uso (CARVALHO, 2016). Os métodos invasivos mais comuns são:

- **Eletrocorticograma (ECoG):** esta técnica mede a atividade elétrica diretamente da superfície do córtex cerebral, onde os eletrodos estão posicionados. Por se tratar de uma técnica invasiva, o paciente tem um pedaço muito pequeno de seu crânio temporariamente removido para a implantação dos eletrodos (NICOLAS-ALONSO; GOMEZ-GIL, 2012).
- **Gravação intracortical:** diferente do ECoG, onde os eletrodos estão posicionados na superfície do córtex, na gravação intracortical, microeletrodos são inseridos no interior da massa cinzenta do cérebro do paciente, o que permite registrar a atividade elétrica de um único neurônio (SUA, *single-unit activity*) ou de um conjunto de neurônios (MUA, *multi-unit activity*) (NICOLAS-ALONSO; GOMEZ-GIL, 2012).

Os métodos não-invasivos são os que não exigem intervenção cirúrgica para a colocação dos sensores, e, por esta razão, são mais empregados em BCIs (RAMADAN; VASILAKOS, 2016; CARVALHO, 2016), e possuem maior flexibilidade e custos mais reduzidos. Todavia, os métodos não-invasivos tendem a apresentar menor relação sinal-ruído e menor resolução espacial quando comparados aos métodos invasivos, e dependendo da técnica empregada, podem também apresentar menor resolução temporal. As técnicas não-invasivas mais comuns são:

- **Eletroencefalografia:** nesta técnica, a atividade elétrica do cérebro é monitorada por meio de eletrodos posicionados no escalpo do paciente e registrada em um conjunto de sinais denominados eletroencefalogramas (EEGs). As principais características do EEG são sua boa resolução temporal (na ordem de milissegundos), baixo custo e alta flexibilidade, o que o torna o método de aquisição mais empregado no

desenvolvimento de BCIs (NICOLAS-ALONSO; GOMEZ-GIL, 2012; RAMADAN; VASILAKOS, 2016; CARVALHO, 2016; AZNAN *et al.*, 2018; MACHADO *et al.*, 2010; VăRBU *et al.*, 2022; AGGARWAL; CHUGH, 2022).

- **Magnetoencefalografia (MEG):** neste método de aquisição as correntes elétricas decorrentes da atividade cerebral produzem campos magnéticos que são monitorados por magnetômetros muito sensíveis, mapeando a atividade magnética do cérebro em neuroimagens funcionais (CARVALHO, 2016). A MEG apresenta resolução espacial maior que a do EEG e aproximadamente a mesma resolução temporal. Contudo, o aparelho de medição é muito grande, o que impede sua portabilidade em aplicações de BCI que requerem deslocamentos do usuário (NICOLAS-ALONSO; GOMEZ-GIL, 2012).
- **Ressonância magnética funcional (fMRI):** nesta técnica a alteração do fluxo sanguíneo é detectada pela medição do nível de oxigênio do sangue com o uso de ressonância magnética. Assim como a MEG, a fMRI possui alta resolução espacial e registra a atividade cerebral em neuroimagens funcionais, mas diferente da técnica anterior, a fMRI apresenta baixa resolução temporal (na ordem de segundos), o que pode inviabilizar o seu uso dependendo da aplicação da BCI (NICOLAS-ALONSO; GOMEZ-GIL, 2012).
- **Espectroscopia de infravermelho próximo (NIRS):** neste método, a atividade cerebral é capturada através do monitoramento do fluxo de oxigenação cerebral e do metabolismo de certas regiões do cérebro, ambos com o uso de feixes de luz no espectro infra-vermelho (700 a 1000 nm). Assim como a fMRI, a NIRS possui baixa resolução temporal (na ordem de segundos) (COYLE *et al.*, 2004; NICOLAS-ALONSO; GOMEZ-GIL, 2012).

Devido ao grande uso da eletroencefalografia, vários estudos foram realizados com o objetivo de buscar a melhor forma de posicionar os eletrodos no escalpo e padronizar as coletas de EEG. Em 1958, foi proposto o padrão 10-20 (JASPER, 1958), permitindo o posicionamento de até 21 eletrodos sobre o escalpo para registro da atividade elétrica do paciente. Com o avanço de sistemas de EEG multicanais, de técnicas de mapeamento da atividade cerebral e do crescente uso de EEG no contexto de BCIs, foi proposto em 1999 o padrão 10-10, que é uma extensão do padrão anterior que possibilita o posicionamento de até 81 eletrodos sobre o escalpo (KLEM *et al.*, 1999), como mostrado na Figura 2.2.

Neste trabalho, utilizamos bases de dados com EEGs capturados a partir de 16 eletrodos dispostos segundo o padrão 10-10, alocados nas regiões dos lobos occipital (O1, O2 e Oz), parietal (POz, PO4, PO3, PO8, PO7, Pz, P1 e P2), sulco central (Cz, C1 e

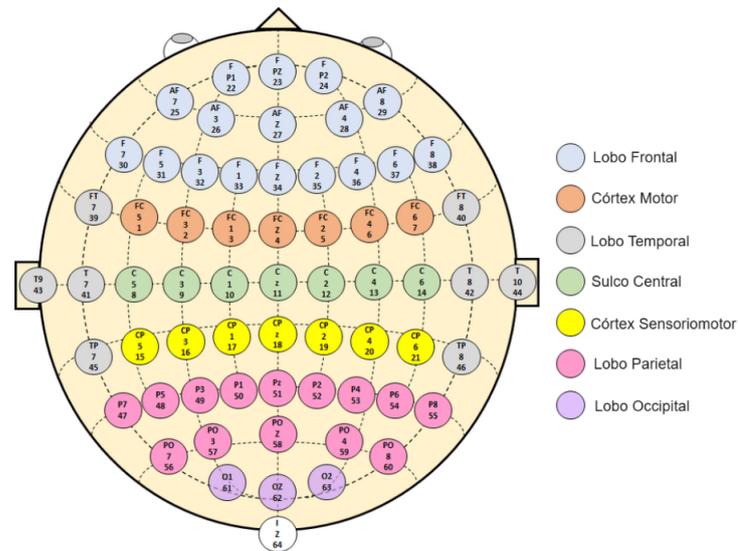


Figura 2.2 – Disposição dos eletrodos no escalpo segundo o padrão internacional 10-10. Adaptado de (ASANZA *et al.*, 2022).

C2) e nos córtex motor (CPz) e sensoriomotor (FCz). A medição dos potenciais elétricos seguiu o modo referencial, de forma que o potencial elétrico associado a cada eletrodo foi dado pela diferença entre o potencial medido neste eletrodo e o potencial medido no eletrodo de referência posicionado nos processos mastoides, que apresenta pouca ou nenhuma atividade elétrica (CARVALHO, 2016).

2.3 Paradigmas de sistemas BCI

O paradigma de uma BCI define como ocorre a interação do usuário com a aplicação e qual o processo neurofisiológico responsável por produzir as atividades cerebrais de interesse, que carregam as intenções do usuário durante o uso da BCI. Os diferentes paradigmas podem ser classificados com base na presença ou ausência de estimulação externa (CARVALHO, 2016).

Nos paradigmas sem estimulação externa, o principal exemplo é o baseado na imagética motora, em que o usuário faz exercícios de imaginação de movimentos, gerando padrões no córtex motor semelhantes aos padrões gerados caso ele realmente realizasse os movimentos imaginados (PFURTSCHELLER; NEUPER, 1997). Outra possibilidade, um pouco menos comum, é a imagética não-motora, em que a atividade cerebral de interesse é gerada por meio de alguma tarefa mental como, por exemplo, imaginar uma figura geométrica ou uma palavra (CURRAN *et al.*, 2004).

Nos paradigmas com estimulação externa, as BCIs são baseadas nos potenciais relacionados a eventos (ERP, do inglês *Event-Related Potential*) e exploram uma característica do cérebro humano denominada atenção seletiva, que permite ao indivíduo se

concentrar nos estímulos externos de interesse e, assim, a atividade cerebral induzida por eles ser mais relevante. As abordagens mais comuns são a P300, baseada em um potencial endógeno denominado P300, que aparece aproximadamente 300 ms após o acontecimento de um evento visual desejado pelo usuário (DONCHIN; SMITH, 1970), e o Potencial Evocado em Regime Estacionário (SSEP, do inglês *Steady-State Evoked Potential*), baseado em um potencial elétrico exógeno que surge nos sinais cerebrais do usuário em decorrência de um estímulo externo oscilando periodicamente com uma frequência pré-definida, denominada de frequência de evocação, e que, em regime, o sinal elétrico induzido passa a apresentar componentes harmônicas e sub-harmônicas da frequência evocada em seu conteúdo espectral (CARVALHO, 2016; VARGAS, 2021).

O potencial induzido pelo paradigma SSEP pode ter sua origem a partir de estímulos auditivos, somatossensoriais e visuais. Dentre estes, a estimulação visual é a mais empregada em BCIs, sendo as interfaces resultantes chamadas de BCIs baseadas no paradigma SSVEP (do inglês *Steady-State Visual Evoked Potential*). A Figura 2.3 mostra um exemplo do espectro de um EEG amostrado do eletrodo Oz de um indivíduo submetido a um estímulo visual oscilando com uma frequência de evocação de 12 Hz. Como pode ser observado, o espectro do sinal amostrado possui pico elevado na frequência de estimulação, além de apresentar picos também nas frequências de 6 Hz e de 24 Hz que são, respectivamente, frequências sub-múltipla e múltipla da frequência de evocação investigada.

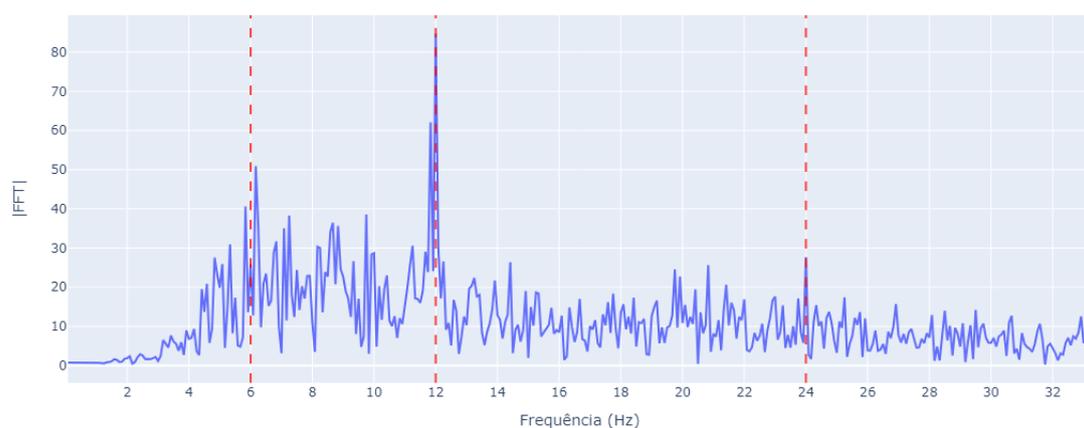


Figura 2.3 – Espectro de magnitude de um EEG no contexto de um sistema BCI-SSVEP, no qual o paciente foi exposto a um estímulo visual com frequência de 12 Hz.

Neste trabalho utilizamos o paradigma SSVEP para definir a forma de interação dos usuários com a BCI e o processo neurofisiológico responsável por produzir as atividades cerebrais de interesse. Para estimulação visual, escolhemos utilizar um monitor

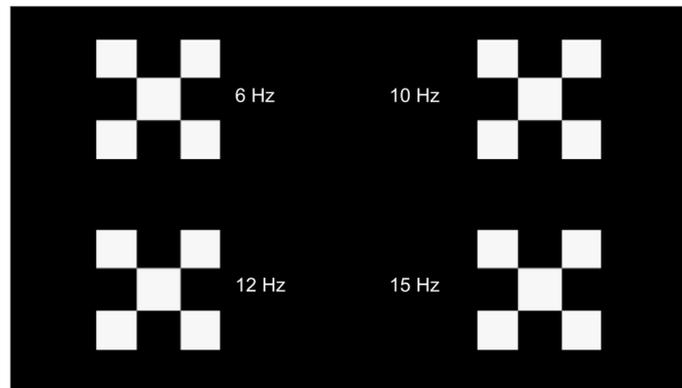


Figura 2.4 – Figura esquemática representando o monitor utilizado na etapa experimental contendo os padrões xadrez que oscilavam com as frequências de evocação 6 Hz, 10 Hz, 12 Hz e 15 Hz.

apresentando simultaneamente quatro padrões xadrez oscilando em frequências de estimulação iguais a 6 Hz, 10 Hz, 12 Hz e 15 Hz, semelhante ao apresentado na Figura 2.4, e associamos a cada frequência de evocação um comando a ser gerado pela BCI. A Figura 2.5 apresenta exemplos de sinais de EEG referentes ao eletrodo Oz amostrados em um dos pacientes durante a etapa experimental. Podemos notar picos claros de energia em todas as frequências de evocação escolhidas.

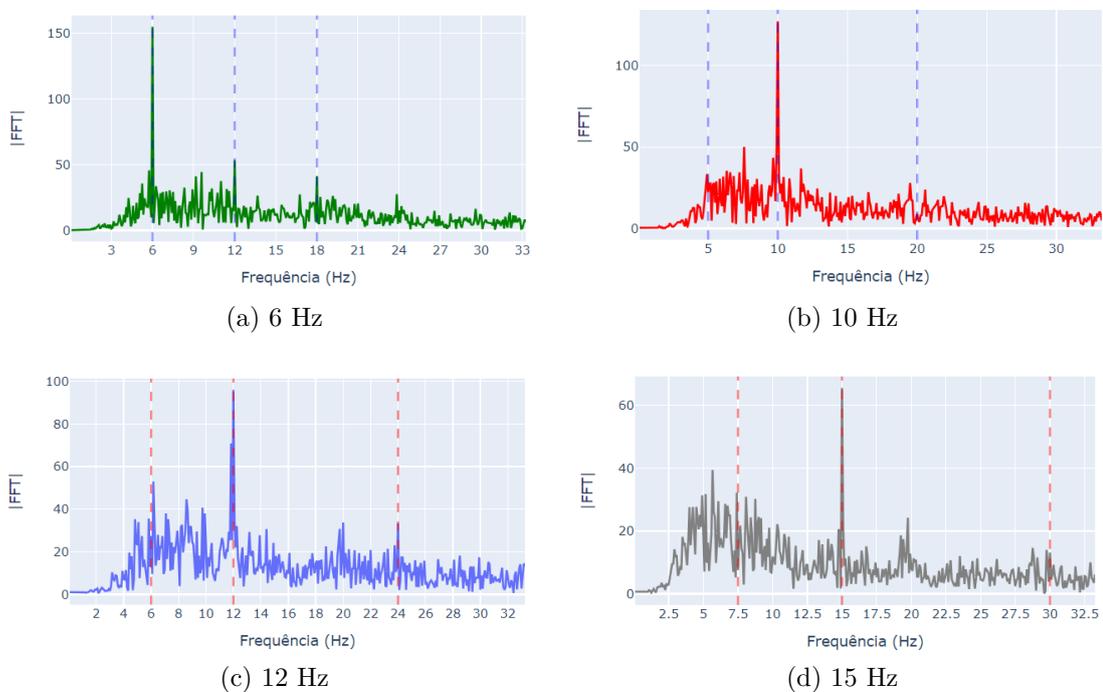


Figura 2.5 – Exemplos de espectros de EEGs referentes a um paciente exposto a estímulos visuais com frequências de oscilação iguais a: (a) 6 Hz, (b) 10 Hz, (c) 12 Hz e (d) 15 Hz.

2.4 Módulo de processamento de sinais

Definidos o SSVEP como paradigma e a eletroencefalografia como método de aquisição dos sinais cerebrais, chegamos ao módulo de processamento de sinais, que pode ser considerado como a principal estrutura da BCI, por ser responsável por gerar os comandos correspondentes às ações desejadas pelo usuário após a aplicação de sucessivas transformações sobre os EEGs monitorados durante o uso da BCI.

2.4.1 Pré-processamento

A etapa de pré-processamento busca melhorar a qualidade dos sinais amostrados, a fim de facilitar as tarefas de extração e seleção de atributos e, sobretudo, a de classificação, aprimorando o desempenho da BCI. A necessidade de uma etapa de pré-processamento é consequência das baixas amplitudes dos sinais amostrados e, principalmente, da presença de artefatos fisiológicos e instrumentais que degradam a qualidade dos sinais cerebrais capturados.

Os artefatos fisiológicos são gerados involuntariamente pelo usuário em virtude do próprio funcionamento de seu sistema fisiológico. Batimento cardíaco, respiração, movimento de músculos, pensamentos, humor, entre outros fatores, geram atividades cerebrais que naturalmente se somam à atividade cerebral induzida pelos estímulos visuais durante a interação do usuário com a interface (MARKAND, 1996). Concomitantemente, existem os artefatos instrumentais que são de origem não fisiológica e estão associados à interferência eletromagnética de equipamentos elétricos e eletrônicos do ambiente, da rede elétrica e do próprio aparelho de monitoramento (CARVALHO, 2016; VARGAS, 2021).

O pré-processamento inicia no domínio analógico com a aplicação de um filtro Notch com frequência central variando entre 50 Hz e 60 Hz visando anular os artefatos associados à interferência eletromagnética da rede elétrica. Em seguida, um filtro passa-faixa de Butterworth com faixa de frequências no intervalo de 5 a 60 Hz é aplicado, pois as BCIs que fazem uso de monitores para projetar os estímulos visuais de interesse do paradigma SSVEP têm a frequência de evocação máxima limitada à taxa de atualização dos monitores (usualmente 60 Hz). Por fim, os sinais de EEG filtrados analogicamente passam por um amplificador e por um conversor analógico-digital, discretizando os sinais amostrados e possibilitando o processamento digital (CARVALHO, 2016).

Embora seja possível a aplicação de inúmeras técnicas de filtragem no domínio digital para melhorar a qualidade dos sinais amostrados, as escolhas do paradigma SSVEP e do método de medição referencial dos EEGs diminuem a necessidade por pré-processamentos custosos computacionalmente, pois os sinais obtidos são relativamente robustos aos artefatos, já que as frequências de evocação empregadas são previamente

conhecidas e definidas logo no início do projeto da BCI. Além disso, foi constatado em (BERTRAND *et al.*, 1985) que, no modo de medição referencial dos potenciais elétricos dos eletrodos, os artefatos tendem a aparecer simultaneamente e com aproximadamente a mesma intensidade em todos os eletrodos posicionados no escalpo. Assim, é possível aplicar o filtro CAR (do inglês, *Common Average Reference*) para atenuar a influência dos artefatos nos sinais amostrados de cada eletrodo.

A Equação (2.1) apresenta a formulação matemática do filtro CAR. Nessa técnica, para cada instante de tempo n , calcula-se a média do potencial elétrico $V^{AVG}(n)$ em função de todos os N eletrodos posicionados no escalpo. Então, o potencial de saída do filtro CAR no instante n para o i -ésimo eletrodo, $V_i^{CAR}(n)$, é dado pelo potencial de entrada $V_i^{ER}(n)$ desse eletrodo no mesmo instante de tempo subtraído do potencial médio $V^{AVG}(n)$:

$$V_i^{CAR}(n) = V_i^{ER}(n) - V^{AVG}(n) = V_i^{ER}(n) - \frac{1}{N} \sum_{j=1}^N V_j^{ER}(n), \quad (2.1)$$

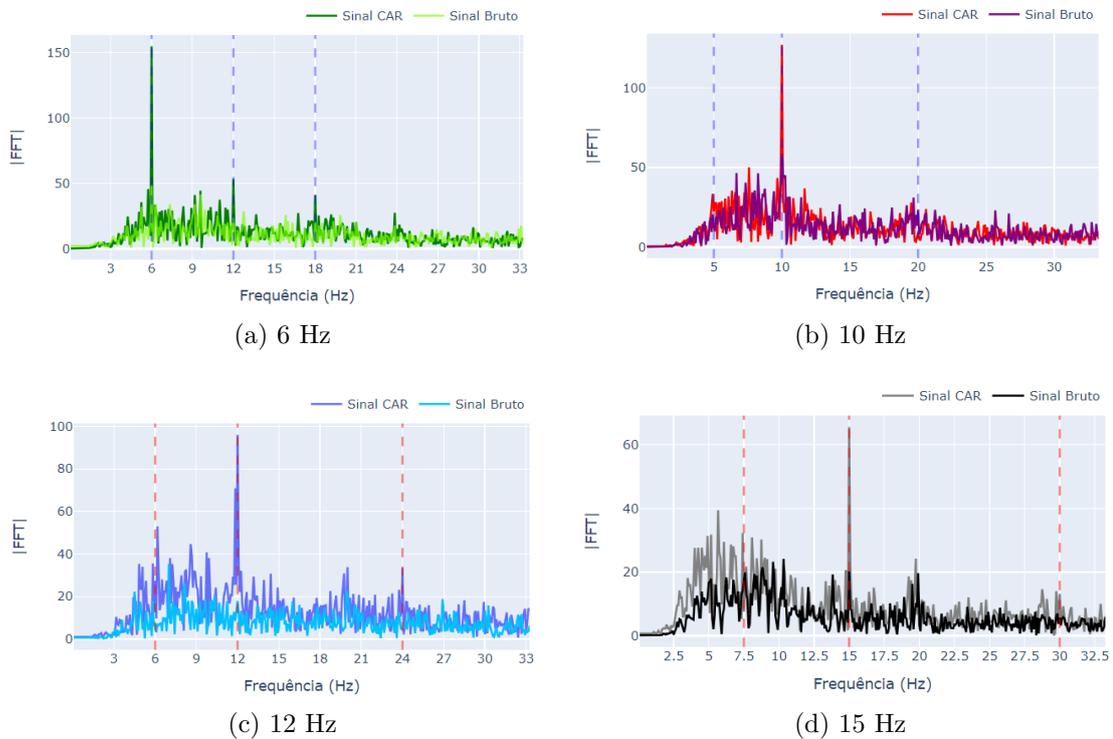


Figura 2.6 – Exemplos de espectros antes e após a aplicação do filtro CAR em sinais de EEG-SSVEP amostrados durante a etapa de experimentação após submeter um paciente aos estímulos visuais com frequências de oscilação iguais a: (a) 6 Hz, (b) 10 Hz, (c) 12 Hz e (d) 15 Hz.

Apesar da simplicidade, o filtro CAR é bastante efetivo no pré-processamento dos EEGs. A título de ilustração, exibimos na Figura 2.6 os espectros de EEGs antes e

após a aplicação do filtro CAR para quatro frequências de excitação. No caso, os espectros obtidos depois da ação do filtro CAR são exatamente os mesmos que foram mostrados na Figura 2.5. Podemos observar que o conteúdo espectral associado a artefatos foi reduzido consideravelmente, pois o filtro CAR deixou as componentes espectrais atreladas às frequências evocadas pelos estímulos visuais mais nítidas e com maior destaque.

2.4.2 Extração de características e seleção de atributos

Na etapa de extração de características, buscamos aplicar técnicas de transformação sobre os sinais amostrados com o objetivo de que a nova representação dos sinais seja eficiente, contendo as informações mais relevantes e discriminativas, e compacta, isto é, que apresente dimensão reduzida para que a necessidade por volumosos conjuntos de dados seja amenizada e, conseqüentemente, o treinamento do classificador da BCI seja efetivo. No contexto de BCIs baseadas no paradigma SSVEP, nas quais os estímulos visuais se dão em frequências específicas e bem definidas, a representação dos EEGs no domínio da frequência se mostra bastante adequada. Neste sentido podem ser aplicadas diversas técnicas de transformação como, por exemplo, transformada discreta de Fourier (DFT, do inglês *Discrete Fourier Transform*), transformada Wavelet, espectrogramas, bancos de filtros, entre outras técnicas que levam os sinais do domínio do tempo para o domínio da frequência (OPPENHEIM *et al.*, 1999) (DUDA *et al.*, 2001), (BISHOP, 2007), (CARVALHO, 2016).

Neste trabalho, empregamos a DFT, com auxílio do algoritmo FFT (*Fast Fourier Transform*), para mapear os EEGs para o domínio da frequência. Seja $x[n]$ um EEG registrado durante T segundos e com uma taxa de amostragem de f_s . O parâmetro N denota a quantidade de amostras ou a duração de $x[n]$. Então, a DFT de N pontos de $x[n]$ é dada por:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j \frac{2\pi}{N} kn}, k = 0, \dots, N-1 \quad (2.2)$$

A k -ésima componente da DFT, $X[k]$, está associada a uma frequência analógica do EEG original por meio da seguinte relação:

$$f_k = k \frac{f_s}{N} \quad (2.3)$$

Finalizada a etapa de extração de características podemos aplicar, caso necessário, técnicas de seleção de atributos para tentar facilitar o treinamento do classificador e, também, aprimorar o seu desempenho ao reter apenas as variáveis mais relevantes (CARVALHO, 2016; BISHOP, 1995; GUYON; ELISSEEFF, 2003).

As técnicas de seleção de atributos são classificadas como filtros ou *wrappers* (GUYON; ELISSEEFF, 2003). Os filtros fazem a ordenação dos atributos por relevância a partir do cálculo de uma medida pré-escolhida. Então, apenas o subconjunto dos atributos mais relevantes é selecionado e passado como entrada para o classificador da BCI. Algumas métricas comumente exploradas em filtros são a correlação de Pearson (BISHOP, 2007) e a informação mútua (COVER; THOMAS, 2006). Note que a seleção feita por um filtro independe do classificador que será utilizado pela BCI.

Por outro lado, nos *wrappers*, a escolha pelo subconjunto de atributos mais relevantes é feita com o auxílio do classificador. Nestas técnicas, vários subconjuntos de atributos são definidos e têm sua qualidade determinada a partir do treinamento do classificador e, por fim, da verificação de seu desempenho em um conjunto de validação. Ao final, o subconjunto que levou ao melhor desempenho do classificador é o escolhido. Exemplos de *wrappers* são as técnicas *forward selection* e *backward elimination* (GUYON; ELISSEEFF, 2003).

Neste trabalho não utilizamos técnicas de seleção de atributos para reduzir a dimensão dos dados de entrada do classificador. Assim, as entradas dos classificadores da BCI foram os próprios EEGs (i.e., os sinais no domínio do tempo após a aplicação do filtro CAR) ou, então, as magnitudes dos espectros dos EEGs (sinais no domínio da frequência).

2.4.3 Classificação

Na etapa de classificação, os atributos dos dados de entrada são combinados e mapeados em um conjunto de variáveis de saída que representam todas as ações/comandos possíveis da tarefa para a qual a BCI foi desenvolvida. Na área de aprendizado de máquina, existe uma extensa literatura voltada ao problema de classificação de padrões (DUDA *et al.*, 2001) (BISHOP, 2007), com o desenvolvimento de técnicas que vão desde as mais simples, como um classificador linear, até as mais complexas, como os modelos baseados em redes neurais profundas.

Neste trabalho, implementamos classificadores da BCI com base em máquinas de vetores-suporte (SVM, do inglês *Support Vector Machines*), em suas versões linear e não-linear, e, também, em redes neurais convolucionais 1-D. A escolha desses modelos foi motivada (*i*) pelo interesse em avaliar o impacto do uso de dados artificiais de EEG no treinamento de modelos mais tradicionais, como as SVMs, e de modelos de redes mais modernas, como as convolucionais e (*ii*) pela necessidade de classificadores que operassem bem tanto no domínio do tempo, processando os EEGs brutos, quanto no domínio da frequência, processando os atributos espectrais desses sinais.

Máquinas de vetores-suporte (SVM)

As máquinas de vetores-suporte foram inicialmente desenvolvidas para tratar problemas de classificação binária em que os dados são linearmente separáveis, ou seja, as amostras das duas classes do problema podem ser separadas no espaço de atributos por um hiperplano. Esta abordagem tem como objetivo obter a fronteira de decisão linear de máxima margem e, portanto, associada ao melhor desempenho possível do classificador (GERON, 2022).

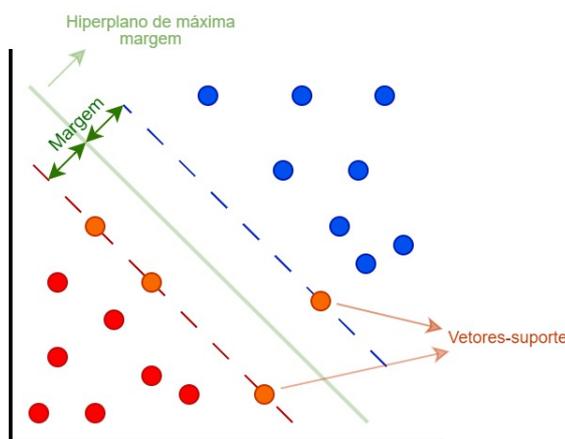


Figura 2.7 – Hiperplano de máxima margem e seus vetores-suporte obtidos com SVM.

A Figura 2.7 mostra o que seria um hiperplano ótimo para o caso em que os dados possuem apenas dois atributos. Como podemos observar, o hiperplano ótimo, também chamado de hiperplano de máxima margem, é aquele que está situado o mais distante possível das amostras mais próximas de ambas as classes. Ele é dado pela combinação linear das amostras mais próximas à fronteira (os vetores-suporte), as quais são obtidas automaticamente na solução do problema de maximização da margem.

Entretanto, como existem problemas de classificação não-lineares, ou seja, problemas em que os dados não são linearmente separáveis, foram propostas extensões da formulação original da SVM para possibilitar a aplicação da técnica em problemas com essa propriedade. A primeira extensão foi a SVM de margem suave, permitindo que algumas amostras violassem a margem da fronteira de decisão que ainda é linear (CORTES; VAPNIK, 1995).

Já a segunda extensão da SVM produz fronteiras de decisão não-lineares utilizando a seguinte estratégia: (1) mapear o problema do espaço original (de atributos) para um espaço de dimensão muito mais elevada de modo que neste novo espaço os dados se tornem linearmente separáveis, (2) resolver o problema linear no novo espaço, e (3) mapear a solução para o espaço original, conforme mostra o diagrama da Figura 2.8.

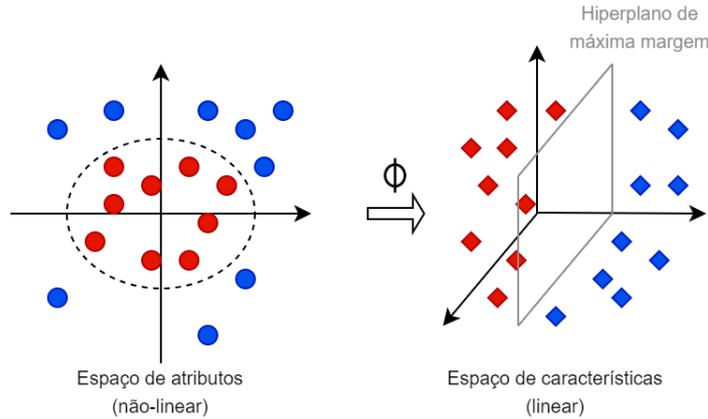


Figura 2.8 – Diagrama representando a ideia da extensão da SVM para tratar problemas não-lineares no espaço de atributos, mas que se tornam lineares no espaço de características.

A dificuldade dessa abordagem está em determinar o mapeamento $\phi(\cdot)$ do espaço de atributos para o espaço de características. Felizmente, na formulação matemática do problema, o mapeamento $\phi(\cdot)$ não é diretamente exigido, mas somente produtos escalares no novo espaço de características. Então, lançando mão da teoria de funções *kernel*, é possível calcular estes produtos escalares no espaço de características sem que os dados sejam realmente mapeados para lá. Na verdade, através do truque do *kernel*, a aplicação de uma função *kernel* $k(\mathbf{u}, \mathbf{v})$ sobre duas amostras no espaço original já fornece o valor do produto escalar de suas versões mapeadas no espaço de características Equação (2.4). Exemplos de funções *kernel* são as funções polinomiais e funções de base radial (RBF, do inglês *Radial Basis Function*) apresentadas, respectivamente, nas Equações (2.5) e (2.6), em que γ , d e as próprias funções *kernel* são hiperparâmetros a serem escolhidos com o auxílio de técnicas de validação cruzada (GERON, 2022).

$$k(\mathbf{u}, \mathbf{v}) = \phi(\mathbf{u})^T \phi(\mathbf{v}) \quad (2.4)$$

$$\text{Polinomial} : k(\mathbf{u}, \mathbf{v}) = (\gamma \mathbf{u}^T \mathbf{v} + r)^d \quad (2.5)$$

$$\text{RBF} : k(\mathbf{u}, \mathbf{v}) = \exp(-\gamma \|\mathbf{u} - \mathbf{v}\|^2). \quad (2.6)$$

Historicamente as máquinas de vetores-suporte são empregadas como classificadores em BCIs nas mais diversas aplicações. Em (TODEREA; CHIUCHISAN, 2017), por exemplo, a SVM é utilizada como classificador em uma BCI baseada em imagética motora para auxiliar no processo de reabilitação de movimento de pacientes. Em (J LI Y, 2007) é proposta uma técnica de SVM semi-supervisionada com o objetivo de reduzir o tempo necessário para obter os modelos de classificação e permitir o uso de forma online.

Em (KAPER *et al.*, 2004), a SVM é utilizada em uma BCI baseada no paradigma P300. Por fim, em (CARVALHO *et al.*, 2015), é feito um estudo comparativo de diversas técnicas de extração, seleção e classificação em BCIs-SSVEP, onde a SVM também é empregada nos diversos ensaios.

Redes Neurais Convolucionais 1D (CNN-1D)

Os classificadores baseados em redes neurais convolucionais 1D, CNN-1D, da Figura 2.9, são formados por uma camada inicial, que usualmente é o próprio dado de entrada, uma ou mais camadas intermediárias compostas por unidades de processamento denominadas filtros ou *kernels* convolutivos, e por uma camada de saída de classificação responsável por mapear a saída da última camada intermediária a uma das classes do problema. Assim como na extensão da SVM para o tratamento de problemas de classificação não-lineares, um dos objetivos do uso das camadas intermediárias nos classificadores baseados em redes neurais convolucionais é mapear os dados de entrada, inicialmente dispostos em um espaço de atributos onde o problema é não-linear, em um espaço onde o problema seja mais linear (saída da última camada intermediária), para que, por fim, a camada de saída de classificação consiga definir a classe correspondente ao dado de entrada a partir do problema mais linear, que tende a ser mais simples e assertivo.

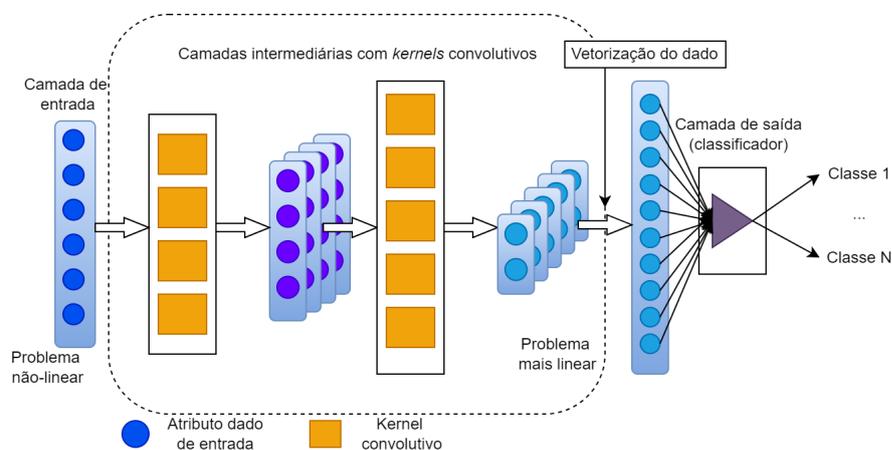


Figura 2.9 – Diagrama representando um classificador baseado em uma rede neural convolucional 1D.

O funcionamento de cada um dos *kernels* convolutivos do classificador CNN-1D da Figura 2.9 é apresentado com detalhes na Figura 2.10. Essa forma de processamento é muito interessante quando os atributos dos dados de entrada apresentam uma estrutura no tempo, como as séries temporais, ou no espaço, como as imagens, pois permite que os atributos vizinhos sejam processados localmente pelos filtros. Considerando um único *kernel* da primeira camada intermediária da rede, o início da operação ocorre com o *kernel* sendo posicionado no começo do dado de entrada e, neste instante, a saída do *kernel* é

dada pelo produto escalar entre seus parâmetros (no caso do exemplo, os três coeficientes w_1 , w_2 e w_3) e os atributos $\{x_1, x_2, x_3\}$ do dado sobre os quais o *kernel* está posicionado. Em seguida, o *kernel* se desloca uma posição para a direita e sua saída é dada pelo produto escalar entre seus parâmetros $\{w_1, w_2, w_3\}$ e um novo subconjunto de atributos do dado de entrada, $\{x_2, x_3, x_4\}$. Esse processo ocorre até que o *kernel* alcance o final do dado de entrada, processando todos os seus atributos. Como resultado final, a saída completa do *kernel* é um dado também estruturado comumente chamado de *feature map*, que pode servir de entrada para os filtros convolucionais seguintes.

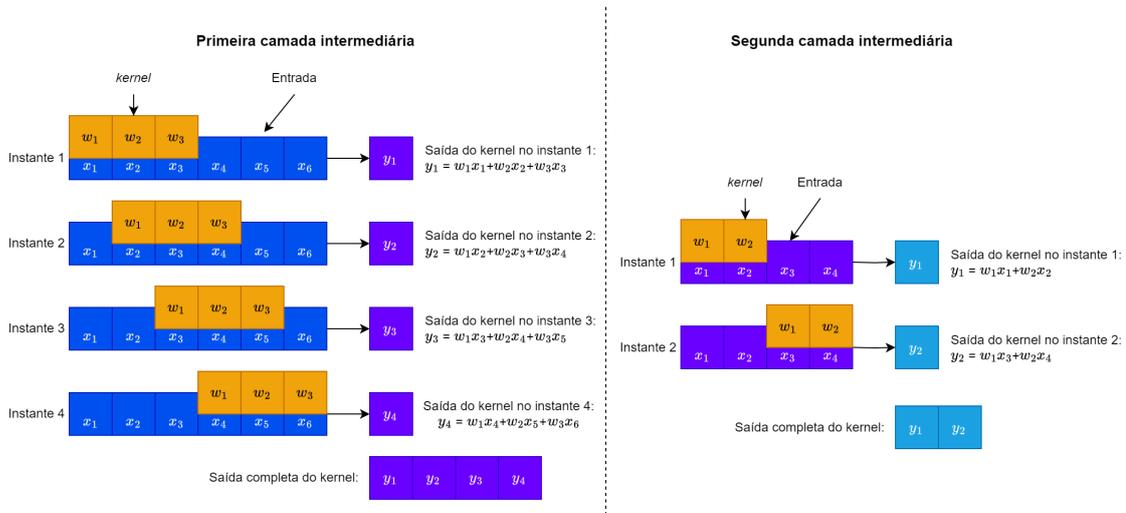


Figura 2.10 – Operação de um kernel aplicando convolução numa amostra de entrada.

A partir das Figuras 2.9 e 2.10 é possível notar que existem alguns hiperparâmetros importantes na definição da estrutura de um modelo de classificação baseado em CNNs. A seguir é apresentada a lista com os principais hiperparâmetros de uma camada convolucional:

- **Número de filtros:** define a quantidade de filtros ou *kernels* que serão utilizados na camada convolucional para processamento de seu dado de entrada. Cada *kernel* produz um dado de saída intermediário, que é justamente um *feature map* ou, simplesmente, um canal. Na rede da Figura 2.9, por exemplo, a saída da primeira camada intermediária é um dado estruturado com 4 canais, e a saída da segunda camada intermediária possui 5 canais. Logo, como cada canal é gerado por um filtro convolutivo, as duas camadas intermediárias possuem, respectivamente, 4 e 5 filtros convolucionais.
- **Tamanho dos filtros:** define a quantidade de parâmetros treináveis de cada *kernel*, que são utilizados na operação de convolução sobre o dado de entrada no cálculo do produto escalar descrito anteriormente. Para camadas convolucionais 1D, o tamanho dos filtros é dado por uma tupla (W, C) , onde W é o número de parâmetros

na primeira dimensão do *kernel* e C é o número de canais do dado de entrada, resultando em $W \times C$ parâmetros treináveis.

- **Stride:** define o passo de deslocamento do filtro, ou seja, de quantos em quantos atributos o filtro se desloca para realizar a operação de produto escalar. Na Figura 2.9, para que a entrada formada por 1 canal com 6 atributos seja processada e convertida em uma saída formada por 4 canais com 4 atributos cada, é necessário que cada *kernel* convolutivo possua tamanho (3, 1) e *stride* igual a 1. Já na segunda camada convolucional, para que sua entrada formada por 4 canais com 4 atributos produza uma saída composta por 5 *feature maps* com 2 atributos cada, é necessário que cada *kernel* convolutivo possua tamanho (2, 4) e *stride* igual a 2.
- **Padding:** define se as extremidades do dado de entrada serão preenchidas com atributos adicionais com valores iguais a zero. O objetivo do *padding* é permitir que os atributos das extremidades do dado também sejam processados pelos filtros convolutivos pois, dependendo da combinação de tamanho do *kernel* e *stride*, os atributos das extremidades podem não ser alcançados.

Usualmente a operação de convolução realizada pelos *kernels* é seguida por uma operação conhecida como *pooling*, que consiste em agregar localmente subconjuntos de atributos de saída, com o objetivo de tornar a saída da camada convolucional invariante a pequenas alterações no dado de entrada. No contexto de imagens, por exemplo, a operação de *pooling* permite que a mesma imagem levemente rotacionada resulte em uma mesma saída da camada convolucional. Os tipos de *pooling* mais conhecidos são o *Max-Pooling*, onde toma-se o maior valor entre os atributos (valores) que estão dentro da vizinhança do filtro de *pooling*, e o *Average-Pooling*, onde calcula-se o valor médio dos atributos que estão dentro da vizinhança do filtro de *pooling*. A Figura 2.11 exemplifica esses dois esquemas de *pooling* aplicados sobre um *feature map* contendo originalmente 10 atributos.

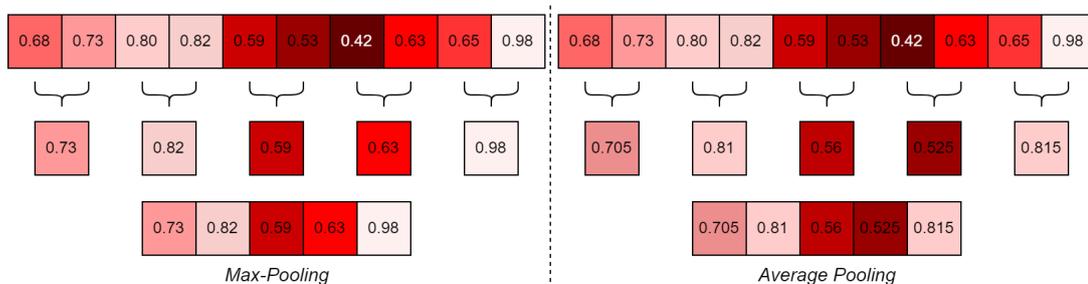


Figura 2.11 – Diagrama representando o processamento realizado pela operação de *pooling*.

Para que o problema inicial não-linear se torne mais linear ao longo dos sucessivos processamentos realizados pelas camadas intermediárias, é necessário empregar camadas ou transformações não-lineares sobre os dados produzidos ao longo da rede.

Para isso, são acopladas às saídas de cada filtro convolutivo **funções de ativação não-lineares** como, por exemplo, as funções de ativação *sigmoid*, *tanh*, *ReLU*, *Leaky ReLU*, *PReLU*, entre outras (GERON, 2022).

Por fim, após todo o processamento dos dados de entrada pelas camadas convolucionais, as saídas resultantes são vetorizadas e tipicamente passam por uma rede totalmente conectada (*fully-connected*) de saída para gerar a classificação. Geralmente, a camada de saída assume a forma de uma função *softmax*, a qual gera as probabilidades de a entrada pertencer a cada uma das classes do problema. Com isso, a decisão do classificador é pela classe associada à saída de maior probabilidade.

Redes Neurais Convolucionais Transpostas 1D

As camadas convolucionais apresentadas anteriormente são utilizadas não apenas nos classificadores CNN-1D das BCIs, mas também nas técnicas generativas que serão apresentadas no Capítulo 3. As camadas convolucionais são empregadas nas redes codificadoras das técnicas baseadas em *autoencoders* variacionais, e nas redes discriminativas das técnicas baseadas em redes adversárias generativas (GANs), realizando o processamento típico dos filtros convolutivos para reduzir gradualmente as dimensões do dado de entrada, isto é, para realizar o processo conhecido como *downsampling*.

Porém, tais modelos generativos também exploram outro tipo de estrutura (e.g., um decodificador, no caso dos *autoencoders*, ou o gerador, no caso da GAN) que realiza um processo reverso de ampliação das dimensões do dado de entrada (*upsampling*). Nesse caso, camadas convolucionais transpostas são empregadas, pois têm a capacidade de gradualmente aumentar as dimensões de entrada utilizando *kernels* ou filtros espaciais, como exemplificado na Figura 2.12.

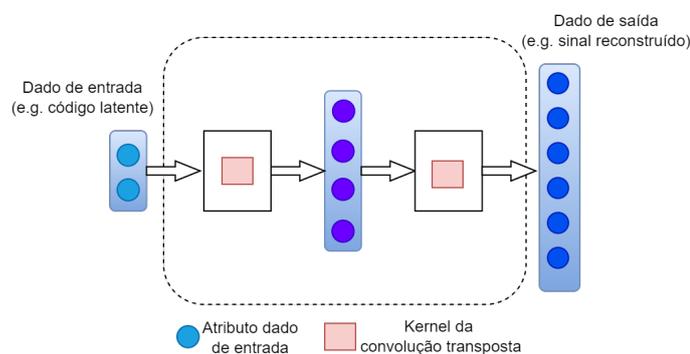


Figura 2.12 – Diagrama representando uma rede neural com camadas convolucionais transpostas 1D.

A Figura 2.13 apresenta a operação de convolução transposta realizada pelo *kernel* da Figura 2.12. Na convolução transposta, para cada atributo do dado de entrada é produzida uma saída parcial de mesma dimensão do *kernel* e com os elementos dados

pela multiplicação de cada parâmetro pelo atributo. Por exemplo, na primeira camada intermediária, o atributo do dado de entrada x_1 tem uma saída parcial z_1 com cada elemento dado pela multiplicação de cada parâmetro do *kernel* $\{w_1, w_2, w_3\}$ pelo atributo x_1 . Da mesma forma, o atributo x_2 tem uma saída parcial z_2 com cada elemento dado pela multiplicação de cada parâmetro do *kernel* $\{w_1, w_2, w_3\}$ pelo atributo x_2 . Após a obtenção de todas as saídas parciais, no caso z_1 e z_2 , as saídas são posicionadas sequencialmente na mesma ordem dos atributos do dado de entrada, mas com a posição inicial de cada saída parcial sendo deslocada em uma posição com relação à saída parcial anterior, sendo o valor igual a um no deslocamento determinado pelo parâmetro *stride* pré-definido. Por fim, com todas as saídas parciais posicionadas, cada elemento da saída final y é dado pela soma dos elementos correspondentes à mesma posição de cada saída parcial.

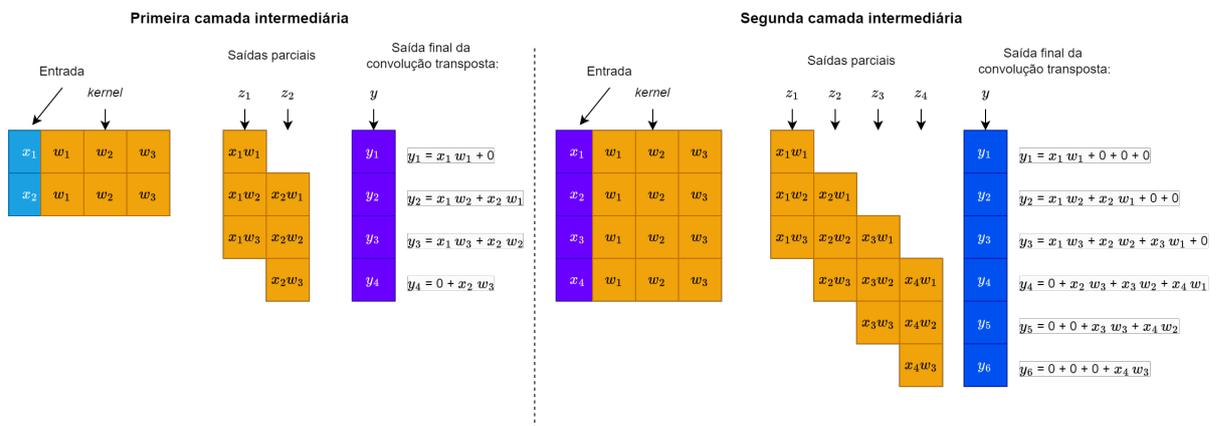


Figura 2.13 – Operação de um *kernel* aplicando convolução transposta em uma amostra de entrada.

2.5 Fechamento

Neste capítulo apresentamos os fundamentos dos modelos de classificação utilizados, que incluem a SVM linear, a SVM não-linear e a CNN-1D, cuja arquitetura é baseada no modelo proposto em (AZNAN *et al.*, 2018) e que é explicado em detalhes na Seção 4.2.1. Esses modelos são empregados para processar dois tipos possíveis de dados de entrada: EEGs-SSVEP no domínio do tempo (sinais brutos de EEG) e EEGs-SSVEP no domínio da frequência (espectros dos EEGs), e seus desempenhos são calculados para um conjunto de dados de validação e utilizados como métrica para comparação da qualidade e do impacto dos dados artificiais nas abordagens com *data augmentation*.

No próximo capítulo apresentaremos as técnicas para obtenção dos modelos generativos que são utilizados nos experimentos computacionais para produzir os EEGs-SSVEP artificiais no domínio do tempo e no domínio da frequência empregados no treinamento dos classificadores com bases de dados aumentadas.

3 Fundamentos dos modelos generativos

Modelos generativos compreendem uma área do aprendizado de máquina que trata da geração de novos dados, chamados de dados sintéticos ou artificiais, cujas características sejam as mais próximas possíveis daquelas apresentadas pelos dados reais, tornando-os idealmente indistinguíveis. As técnicas para obtenção de modelos generativos partem da hipótese da existência de uma distribuição de probabilidade P_{data} que, ao ser amostrada, permite a geração dos dados reais (DOERSCH, 2016). Com isso, o objetivo dessas técnicas pode ser resumido em aproximar essa distribuição de probabilidade P_{data} e, assim, utilizar o modelo generativo obtido para sintetizar novas amostras.

Neste capítulo, apresentamos a ideia por trás de duas técnicas fundamentais e de grande relevância na área de modelos generativos, o *autoencoder* variacional (VAE, do inglês *Variational Autoencoder*), proposto em (KINGMA; WELLING, 2014), e as redes adversárias generativas (GANs, do inglês *Generative Adversarial Networks*), propostas em (GOODFELLOW *et al.*, 2014), que formam a base de todas as técnicas utilizadas neste trabalho.

3.1 Autoencoder Variacional (VAE)

O *autoencoder* variacional é composto por uma rede codificadora (*encoder* Q) que recebe como entrada um dado real X , e por uma rede decodificadora (*decoder* P) que busca reconstruir o dado de entrada X a partir de uma variável latente z amostrada da distribuição de probabilidade $Q(z|X)$ obtida com o mapeamento do *encoder* (Figura 3.1). No final do processo, caso o treinamento seja bem sucedido, o *encoder* é descartado e o *decoder* torna-se o modelo generativo responsável por sintetizar os dados artificiais, sendo a distribuição $Q(z|X)$ o estímulo de entrada (Figura 3.1).

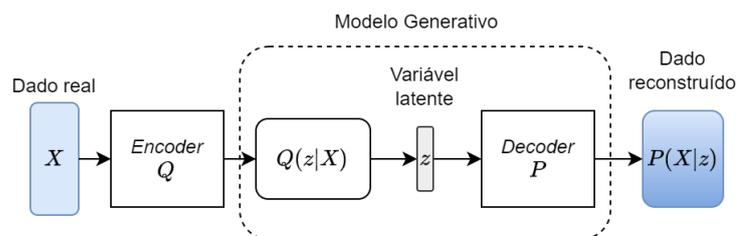


Figura 3.1 – Versão simplificada do *autoencoder* variacional.

Como mencionado no início deste capítulo, o desafio de um modelo generativo é obter uma aproximação para a distribuição de probabilidade P_{data} que gera os dados reais. A proposta apresentada pelo VAE não faz suposições sobre o formato da distribuição

P_{data} , e se utiliza da propriedade de que qualquer distribuição de probabilidade pode ser obtida a partir da distribuição normal padrão, isto é, da distribuição normal que possui média igual a zero e desvio padrão igual a um (DOERSCH, 2016). Com essa ideia, o VAE busca, em sua formulação, mapear com o *encoder* os dados reais X em uma distribuição normal padrão e, em seguida, utilizar o *decoder* para reconstruir o dado de entrada X na saída do modelo que, ao mesmo tempo, mapeia internamente (na própria estrutura do *decoder*) a distribuição normal padrão em uma distribuição que aproxima P_{data} , conforme apresentado na Figura 3.2. No final do processo de treinamento do VAE, o modelo generativo resultante consiste na rede decodificadora gerando sinais artificiais a partir de amostras da distribuição normal padrão de entrada. Caso a aproximação da distribuição P_{data} seja adequada, os dados sintetizados pelo *decoder* tendem a ser verossímeis e com características semelhantes aos dados reais.

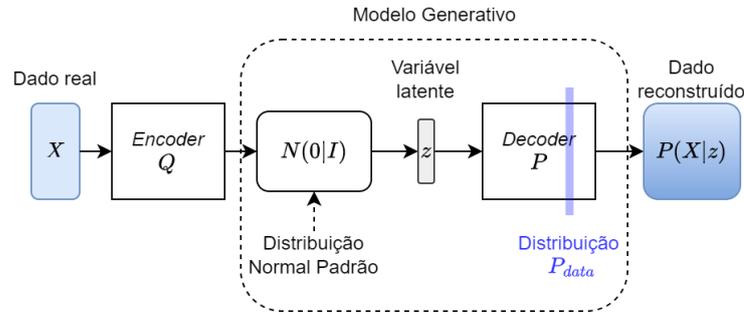


Figura 3.2 – Versão simplificada final do *autoencoder* variacional com a formulação proposta no problema.

Outro ponto importante que deve ser destacado antes de partirmos para a formulação matemática é que o problema de obter a distribuição P_{data} torna-se computacionalmente tratável devido à limitação no espaço de busca imposta pelo *encoder*, que faz a técnica levar em consideração apenas as variáveis z mais prováveis de terem gerado X . Com o cenário apresentado, o critério a ser minimizado durante o treinamento do VAE é apresentado na Equação (3.1).

$$\min \mathbb{D}[Q(z|X)||P(z)] + \mathbb{E}_{z \sim Q}[-\log P(X|z)], \quad (3.1)$$

em que $\mathbb{D}[\cdot]$ representa a Divergência de Kullback-Leibler entre a distribuição $Q(z|X)$ correspondente ao *encoder* e a distribuição alvo $P(z)$. Na formulação do problema, $P(z)$ é a distribuição normal padrão $N(0, I)$ e $Q(z|X)$ é uma distribuição normal com médias e variâncias definidas pelo mapeamento da rede codificadora, isto é, $N(z|\mu(X), \Sigma(X))$. Logo, o primeiro termo busca aproximar a distribuição $Q(z|X)$ à distribuição normal padrão $N(0, I)$. Já o segundo termo representa o erro associado à reconstrução das amostras de entrada X , que pode ser qualquer medida que leve em consideração a diferença entre os atributos correspondentes do dado reconstruído $P(X|z)$ e do dado real de entrada X .

Para tornar o problema de otimização computacionalmente menos custoso e permitir o uso de técnicas de otimização baseadas em gradiente, a esperança $\mathbb{E}_{z \sim Q}[-\log P(X|z)]$ é aproximada por apenas uma única amostra $-\log P(X|z)$, procedimento necessário uma vez que a operação de amostragem é não diferenciável. A Equação (3.2) apresenta novamente o critério de otimização, agora com essa simplificação.

$$\min \mathbb{D}[Q(z|X)||N(0, I)] - \log P(X|z), \quad (3.2)$$

Infelizmente, a simplificação no erro de reconstrução faz desaparecer a dependência de $P(X|z)$ com $Q(z|X)$, também impossibilitando o uso de técnicas de retropropagação de erro. Para contornar essa inviabilidade, utiliza-se o Truque da Reparametrização (do inglês, *Reparameterization Trick*), em que a variável aleatória z é gerada pela composição do mapeamento determinístico $Q(z|X)$ do *encoder* com uma variável ϵ amostrada de uma distribuição normal padrão $N(0, I)$ que está fora do fluxo que interliga o *decoder* com o *encoder*, de maneira que passa a ser possível propagar derivadas da função custo desde o *decoder* até as camadas do *encoder*. A Figura 3.3 apresenta a estrutura final do VAE e a Equação (3.3) apresenta a função custo (ou função perda), que foi possível ser obtido analiticamente pois as duas distribuições envolvidas são distribuições normais.

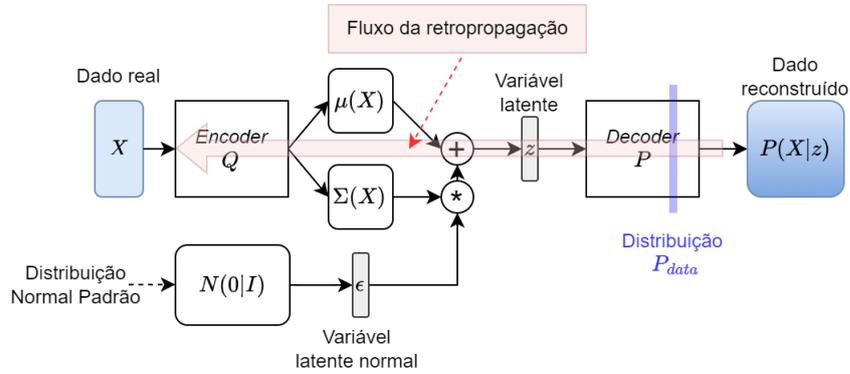


Figura 3.3 – Versão final do *autoencoder* variacional com o truque da reparametrização.

$$\min \mathbb{D}[N(z|\mu(X), \Sigma(X))||N(0, I)] + \mathbb{E}_{X \sim \text{dataset}} [\mathbb{E}_{\epsilon \sim N(0, I)} [-\log P(X|z = \mu(X) + \Sigma^{1/2}(X) * \epsilon)]] \quad (3.3)$$

Por fim, como já discutido anteriormente, após o treinamento do *autoencoder* variacional, o *encoder* é descartado e o *decoder* é tomado como o modelo generativo, que recebe como entrada amostras da distribuição normal padrão (Figura 3.4).

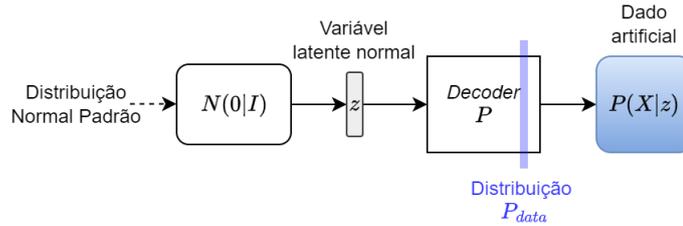


Figura 3.4 – Modelo generativo obtido com o VAE.

3.2 Redes Adversárias Generativas (GAN)

A proposta utilizada pela GAN para obter o modelo generativo e, implicitamente, aproximar a distribuição P_{data} que gera os dados reais, envolve o confronto entre duas redes: a discriminativa D e a generativa G . A rede geradora (ou generativa) G é responsável por criar amostras completas que sejam compatíveis com amostras reais do domínio de interesse, cujas características imitem com fidelidade aquelas que os dados reais possuem. Para isso, a rede G realiza um mapeamento de um vetor latente z , tomado de uma distribuição de probabilidades pré-estabelecida (tipicamente, a distribuição Gaussiana padrão) em um dado completo (e.g., uma imagem ou uma série temporal).

Por sua vez, a rede discriminadora é um classificador que tem como tarefa identificar se a amostra apresentada em sua entrada é real, caso pertença à base de dados de treinamento, ou sintética, caso tenha sido produzida pela rede generativa, conforme diagrama apresentado na Figura 3.5.

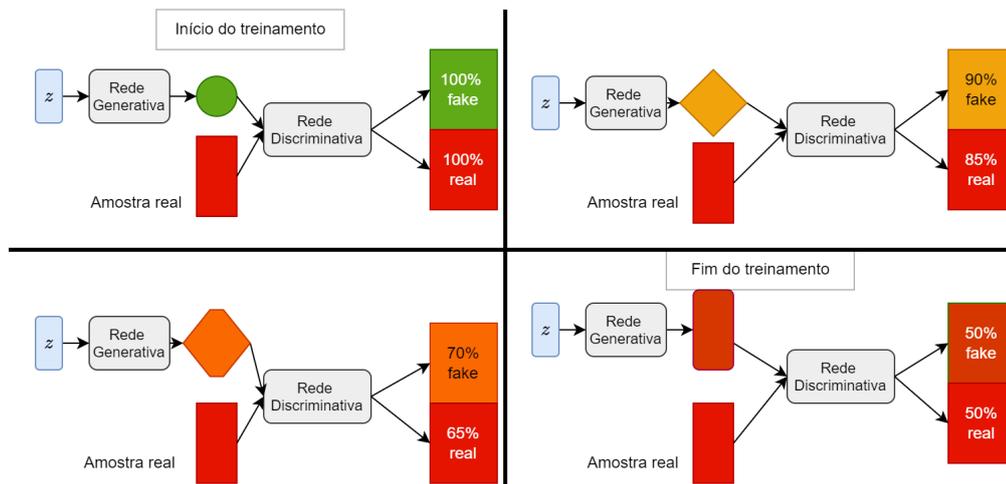


Figura 3.5 – Diagrama representando esquematicamente o treinamento de uma GAN.

No início do treinamento a rede generativa sintetiza amostras completamente distintas das amostras reais, tornando fácil a tarefa da rede discriminativa. Conforme o treinamento progride, a rede generativa sintetiza amostras cada vez mais semelhantes às amostras reais, dificultando gradativamente a identificação da origem do dado pela rede discriminativa. Concomitantemente, a rede discriminativa também busca melhorar seu

desempenho, dificultando o lado da rede generativa, que precisa gerar amostras cada vez melhores. No fim do processo de treinamento, é desejado que a rede generativa se torne tão boa que seja capaz de produzir amostras idealmente indistinguíveis das amostras reais, reduzindo a taxa de acerto da rede discriminadora para 50%, que configura o caso de indeterminação completa por parte desse classificador. Assim, independente do quão boa era a rede discriminativa durante o processo de treinamento, no final o melhor que ela pode fazer é “chutar” se a amostra apresentada é real ou sintética.

O confronto entre as duas redes fica evidente na função custo proposta por (GOODFELLOW *et al.*, 2014), mostrada na Equação (3.4), onde $D(X)$ e $D(G(z))$ representam, respectivamente, as probabilidades associadas às amostras reais e sintéticas na saída da rede discriminativa. Nessa formulação, a probabilidade igual a um indica que as amostras são reais e a probabilidade igual a zero indica que as amostras são sintéticas. Diante deste cenário, a rede discriminativa busca maximizar seu desempenho classificando amostras sintéticas como falsas (probabilidade próxima de zero) e amostras reais como verdadeiras (probabilidade próxima de um), enquanto a rede generativa busca maximizar seu desempenho “enganando” a rede discriminativa, fazendo com que ela classifique amostras sintéticas como verdadeiras (probabilidade próxima de um), resultando no que é conhecido como jogo *minmax*.

$$\min_G \max_D \mathbb{E}_{X \sim P_{data}} [\log D(X)] + \mathbb{E}_{z \sim P_z} [\log(1 - D(G(z)))] \quad (3.4)$$

Embora a ideia da GAN seja simples, seu treinamento não é uma tarefa fácil, pois a busca pelo equilíbrio entre as duas redes apresenta um comportamento extremamente instável. Diante dessa dificuldade, vários estudos e arquiteturas foram propostas com o intuito de tornar o treinamento adversário da GAN mais estável. A seguir apresentamos três dessas propostas: a DCGAN (RADFORD *et al.*, 2015), a WGAN (ARJOVSKY *et al.*, 2017) e a AC-GAN (ODENA *et al.*, 2017).

3.2.1 Deep Convolutional GAN (DCGAN)

No início as GANs eram desenvolvidas quase que exclusivamente para geração de imagens e, por esse motivo, muitas das arquiteturas de redes generativa e discriminativa se baseavam nos modelos de classificação de imagens da competição ImageNet. Um dos modelos vencedores da competição, VGG16 (SIMONYAN; ZISSERMAN, 2015), propôs um módulo de processamento muito replicado pela comunidade com as seguintes etapas: (1) aplicação de filtros convolutivos sem mudança na dimensão das *feature maps* de entrada, (2) aplicação de uma operação de *pooling* para reduzir pela metade a dimensão das *feature maps* e (3) aplicação de uma função de ativação não-linear, geralmente a

ReLU.

Contudo, as etapas (2) e (3) presentes no módulo de processamento da VGG16 provocam perdas significativas de informação dos dados durante o processamento, dificultando muito a estabilidade no treinamento da GAN. Diante disto, a DCGAN proposta por (RADFORD *et al.*, 2015) busca definir um conjunto de hiperparâmetros que tornam o treinamento das GANs com redes profundas mais estável. As principais modificações propostas foram:

- Substituição das etapas (1) e (2) por filtros convolutivos que já realizam a redução da dimensão das *feature maps* na rede discriminativa; por sua vez, na rede generativa os filtros convolutivos realizam o aumento da dimensão das *feature maps* (de volta à dimensão dos dados originais) por meio de camadas convolucionais transpostas;
- Substituição das funções de ativação *ReLU* por *Leaky ReLU*;
- Normalização das amostras reais de treinamento para o intervalo $[-1, 1]$ e o uso da função de ativação *tanh* na camada de saída da rede generativa para que os atributos das amostras sintéticas também fiquem no mesmo intervalo $[-1, 1]$;
- Uso da técnica de *batch normalization* para reduzir problemas relacionados a inicializações ruins e melhorar o fluxo do gradiente durante a retropropagação do erro no processo de treinamento (IOFFE; SZEGEDY, 2015);
- Remoção das camadas densas (*fully-connected*), exceto a primeira da rede generativa e a última da rede discriminativa.

Dessa forma, a estrutura geral de uma DCGAN segue o modelo apresentado na Figura 3.6.

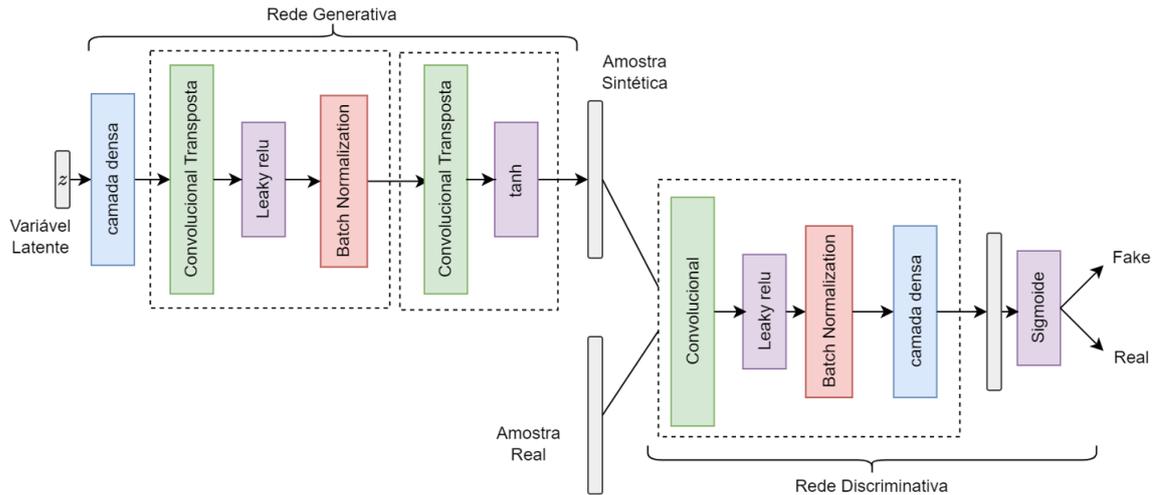


Figura 3.6 – Exemplo de estruturas das redes generativa e discriminativa de uma arquitetura do tipo DCGAN.

3.2.2 Wasserstein GAN (WGAN)

A *Wasserstein* GAN (ARJOVSKY *et al.*, 2017) é uma proposta que torna o treinamento menos sensível à escolha da arquitetura das redes adversárias e dos hiperparâmetros. A principal mudança com relação à GAN original está na substituição da rede discriminativa por uma rede chamada de crítica, que ao invés de classificar a amostra como real ou sintética fornece uma medida do quão real ou quão falsa aquela amostra aparenta ser. O interessante desta abordagem é que a nova função custo, *Wasserstein*, também está relacionada com a qualidade da amostra sintetizada. As principais modificações propostas na WGAN são:

- Substituição da função de ativação sigmoide na camada de saída da rede discriminativa por uma função de ativação linear na rede crítica.
- Substituição da função custo de Entropia Cruzada pela função custo *Wasserstein* dada pela Equação (3.5), em que $\Pi(\mathbb{P}_r, \mathbb{P}_g)$ denota o conjunto de todas as distribuições de probabilidade conjuntas em que as distribuições marginais são \mathbb{P}_r e \mathbb{P}_g , respectivamente. Intuitivamente, $\gamma(x, y)$ indica quanta “massa” deve ser transportada de x para y para transformar as distribuições \mathbb{P}_r na distribuição \mathbb{P}_g .

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y) \sim \gamma} [\|x - y\|] \quad (3.5)$$

- Como na WGAN as saídas da rede crítica não representam mais probabilidades, os rótulos para as amostras sintéticas correspondem a 1, enquanto para as amostras reais eles passam a ser -1 .

- Limitação no valor dos pesos sinápticos de ambas as redes para o intervalo $[-0.01, 0.01]$

É pertinente mencionar que em cada época de treinamento da WGAN, ocorrem N iterações de ajuste dos parâmetros da rede crítica para cada iteração de ajuste da rede generativa.

3.2.3 Auxiliary Classifier GAN (AC-GAN)

Embora as técnicas anteriores sejam capazes de sintetizar amostras de diferentes classes com um único modelo generativo, os VAEs e, especialmente as GANs, podem sofrer com um fenômeno denominado *mode collapse*, em que durante o treinamento o modelo generativo se especializa em produzir dados com padrões limitados, falhando em capturar corretamente a diversidade da distribuição dos dados de entrada como, por exemplo, na capacidade de sintetizar amostras de classes distintas. Para contornar esse problema é comum o treinamento de um modelo generativo para cada classe do problema abordado, o que aumenta o custo computacional no desenvolvimento de modelos generativos baseados em GANs e em VAEs. Nesse sentido, a AC-GAN apresentada nesta seção é uma das técnicas que possibilita obter um único modelo generativo capaz de produzir amostras de mais de uma classe, e que permite controlar a classe da amostra sintetizada por meio de uma entrada adicional c que representa a classe que se deseja sintetizar (ODENA *et al.*, 2017).

A AC-GAN se baseia no conceito de aprendizado multi-tarefa (*multi-task learning* (ZHANG; YANG, 2022)), onde os modelos envolvidos são treinados para resolver mais de um problema simultaneamente. Nesta arquitetura, uma segunda camada de saída é anexada em paralelo à camada de saída principal da rede discriminativa da GAN, compartilhando as mesmas camadas intermediárias, conforme mostra a Figura 3.7, e tem como tarefa identificar a classe à qual uma amostra real ou uma amostra sintética pertence. Desta forma, a função objetivo a ser minimizada durante o treinamento da AC-GAN contém duas componentes: (i) a função objetivo da própria GAN, apresentada na Equação (3.4), e (ii) a função de perda correspondente à entropia cruzada associada ao problema de identificar as classes das amostras, apresentada na Equação (3.6)

$$Loss_{CE} = - \sum_{i=0}^{M-1} \sum_{c=1}^N y_{i,c} \log \hat{y}_c(X_i), \quad (3.6)$$

em que M é o número de amostras, N é o número de classes distintas, \hat{y} é a probabilidade de saída inferida pelo classificador auxiliar e $y_{i,c}$ é a saída desejada dada pela seguinte expressão:

$$y_{i,c} = \begin{cases} 1, & \text{se } X_i \in \text{classe } c \\ 0, & \text{caso contrário} \end{cases} \quad (3.7)$$

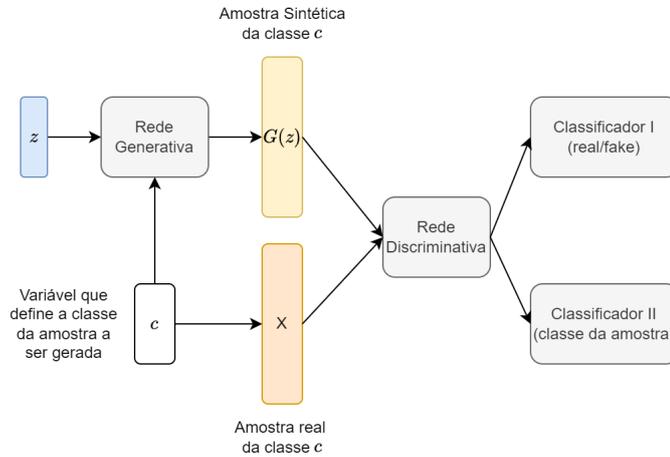


Figura 3.7 – Diagrama de uma rede do tipo AC-GAN.

3.2.4 Adversarial Autoencoder (AAE)

O *autoencoder* adversário (MAKHZANI *et al.*, 2016) é uma técnica que reúne características do *autoencoder* variacional e das redes adversárias generativas para aproximar a distribuição de probabilidade P_{data} que gera os dados. Da mesma forma que o VAE, o *autoencoder* adversário é formado por uma rede codificadora (*encoder*) que mapeia a entrada X em uma distribuição $Q(z|X)$ e por uma rede decodificadora (*decoder*) que busca reconstruir a entrada X a partir de uma variável aleatória z amostrada de $Q(z|X)$, como mostra a Figura 3.8.

A diferença entre as duas técnicas está no critério que relaciona a distribuição $Q(z|X)$ e a distribuição alvo, agora denotada por $P(w)$, conforme indicado na Figura 3.8. No caso do *autoencoder* variacional, essa relação é dada pelo termo de regularização na função custo (3.1) definida pela Divergência de Kullback-Leibler, cuja expressão é obtida analiticamente para o caso em que as duas distribuições envolvidas são distribuições normais. No *autoencoder* adversário, o termo de regularização que relaciona as duas distribuições é dado com o treinamento adversário, portanto substituindo a Divergência de Kullback-Leibler pelo jogo *minmax* das redes adversárias discutido na Seção 3.2.

As amostras reais e sintéticas no treinamento adversário desse modelo correspondem às variáveis latentes amostradas, respectivamente, da distribuição alvo $P(w)$ e da distribuição do *encoder* $Q(z|X)$. Nesse sentido, o *encoder* atua como a rede generativa do treinamento adversário e a rede discriminativa tem como tarefa identificar se a variável latente apresentada tem origem da distribuição alvo $P(w)$ (real) ou da distribuição do *encoder* $Q(z|X)$ (sintética).

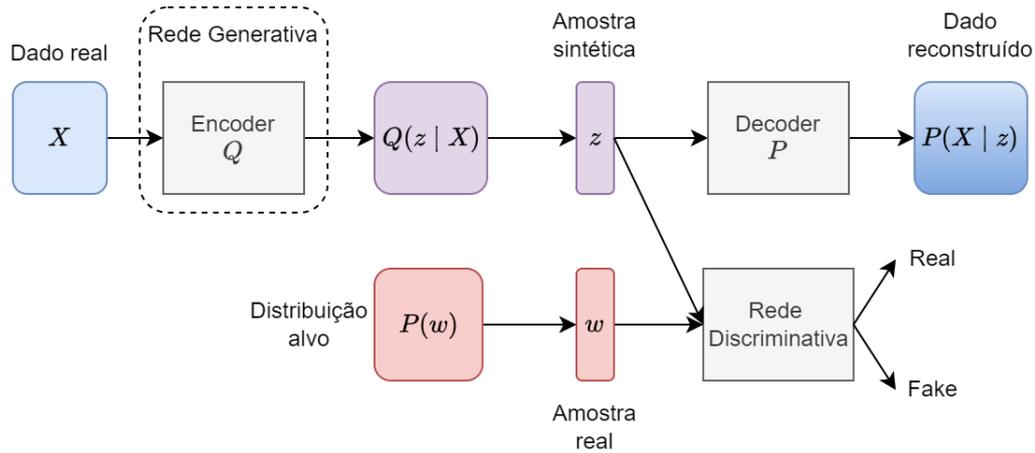


Figura 3.8 – Diagrama da estrutura geral de um *autoencoder* adversário.

O treinamento do *autoencoder* adversário envolve duas etapas principais: (1) a etapa de reconstrução, cujo objetivo é minimizar o erro entre a amostra reconstruída e a entrada X e (2) a etapa de regularização, que consiste no treinamento adversário entre o *encoder* e a rede discriminativa. É importante frisar a existência de dois modelos generativos nessa arquitetura, o primeiro correspondendo à rede generativa do treinamento adversário que sintetiza códigos latentes e o segundo correspondente ao *decoder* que sintetiza as amostras no domínio no qual estamos interessados.

Essa nova forma de relacionar as duas distribuições traz alguns pontos positivos. O primeiro deles está na possibilidade de utilizar distribuições $P(w)$ e $Q(z|X)$ que não sejam apenas distribuições normais, pois não é mais necessário obter analiticamente uma expressão para a Divergência de Kullback-Leibler que as relacione. O segundo ponto está na redução das dimensões das amostras reais e sintéticas do treinamento adversário, pois as variáveis latentes possuem dimensões bem menores do que os dados de interesse no domínio original. O terceiro ponto positivo está na possibilidade de utilizar todas as modificações propostas pela DCGAN, WGAN e AC-GAN para melhorar ainda mais o treinamento do modelo.

3.3 Modelos generativos condicionados

Como apresentado na Seção 3.2.3, as técnicas baseadas no VAE e nas GANs sofrem com o fenômeno de *mode collapse* (KOSSALE *et al.*, 2022), o que dificulta a obtenção de modelos generativos capazes de produzir dados de classes diferentes. Na mesma Seção 3.2.3 foi apresentada a técnica AC-GAN, que é uma das maneiras possíveis para obter um único modelo generativo capaz de sintetizar amostras de várias classes e que permite controlar a classe do dado sintetizado. Para isso, a técnica AC-GAN incorpora no dado de entrada o rótulo da classe e adiciona o problema de identificar a classe à qual

a amostra real ou sintética pertence.

O uso dos rótulos das classes como informação adicional aos modelos pode ser estendido para todas as técnicas discutidas neste capítulo. Assim, podemos obter modelos generativos com a característica de gerar amostras sintéticas de mais de uma classe fornecendo, nas técnicas VAE, GAN, DCGAN, WGAN e AAE, a classe a qual desejamos que a amostra sintética pertença (incorporando essa informação na variável aleatória z de entrada das redes generativa e decodificadora) e a classe a qual a amostra real de treinamento pertence (incorporando essa informação no próprio dado real).

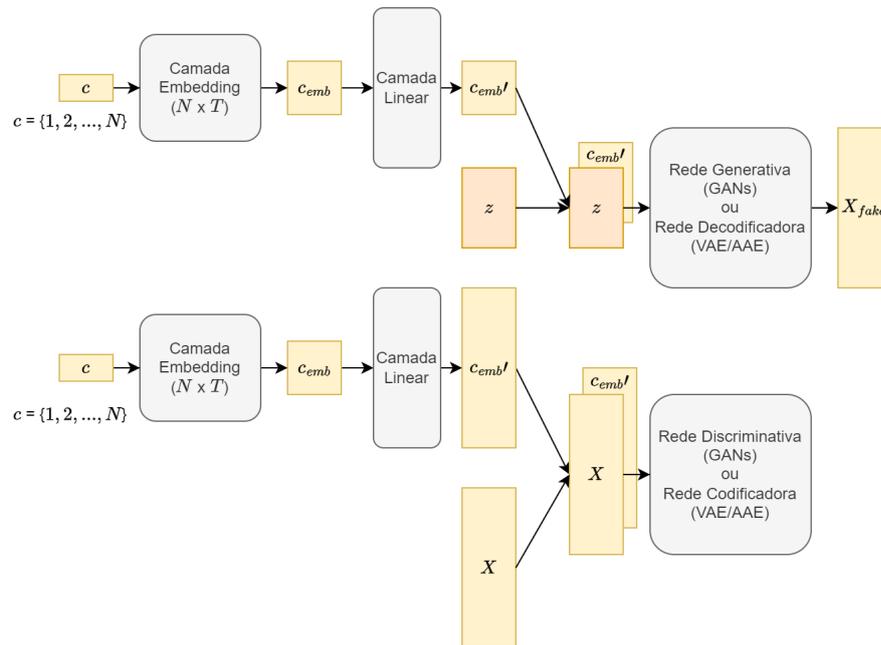


Figura 3.9 – Uso da camada de *embedding* para incorporar a informação do rótulo da classe.

A Figura 3.9 apresenta como o rótulo da classe é incorporado aos dados durante o processo de treinamento. Para o caso geral em que desejamos obter um único modelo generativo para sintetizar amostras de N classes distintas, associamos inicialmente à cada classe um número natural de 1 a N . Em seguida, passamos esse rótulo categorizado pela camada de *embedding*, que consiste em uma matriz de parâmetros ajustáveis de dimensão $N \times T$, em que cada linha está associada a uma das N classes do problema e seus T elementos formam um vetor de números reais que representa a classe correspondente no espaço de *embeddings* (BROWNLEE, 2021).

Por fim, o vetor de *embedding* passa por uma camada linear cuja quantidade de parâmetros é ou múltipla do número de atributos da variável latente z , quando incorporamos a informação do rótulo da classe na rede generativa das GANs ou na rede decodificadora do VAE/AAE, ou do número de atributos da amostra real X ou sintética X_{fake} , quando incorporamos a informação do rótulo da classe na rede discriminativa das

GANs ou na rede codificadora do VAE/AAE. Por fim, a saída da camada linear é anexada à variável latente ou ao dado real/sintético como um canal adicional. Ao final do treinamento, os parâmetros das camadas de *embedding* e linear são obtidos conjuntamente com os do modelo generativo, e o controle da classe da amostra a ser sintetizada é feito passando o rótulo categorizado (1 a N) da classe desejada.

3.4 Fechamento

Neste capítulo, apresentamos os fundamentos teóricos de uma ampla gama de modelos generativos que serão empregadas para a síntese de sinais de EEG-SSVEP, tanto no domínio do tempo quanto no domínio da frequência. As arquiteturas investigadas incluem DCGAN, VAE, WGAN, AC-GAN, cDCGAN, cVAE, cWGAN, AAE e cAAE. Com esses modelos seremos capazes de sintetizar EEGs-SSVEP no domínio do tempo, gerando sinais brutos que podem ser diretamente utilizados pelos classificadores da BCI, ou no domínio da frequência, onde sintetizamos os espectros dos sinais de EEG. Avaliaremos a eficácia desses modelos na geração de dados artificiais de alta qualidade que podem ser utilizados para treinar classificadores de BCIs, examinando como os dados artificiais gerados por eles impactam tanto classificadores mais simples, como SVM lineares e não-lineares, quanto classificadores mais complexos, como CNNs-1D.

4 *Data augmentation* aplicado a BCIs-SSVEP

Um dos principais desafios no desenvolvimento de BCIs é a escassez de amostras de sinais cerebrais, que é consequência tanto do próprio procedimento experimental para aquisição dos sinais cerebrais, pois retorna uma quantidade reduzida de amostras que dificulta a obtenção de modelos de classificação de alto desempenho, quanto da diversidade de manifestações que o cérebro do usuário pode apresentar durante o uso, exigindo frequente retreinamento do classificador para que a BCI apresente desempenho satisfatório em diferentes situações.

A diversidade de manifestações do cérebro humano pode ser causada por diversos fatores, dentre eles, mudanças fisiológicas, mudanças de ambiente ou mudanças no estado físico, mental e emocional do usuário. Como consequência, o usuário pode apresentar respostas cerebrais completamente distintas que levam seus sinais cerebrais a apresentarem comportamento não-estacionário, isto é, com propriedades estatísticas que variam ao longo do tempo, e que são responsáveis por degradar a capacidade de generalização do classificador da BCI (AZNAN *et al.*, 2019; AZNAN *et al.*, 2020).

Já o número limitado de amostras obtidas durante a etapa de aquisição dos sinais cerebrais decorre das características do procedimento experimental, que tende a ser oneroso, demorado e cansativo para o usuário, pois exige a calibração da BCI e várias repetições do experimento para que ao final do procedimento se consiga, no melhor cenário, algumas dezenas de amostras de sinais cerebrais. Esse quadro é ainda mais penoso para novos usuários ou aqueles que são menos experientes, pois exige também um treinamento com relação a como utilizar a BCI.

Diante do cenário apresentado, diversas técnicas foram propostas com o objetivo de mitigar o problema da escassez de dados no desenvolvimento de BCIs e, atualmente, podemos agrupá-las em três grandes abordagens: as clássicas técnicas de *feature engineering*, as técnicas baseadas em *data augmentation* e as técnicas mais recentes que utilizam conceitos relacionados a *transfer learning*.

As técnicas de *feature engineering* (engenharia de atributos) buscam mitigar a escassez de amostras atacando o conhecido problema da maldição da dimensionalidade (BELLMAN *et al.*, 1957), em que o volume de dados necessário para treinar adequadamente um modelo tem relação exponencial com o número de atributos que as amostras do problema possuem (BELLMAN *et al.*, 1957; GERON, 2022). No contexto de BCIs os

sinais cerebrais tendem a apresentar um número elevado de atributos, e o emprego de técnicas de *feature engineering* para extração e seleção dos atributos mais relevantes e discriminativos têm se mostrado capaz de reduzir a dimensão dos dados e, consequentemente, a necessidade por grandes volumes de amostras. Nesta abordagem podem ser aplicadas técnicas de extração de características como a transformada de Fourier, transformada de Wavelet, espectrogramas, bancos de filtros, decomposição com PCA (*Principal Component Analysis*) (OPPENHEIM *et al.*, 1999; DUDA *et al.*, 2001; BISHOP, 2007; CARVALHO, 2016), e também técnicas de seleção de atributos como filtros e *wrappers* (GUYON; ELISSEEFF, 2003).

Nos últimos anos o problema da escassez de amostras tem sido abordado com o auxílio de técnicas em que aumentam artificialmente a quantidade de dados, as quais compreendem a linha de *data augmentation*, em que as amostras reais obtidas durante o processo de experimentação são utilizadas para gerar sinais sintéticos que apresentam características e comportamentos semelhantes aos dos sinais reais, sendo possível incorporá-los no conjunto de dados de treinamento dos classificadores da BCI e, assim, potencialmente melhorar seus desempenhos. Diferentes técnicas têm sido empregadas para este fim, desde as mais simples como inserção de ruídos até o uso de modelos complexos de aprendizado profundo que se mostraram efetivos em problemas de processamento de imagens, como os VAEs (KINGMA; WELING, 2014), as GANs (GOODFELLOW *et al.*, 2014) e outros modelos generativos, como os modelos de difusão (HO *et al.*, 2020).

Embora os modelos generativos anteriores apresentem potencial e capacidade de melhorar o desempenho do classificador da BCI, eles ainda exigem que o usuário realize o procedimento experimental para aquisição dos sinais cerebrais todas as vezes que o classificador precisar ser treinado ou retreinado. Diante disso, trabalhos mais recentes tem apresentado propostas que associam técnicas de *data augmentation*, em particular os modelos generativos, a técnicas de *transfer learning* com o objetivo de reduzir ou até remover a necessidade de novos usuários realizarem todo o processo de experimentação, ao mesmo tempo em que diminuem o problema decorrente da escassez de dados. Tipicamente, essas técnicas envolvem o uso de sinais cerebrais de vários usuários já conhecidos para obter modelos generativos capazes de (i) extrair a componente dos sinais que é comum a todos os usuários, (ii) sintetizar amostras artificiais genéricas correspondentes a essa parte comum, (iii) extrair as características particulares dos sinais cerebrais de cada usuário e (iv) incorporar as características particulares às amostras artificiais genéricas para produzir os sinais artificiais correspondentes a um usuário específico (AZNAN *et al.*, 2020; KWON; IM, 2022).

Neste capítulo, apresentamos os trabalhos presentes na literatura que abordam o problema da escassez de amostras com técnicas de *data augmentation* no contexto de

BCIs-SSVEP baseadas em EEGs. A Seção 4.1 inicia a revisão bibliográfica com o trabalho de (LASHGARI *et al.*, 2020), que fornece uma visão geral de como várias técnicas de *data augmentation* têm sido utilizadas em BCIs com EEGs nos mais diversos contextos. Dada essa visão geral, a Seção 4.2 destaca os três principais trabalhos identificados na revisão bibliográfica e que abordam a síntese de EEGs-SSVEP artificiais com o auxílio de modelos generativos. Por fim, apresentamos na Seção 4.3 a proposta deste trabalho.

4.1 *Data augmentation* em BCIs com EEGs

Em (LASHGARI *et al.*, 2020), os autores compilaram os trabalhos encontrados na literatura que tratavam da síntese de EEGs artificiais em sistemas BCIs. Para isso, realizaram buscas por *papers* de conferências, artigos de revistas e pré-prints eletrônicas nas bases de dados do Google Scholar, Web of Science e PubMed, que abordassem o uso de técnicas de *data augmentation* para EEGs em BCIs voltadas a tarefas de classificação com modelos de *deep learning*. Após uma criteriosa análise dos 263 trabalhos encontrados, 53 foram selecionados para compor o estudo por atenderem todos os requisitos estabelecidos.

Tabela 4.1 – Quadro resumo da distribuição dos 53 trabalhos compilados em (LASHGARI *et al.*, 2020) com relação ao tipo de tarefa de classificação da BCI, arquitetura de classificador, formato do dado de entrada e presença ou ausência de pré-processamento.

Tarefa de Classificação	%	Arquitetura do Classificador	%
Detecção de convulsão	24	CNN	62
Imaginação de movimentos	21	Híbrido	16
Estágios do sono	15	MLP	8
Reconhecimento de emoções	15	SAE	6
Tarefa cognitiva	9	LSTM	6
Tarefa motora	8	RNN	2
Tarefa visual	8		
Dado de Entrada	%	Pré-processamento	%
Sinal bruto de EEG	36	Remoção manual de artefatos com algum filtro passa-alta, passa-baixa e/ou passa-faixa	85
Feature calculada	49		
Espectrograma	15		

A Tabela 4.1 apresenta um quadro que fornece a distribuição dos 53 trabalhos selecionados de acordo com (i) a tarefa de classificação envolvida, (ii) a arquitetura utilizada pelo classificador da BCI, (iii) o formato do dado de entrada e (iv) se algum pré-processamento foi realizado sobre o dado de entrada. Podemos notar que a maior parte dos trabalhos de *data augmentation* em BCIs com EEGs se concentram nas tarefas de

detecção de convulsão (24%) e imaginação de movimentos (21%), sendo a primeira uma tarefa ainda mais desafiadora por se tratar de um problema com classes desbalanceadas, já que há um número consideravelmente menor de amostras ligadas à ocorrência de um quadro de convulsão do que de amostras associadas ao funcionamento normal do cérebro, e a segunda uma abordagem clássica em BCIs com finalidade de controlar algum dispositivo. Ainda com relação às tarefas de classificação, podemos verificar que a quantidade de estudos voltados ao grupo de tarefas visuais, no qual se encaixa a abordagem baseada no paradigma SSVEP, é bem reduzida (8%), o que reforça a importância deste trabalho.

Nota-se ainda na Tabela 4.1 que existe um predomínio no uso de redes convolucionais como arquitetura do modelo de classificação em BCIs (62%) e que 85% dos trabalhos utilizam técnicas de pré-processamento para remoção de artefatos com o objetivo de melhorar a relação sinal-ruído dos EEGs e, por consequência, melhorar a qualidade dos dados fornecidos como entrada ao classificador. Com relação ao formato dos dados, 36% dos trabalhos utilizam diretamente sinais brutos de EEG como entrada do classificador, 49% utilizam vetores de atributos extraídos dos EEGs, como atributos espaço-temporais, densidade espectral de potência, filtros espaciais, entropia, entre outros, e 15% utilizam espectrogramas para serem processados como imagens.

Tabela 4.2 – Quadro resumo da distribuição dos 53 trabalhos compilados em (LASHGARI *et al.*, 2020) em função do tipo de técnica de *data augmentation* utilizada.

Técnica de <i>data augmentation</i>	%
Janelamento (<i>Sliding Window</i>)	24
GANs	21
Amostragem (<i>Sampling</i>)	17
Adição de ruído	17
Outros	11
Recombinação de segmentação	6
Transformada de Fourier	4

A Tabela 4.2 apresenta a distribuição dos 53 trabalhos em função da categoria da técnica de *data augmentation* utilizada para sintetizar os EEGs artificiais. A técnica de janelamento consiste em pegar um EEG completo registrado durante o período de aquisição e dividi-lo, com ou sem sobreposição, em janelas (blocos ou segmentos) menores. Apesar da simplicidade, a técnica de janelamento é bem utilizada nos trabalhos (24%) e já promove um bom ganho de desempenho nos classificadores. A segunda categoria mais utilizada é a de técnicas baseadas em redes adversárias generativas (21%), que buscam aproximar a distribuição de probabilidade responsável por gerar os EEGs e, assim, utilizar uma rede generativa capaz de criar EEGs artificiais. Fechando as principais categorias, temos as técnicas de amostragem e de adição de ruído, que juntas correspondem a 34% dos trabalhos.

Ao final do estudo, embora a ressalva dada pelos autores sobre a quantidade reduzida de trabalhos analisados, conclui-se que o uso de técnicas de *data augmentation* pode aumentar consideravelmente o desempenho de classificadores com modelos de *deep learning* em BCIs baseadas em EEGs, com as técnicas de adição de ruído e janelamento promovendo os maiores ganhos, e a tarefa cognitiva sendo a mais beneficiada pelo uso dos dados gerados artificialmente. Ainda, os autores propõem algumas diretrizes para reportar a metodologia e os resultados, devido à falta de clareza encontrada na exposição de grande parte dos trabalhos. As principais diretrizes apontadas por (LASHGARI *et al.*, 2020) são:

- Descrever claramente o método de *data augmentation* proposto.
- Descrever claramente a base de dados (número de indivíduos participantes, número de experimentos (*trials*) e número de classes).
- Testar o método proposto em uma base de dados pública, permitindo uma melhor comparação das técnicas de *data augmentation*.
- Descrever claramente a arquitetura dos modelos e a escolha dos hiperparâmetros.
- Reportar a acurácia do modelo antes e depois do uso da base de dados aumentada.
- Compartilhar, se possível, os dados e códigos para facilitar a reprodução do trabalho.

4.2 *Data augmentation* em BCIs-SSVEP com EEGs

Apresentada a visão geral de como se encontra a literatura voltada para a síntese de EEGs com técnicas de *data augmentation* em sistemas BCIs, conduzimos nossa revisão bibliográfica em direção aos trabalhos (AZNAN *et al.*, 2019), (AZNAN *et al.*, 2020) e (KWON; IM, 2022), que tratam da síntese de EEGs com modelos generativos de *deep learning* para o contexto específico de BCIs baseadas no paradigma SSVEP, que foi o foco de nosso trabalho. As próximas subseções descrevem com maiores detalhes os três artigos citados e indicam o objetivo de cada trabalho, as bases de dados utilizadas, os modelos generativos empregados, os experimentos realizados e, por fim, os resultados obtidos por cada um.

4.2.1 Trabalho de (AZNAN *et al.*, 2019)

Em (AZNAN *et al.*, 2019) os autores tinham como objetivo demonstrar a capacidade de modelos generativos produzirem EEGs-SSVEP artificiais muito semelhantes aos EEGs reais, além de mostrar como o uso desses EEGs sintéticos poderia melhorar o

desempenho dos classificadores da BCI. Para isso, foram utilizadas duas bases de dados com EEGs de 3 segundos de duração capturadas por eletrodos posicionados na região parietal e occipital (P7, P3, Pz, P4, P8, O1, O2), frontal (Fz) e com referência na posição Az, enquanto os usuários das BCIs visualizavam objetos piscando nas frequências de 10, 12 e 15 Hz em um monitor. É importante destacar que embora os autores tenham capturado EEGs de todos os eletrodos citados anteriormente, nos experimentos computacionais são utilizados apenas 2 canais, os quais não são especificados no material publicado.

A primeira base de dados, *video-stimuli dataset*, foi obtida de maneira *offline* e é composta por EEGs-SSVEP do usuário S01 capturadas ao visualizar no monitor os objetos piscando de um vídeo pré-gravado. Essa base é composta por 50 amostras de cada frequência de evocação (10, 12 e 15 Hz) e seus dados foram utilizados exclusivamente para o treinamento dos modelos generativos. A segunda base de dados, *NAO dataset*, é formada também por uma base *offline*, com EEGs obtidos da mesma maneira que na base anterior para os usuários S01, S02 e S03, e por uma base *online* constituída por EEGs dos mesmos usuários enquanto tele-operavam um robô em tempo real que apresentava no monitor objetos piscando nas frequências de evocação 10, 12 e 15 Hz. A parte *offline* da base *NAO dataset* contém 50 amostras de cada frequência de evocação para cada usuário, enquanto a parte *online* da base contém 30 amostras de cada frequência de evocação para cada um.

Para sintetizar os EEGs-SSVEP artificiais, foram empregadas três arquiteturas de modelos generativos de propósito geral, DCGAN, WGAN e VAE, cujas estruturas estão apresentadas na Tabela 4.3. Para cada arquitetura, foi desenvolvido um modelo generativo separado para cada uma das frequências de evocação, resultando em um total de nove modelos generativos: DCGAN - 10 Hz, DCGAN - 12 Hz, DCGAN - 15 Hz, WGAN - 10 Hz, WGAN - 12 Hz, WGAN - 15 Hz, VAE - 10 Hz, VAE - 12 Hz e VAE - 15 Hz. Todos os modelos foram treinados exclusivamente com as amostras do usuário S01 do conjunto de dados *video-stimuli dataset*, sendo que cada modelo foi treinado apenas com amostras da frequência de evocação correspondente. Ao final do processo, 500 EEGs-SSVEP artificiais de 3 segundos de duração foram gerados por cada um dos nove modelos generativos para serem utilizados nos cenários investigados.

Tabela 4.3 – Tabela resumo com a estrutura das redes de cada uma das arquiteturas de modelo generativo - DCGAN, WGAN e VAE.

Arquitetura	Rede	Composição
DCGAN	Generativa	- Camada densa de entrada com função de ativação PreLU - 2 camadas convolucionais transpostas 1D com função de ativação PreLU - 1 camada convolucional transposta 1D com função de ativação sigmoide
	Discriminativa	- Camada convolucional 1D de entrada, com <i>batch-normalization</i> , <i>leaky-ReLU</i> (0.2) e <i>max-pooling</i> - Camada densa com <i>leaky-ReLU</i> (0.2) - Camada densa linear de saída
WGAN	Generativa	- Mesma estrutura da rede generativa da DCGAN
	Crítica	- Mesma estrutura da rede discriminativa da DCGAN
VAE	Codificadora	- Camada convolucional 1D de entrada com <i>batch-normalization</i> , função de ativação PreLU e <i>max-pooling</i> - Camada densa linear para o vetor de médias - Camada densa linear para o vetor de variâncias
	Decodificadora	- Cama densa linear de entrada - 2 camadas convolucionais transpostas 1D com função de ativação PreLU - 1 camada convolucional transposta 1D de saída com função de ativação sigmoide

A Figura 4.1 apresenta a estrutura desenvolvida pelos autores em (AZNAN *et al.*, 2018) para os modelos de classificação. Este tipo de modelo processa os EEGs-SSVEP brutos de 3 segundos de duração utilizando um bloco SCU (do inglês, *SSVEP Convolutional Unit*) composto por uma camada convolucional, *batch normalization* e uma operação de *max-pooling*, uma camada densa e, por fim, uma camada *softmax* para fornecer a probabilidade de o EEG de entrada pertencer a cada uma das frequências de evocação do problema.

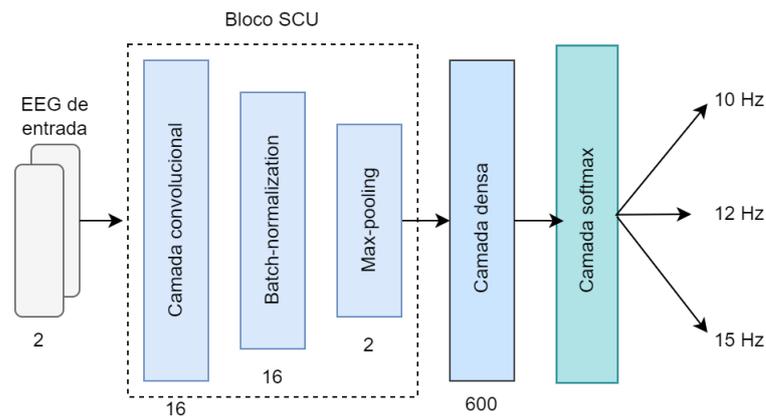


Figura 4.1 – Arquitetura do modelo de classificação com bloco SCU para processamento de EEGs-SSVEP brutos.

O primeiro experimento do estudo demonstrou que os modelos generativos obtidos produzem EEGs-SSVEP artificiais coerentes, com comportamento semelhante aos dados originais. Para isso, foram apresentados os espectros dos EEGs-SSVEP artifi-

ais, similarmente à Figura 2.5, e constatou-se que os EEGs artificiais exibiam picos de magnitude nas mesmas frequências de evocação dos EEGs reais.

O segundo experimento consistia em treinar os modelos de classificação dos usuários S01, S02 e S03 com bases de dados formadas pelos EEGs reais obtidos de forma *offline* da base *NAO dataset* e por amostras artificiais, e comparar os seus desempenhos com os desempenhos dos mesmos classificadores treinados somente com as amostras reais dos usuários. A Figura 4.2 ilustra esquematicamente o treinamento do classificador com o desempenho base de referência (sem o uso de dados sintéticos) e o treinamento do classificador com dados sintéticos de um dos três tipos de modelos generativos. Como resultado, observou-se um aumento no desempenho para os três usuários analisados, com a WGAN sendo a única capaz de gerar EEGs artificiais que proporcionaram melhorias para todos os indivíduos. No entanto, nenhum dos modelos conseguiu oferecer o maior ganho em todos os casos.

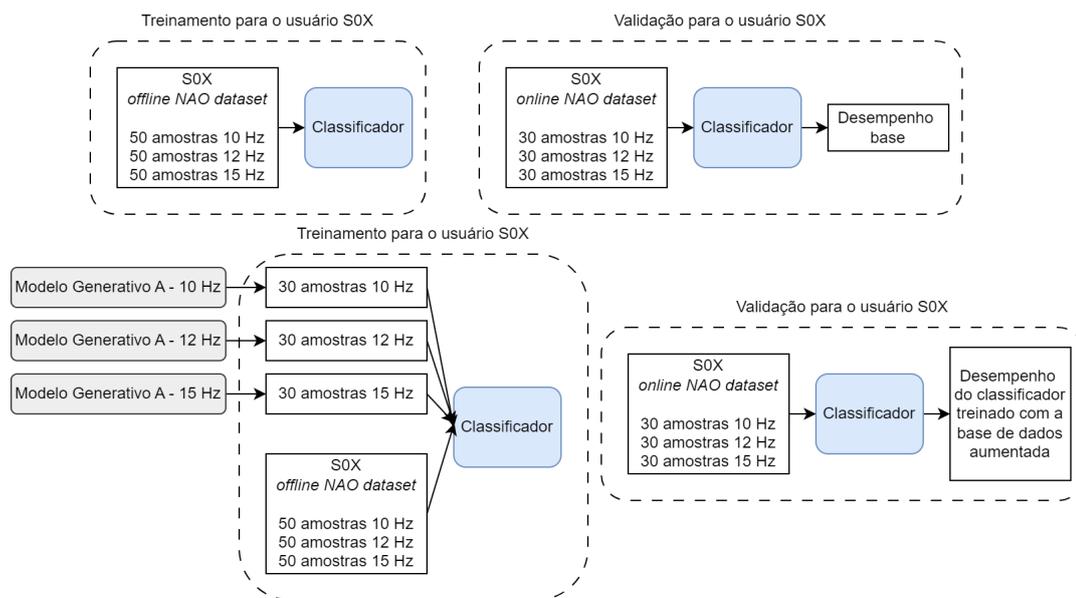


Figura 4.2 – Diagrama esquemático dos treinamentos realizados no segundo experimento do estudo.

Por fim, o terceiro experimento, ilustrado na Figura 4.3, consistia em pré-treinar os modelos de classificação dos usuários S01, S02 e S03 apenas com as 500 amostras artificiais de cada classe geradas pelos modelos generativos, realizar um ajuste fino (*fine-tuning*) dos classificadores pré-treinados com os dados reais dos usuários e comparar os desempenhos desses classificadores com os desempenhos dos mesmos classificadores treinados somente com suas amostras reais. Os resultados deste experimento mostraram que, além do uso de EEGs artificiais no pré-treinamento dos modelos melhorar os desempenhos (em alguns casos superando os resultados do experimento anterior), a convergência no treinamento dos classificadores foi mais rápida, exigindo um menor número de épocas

de treinamento para alcançar os mesmos níveis de desempenho.

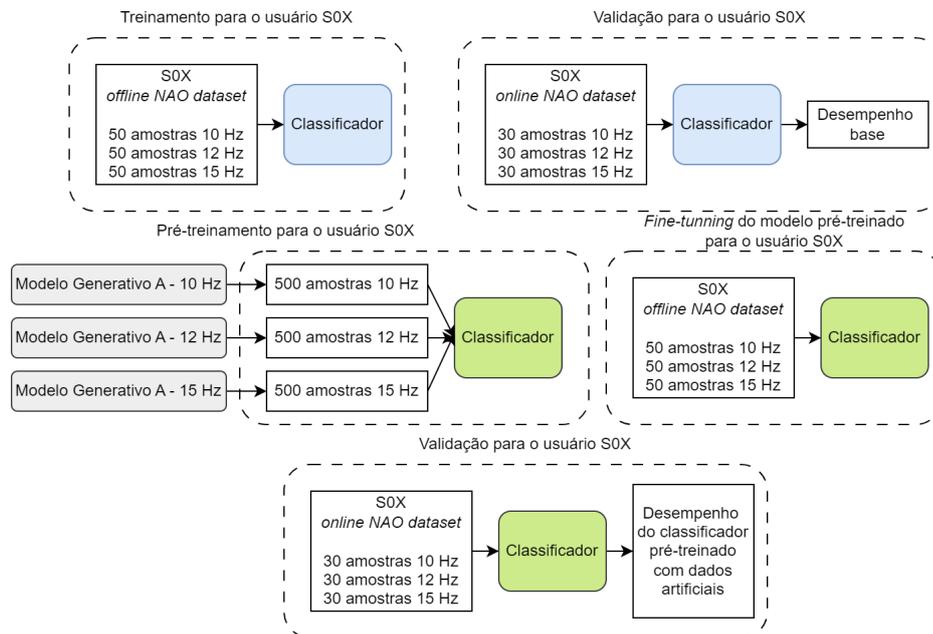


Figura 4.3 – Diagrama esquemático dos treinamentos realizados no terceiro experimento do estudo.

Os autores concluem que o uso de sinais de EEG sintéticos pode ser altamente benéfico para melhorar a precisão, a taxa de convergência e a capacidade de generalização de qualquer modelo treinado para classificar dados de EEG-SSVEP, e as avaliações extensivas demonstraram a eficácia dos dados sintéticos gerados pelos modelos DCGAN, WGAN e VAE em múltiplos experimentos, com a inclusão dos dados gerados sempre melhorando os resultados (AZNAN *et al.*, 2019). No segundo experimento, por exemplo, em que os modelos de classificação foram treinados com bases de dados aumentadas, os novos classificadores obtidos para os usuários S01, S02 e S03 apresentaram ganhos de até 2 pontos percentuais, 3 pontos percentuais e 5 pontos percentuais, respectivamente. Já no terceiro experimento, em que os modelos foram pré-treinados com dados sintéticos e depois treinados com dados reais, os novos classificadores obtidos para os usuários S01, S02 e S03 apresentaram ganhos de até 6 pontos percentuais, 6 pontos percentuais e 3 pontos percentuais, respectivamente.

4.2.2 Artigo de (AZNAN *et al.*, 2020)

Baseando-se nos resultados promissores do trabalho anterior sobre o uso de EEGs-SSVEP sintetizados por modelos generativos, os autores propõem em (AZNAN *et al.*, 2020) uma nova arquitetura de modelo generativo capaz de gerar sinais cerebrais independentes do indivíduo, com o objetivo de facilitar o processo de calibração da BCI

para usuários inexperientes e, em alguns casos, até eliminar a necessidade de calibração do sistema.

Além do comportamento não estacionário que os EEGs podem apresentar para um mesmo usuário, o que torna a manutenção do classificador da BCI projetado para ele uma tarefa recorrente, os EEGs são altamente específicos ao usuário que os gerou. Isso significa que os EEGs de um indivíduo possuem características que podem ser bem distintas dos EEGs de outras pessoas, limitando drasticamente o uso dos EEGs desse indivíduo no treinamento de classificadores para outros usuários. Com efeito, o cenário inter-usuário (*cross-subject*) ainda é bastante desafiador para o projeto de sistemas BCI, dado o risco de se obter um classificador com baixa generalização para novos usuários. Dado esse quadro, a arquitetura proposta e denominada SIS-GAN (do inglês, *SSVEP Invariant Subject - GAN*) busca aprender a sintetizar EEGs-SSVEP invariantes ao indivíduo, isto é, EEGs que apresentam apenas uma parte comum a todos os usuários, sem as particularidades de cada um.

Nesse estudo foram utilizadas duas bases de dados com EEGs de 3 segundos de duração capturados por eletrodos posicionados nas regiões parietal e occipital (P7, P3, Pz, P4, P8, O1, O2), frontal (Fz) e com referência na posição Az, enquanto os usuários das BCIs visualizam objetos piscando nas frequências de 10, 12 e 15 Hz em um monitor com taxa de atualização igual a 60 Hz. A primeira base de dados, *offline dataset*, é formada por 60 amostras de EEG-SSVEP de cada frequência, 10, 12 e 15 Hz, para 9 usuários inexperientes (S0X), capturadas ao visualizarem no monitor objetos piscando nas frequências de estímulo em um vídeo pré-gravado. Já a segunda base de dados, *online dataset*, é constituída por 30 EEGs-SSVEP de cada frequência de 3 usuários experientes (T0X) no uso da BCI e registrados ao visualizarem no monitor objetos piscando nas mesmas frequências de estímulo enquanto tele-operavam um robô em tempo-real.

A Tabela 4.4 apresenta as estruturas das redes generativas, discriminativas e dos classificadores das três arquiteturas investigadas no estudo: DCGAN, AC-GAN e SIS-GAN. Conforme mencionado no trabalho anterior (AZNAN *et al.*, 2019), a DCGAN é uma arquitetura de modelo generativo que permite a geração de dados sintéticos de apenas uma única classe, exigindo o treinamento de uma DCGAN para cada classe do problema. A AC-GAN, detalhada na Seção 3.2.3, resolve essa limitação, permitindo a síntese de dados artificiais de várias classes com um único modelo generativo, além de possibilitar o controle da classe das amostras a serem sintetizadas.

A SIS-GAN proposta é uma extensão da AC-GAN que, além de incluir uma entrada adicional para o controle da classe e uma rede classificadora auxiliar (C_{classe}) responsável por identificar a classe à qual a amostra real ou sintética pertence, possui uma segunda rede classificadora auxiliar já treinada ($C_{usuário}$) cuja função é identificar a

qual indivíduo a mesma amostra real ou sintética pertence, conforme é apresentado na Figura 4.4. O classificador ($C_{usuário}$) é capaz de identificar facilmente o sinal do usuário, já que os EEGs-SSVEP são altamente específicos a cada um.

Tabela 4.4 – Tabela resumo com as estruturas das redes generativas, discriminativas e classificadoras das redes DCGAN, AC-GAN e SIS-GAN.

Arquitetura	Rede	Composição
DCGAN	Generativa	- 5 camadas convolucionais com <i>stride</i> fracionário com: . <i>Batch-Normalization</i> e PreLU
	Discriminativa	- 4 camadas convolucionais 1D com: . <i>Batch-Normalization</i> , PReLU e <i>DropOut</i> - Camada densa linear com 2 neurônios, um para indicar amostra real, outra para indicar amostra falsa
AC-GAN	Generativa	- 5 camadas convolucionais com <i>stride</i> fracionário com: . <i>Batch-Normalization</i> e PreLU
	Camadas intermediárias comuns às redes discriminativa e classificadora	- 4 camadas convolucionais 1D com: . <i>Batch-Normalization</i> , PReLU e <i>DropOut</i>
	Discriminativa	- Camada densa linear com 2 neurônios, um para indicar amostra real, outra para indicar amostra falsa
	Classificadora (Saída: Classe)	- Camada densa linear com 3 neurônios, uma para indicar cada frequência do EEG (10 Hz, 12 Hz e 15 Hz)
SIS-GAN	Generativa	- 5 camadas convolucionais com <i>stride</i> fracionário com: . <i>Batch-Normalization</i> e PreLU
	Camadas intermediárias comuns às redes discriminativa e classificadora	- 4 camadas convolucionais 1D com: . <i>Batch-Normalization</i> , PReLU e <i>DropOut</i>
	Discriminativa	- Camada densa linear com 2 neurônios, um para indicar amostra real, outra para indicar amostra falsa
	Classificadora (Saída: Classe)	- Camada densa linear com 3 neurônios, uma para indicar cada frequência do EEG (10 Hz, 12 Hz e 15 Hz)
	Classificadora (Saída: Usuário)	- Camada densa linear com 9 neurônios, cada uma indicando um dos indivíduos da base de dados <i>offline</i>

O treinamento dessa rede envolve três etapas: (i) minimização do erro associado à rede discriminativa, com o objetivo de que a rede generativa seja capaz de gerar EEGs artificiais muito semelhantes aos EEGs reais dos usuários; (ii) minimização do erro associado à rede classificadora C_{classe} , cujo objetivo é identificar corretamente a frequência de evocação do EEG real ou sintético e, em última instância, fazer com que a rede generativa seja capaz de gerar os EEGs de uma classe específica; e (iii) maximização do erro associado à rede classificadora $C_{usuário}$, pois, como o objetivo é gerar EEGs artificiais invariantes ao usuário, um pior desempenho dessa rede indica que os EEGs artificiais sintetizados pela rede generativa estão se tornando cada vez mais genéricos, sem as características particulares de cada usuário. Durante o treinamento apenas as redes generativa, discriminativa e classificadora C_{classe} têm seus parâmetros ajustados a cada iteração, enquanto o classificador auxiliar $C_{usuário}$ tem seus parâmetros fixos.

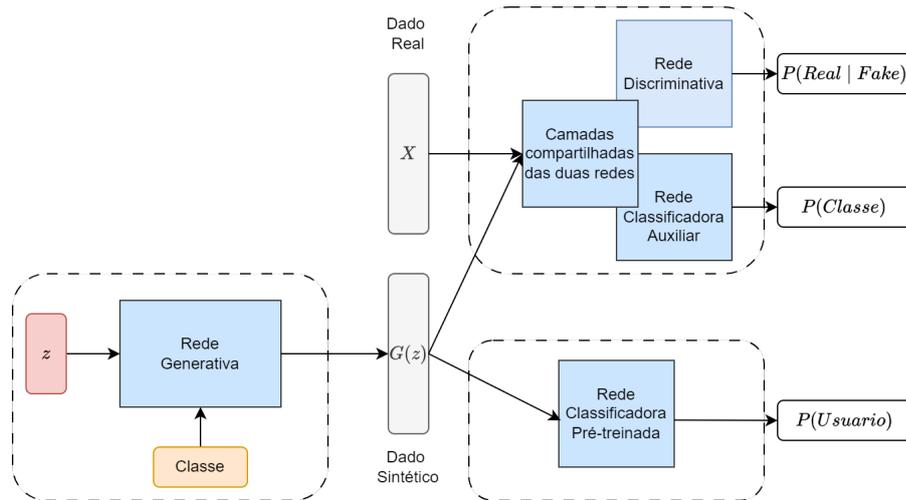


Figura 4.4 – Diagrama do modelo proposto SIS-GAN.

No primeiro experimento, os autores realizaram o procedimento apresentado na Figura 4.5 separadamente para cada um dos usuários, onde os modelos generativos foram treinados apenas com os dados do usuário para o qual o classificador da BCI estava sendo desenvolvido. Por exemplo, para o usuário S01, três classificadores foram obtidos e seus desempenhos comparados com um conjunto de validação formado por 20% de seus EEGs da base de dados *offline*. O primeiro classificador, $C_{S01-baseline}$, foi treinado com os 80% dos EEGs da base de dados, o que corresponde a 48 amostras por classe. Em seguida, o mesmo subconjunto de EEGs foi utilizado para treinar 3 DCGANs, uma para cada frequência de evocação. Após o treinamento, foram sintetizados 48 EEGs artificiais de cada frequência, que foram incorporados ao subconjunto de EEGs reais no treinamento do segundo classificador, $C_{S01-Aug.Dataset-DCGAN}$. Por último, o mesmo subconjunto de EEGs reais foi utilizado para treinar uma única AC-GAN. Finalizado o treinamento, foram gerados 48 EEGs por classe, que também foram adicionados ao subconjunto de 80% de EEGs reais para o treinamento do terceiro classificador, $C_{S01-Aug.Dataset-AC-GAN}$. Os resultados mostram que o uso das amostras geradas pela AC-GAN também promove ganhos de desempenho quando comparado com os casos em que os classificadores foram treinados apenas com amostras reais. Além disso, os ganhos obtidos com o uso dos EEGs artificiais desse modelo são superiores ao do uso dos EEGs sintetizados pelas 3 DCGANs.

No segundo experimento, os autores realizaram o procedimento da Figura 4.6, em que o processo de desenvolvimento da BCI para o usuário S0X utilizou apenas EEGs dos outros 8 indivíduos da base de dados *offline*. Por exemplo, para o usuário S01, cinco classificadores foram obtidos e seus desempenhos comparados com um conjunto de validação formado apenas pelos seus EEGs da base de dados *offline*. O primeiro classificador, $C_{S01-baseline}$, foi treinado apenas com as amostras reais dos 8 indivíduos S02 a S09. Em seguida, os EEGs reais desses indivíduos foram utilizados separadamente para treinar

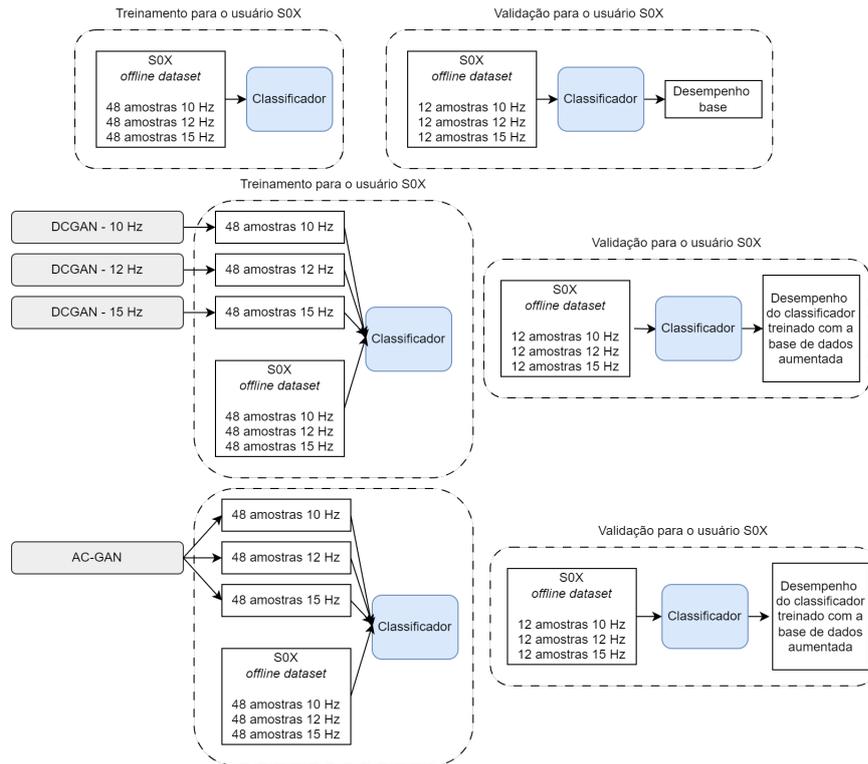


Figura 4.5 – Diagrama esquemático dos treinamentos realizados no primeiro experimento do estudo.

8 AC-GANs, uma para cada pessoa. Após o treinamento dos modelos, 60 EEGs artificiais de cada frequência foram criados por cada modelo e utilizados de duas maneiras: (i) incorporando os EEGs artificiais aos EEGs reais dos usuários S02 a S09 no treinamento do classificador $C_{S01-Aug.Dataset-ACGAN}$ e (ii) utilizando apenas EEGs artificiais sintetizados no treinamento do classificador $C_{S01-Synt.Dataset-ACGAN}$. Por fim, os EEGs reais dos mesmos indivíduos foram utilizados no treinamento de uma SIS-GAN, capaz de sintetizar EEGs artificiais invariantes ao sujeito para todas as frequências de evocação. Assim como na AC-GAN, os EEGs artificiais produzidos pela SIS-GAN foram utilizados de duas formas, com a primeira incorporando os EEGs sintetizados aos EEGs reais dos usuários S02 a S09 no treinamento do classificador $C_{S01-Aug.Dataset-SISGAN}$, e a segunda utilizando apenas os EEGs invariantes ao sujeito no treinamento do classificador $C_{S01-Synt.Dataset-SISGAN}$.

Como esperado, os desempenhos-base dos classificadores do segundo experimento foram inferiores aos desempenhos-base do primeiro experimento. Isso ocorreu porque no primeiro experimento os classificadores foram treinados com os EEGs reais do usuário para o qual a BCI estava sendo desenvolvida, enquanto no segundo experimento, os classificadores foram treinados com os EEGs de todos os usuários, exceto os daquele para o qual a BCI estava sendo implementada. Ao comparar os desempenhos-base dos classificadores do segundo experimento com suas versões obtidas com o uso de base de

dados aumentada, observou-se ganhos de desempenho para alguns usuários e degradação para outros. Por fim, constatou-se que os classificadores treinados apenas com os dados artificiais sintetizados pela SIS-GAN promoveram ganhos de desempenho para todos os usuários analisados, o que mostra a viabilidade desse modelo no objetivo principal do estudo, de facilitar a etapa de calibração da BCI para novos usuários.

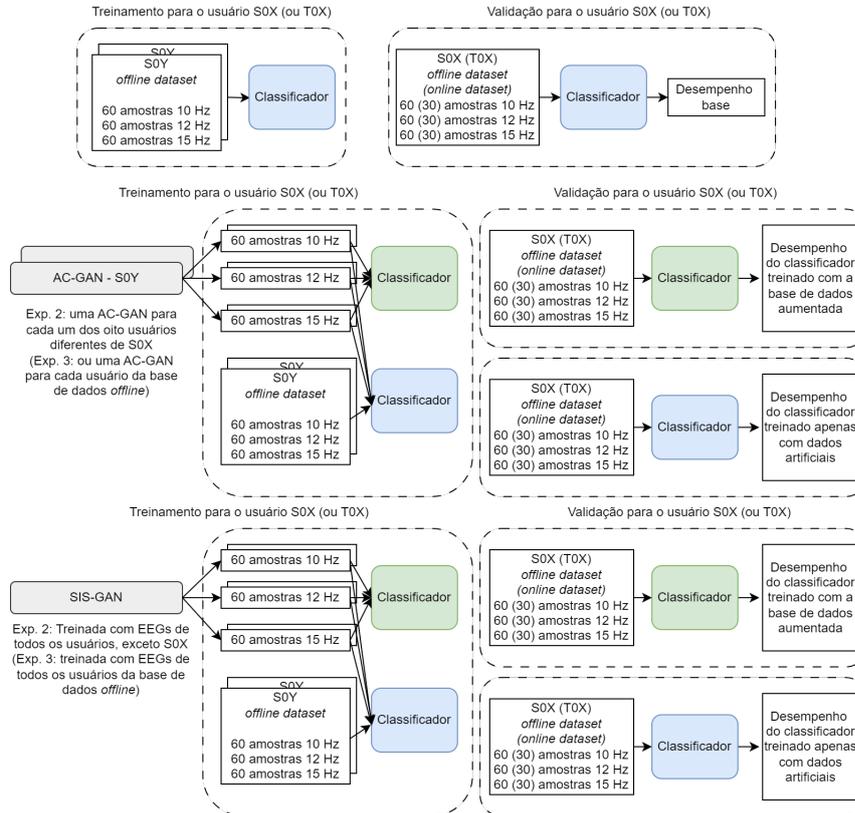


Figura 4.6 – Diagrama esquemático dos treinamentos realizados no segundo e terceiro experimento do estudo. No caso do terceiro experimento, os EEGs de todos os usuários S0X foram utilizados para obter os classificadores, enquanto os usuários T0X foram utilizados para validação.

No terceiro experimento, os autores realizaram o mesmo procedimento da Figura 4.6, mas agora utilizando duas tarefas (e bases) distintas, a *offline* e a *online*, em que o processo de desenvolvimento da BCI para o usuário T0X utilizou apenas EEGs dos 9 indivíduos da base dados *offline*. Para o usuário T0X cinco classificadores foram obtidos e seus desempenhos comparados com um conjunto de validação formado apenas pelos seus EEGs da base de dados *online*. O primeiro classificador, $C_{T0X-baseline}$, foi treinado apenas com as amostras reais dos indivíduos S01 a S09. Em seguida, os EEGs reais desses usuários foram utilizados separadamente para treinar 9 AC-GANs, uma para cada pessoa. Após o treinamento dos modelos, 60 EEGs artificiais de cada frequência foram gerados por cada modelo e utilizados de duas maneiras: (i) incorporando os EEGs artificiais aos EEGs reais dos usuários S01 a S09 no treinamento do classificador

$C_{T0X-Aug.Dataset-ACGAN}$ e (ii) utilizando apenas EEGs artificiais sintetizados no treinamento do classificador $C_{T0X-Synt.Dataset-ACGAN}$. Finalmente, os EEGs reais dos mesmos indivíduos foram utilizados no treinamento de uma SIS-GAN e, assim como feito para a AC-GAN, os EEGs artificiais produzidos pela SIS-GAN foram utilizados de duas formas, com a primeira incorporando os EEGs sintetizados aos EEGs reais dos usuários S01 a S09 no treinamento do classificador $C_{T0X-Aug.Dataset-SISGAN}$ e a segunda utilizando apenas os EEGs invariantes ao sujeito no treinamento do classificador $C_{T0X-Synt.Dataset-SISGAN}$. Os resultados obtidos no segundo experimento se estendem igualmente para este.

Os autores concluem que os resultados experimentais demonstram a eficácia dos dados gerados sinteticamente para aumentar o desempenho de um modelo de classificação de EEGs-SSVEP, melhorando a generalização para sujeitos não vistos ao realizar a classificação sem calibração. No segundo experimento, por exemplo, onde foi utilizado apenas as amostras do *offline dataset*, o desempenho médio ao longo dos 9 usuários investigados saiu do *baseline* de 68,8% para um desempenho médio de 72,5% quando utilizados apenas os EEGs artificiais sintetizados pela SIS-GAN. Além disso, no terceiro experimento, em que a validação foi feita com o uso das amostras da *online dataset*, o desempenho médio ao longo dos 3 usuários inexperientes saiu do *baseline* de 49,5% para um desempenho médio de 66% quando utilizados apenas os EEGs artificiais sintetizados pela SIS-GAN.

4.2.3 Artigo de (KWON; IM, 2022)

O problema da falta de dados em BCIs-SSVEP com EEGs está fortemente relacionado às dificuldades presentes no procedimento experimental para aquisição dos sinais cerebrais, especialmente devido ao processo de calibração da BCI. Em (KWON; IM, 2022), os autores apresentam uma proposta promissora de arquitetura de modelo generativo capaz de produzir EEGs-SSVEP a partir de EEGs em repouso do usuário, os quais podem ser utilizados no treinamento do classificador da BCI. As principais vantagens dessa abordagem são a redução no tempo de calibração da BCI e também a maior facilidade na aquisição de dados, já que a coleta dos EEGs em repouso é menos complexa e mais confortável para o usuário.

O modelo proposto, *Signal-to-signal StarGAN* (S2S-StarGAN), para geração de EEGs-SSVEP é baseado no conhecido modelo StarGAN-v2, originalmente desenvolvido para problemas de tradução de imagens de múltiplos domínios, que permite a geração de imagens que preservam o conteúdo original da imagem de entrada enquanto adotam estilos de diferentes domínios. O *Signal-to-signal StarGAN* é adaptado para preservar as particularidades dos EEGs em repouso do usuário alvo e incorporar o estilo associado aos EEGs-SSVEP, que foi aprendido durante o treinamento do modelo, para produzir os

EEGs-SSVEP desse usuário.

Neste estudo, os EEGs foram capturados utilizando os eletrodos Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T4, T5, P3, Pz, P4, T6, O1, Oz e O2. No entanto, apenas os três últimos, O1, Oz e O2, posicionados na região occipital, foram utilizados nos experimentos. Cada registro de EEG possui 6 segundos de duração, com os 2 segundos iniciais correspondendo à etapa de instrução e os 4 segundos restantes correspondendo ao EEG-SSVEP associado à tarefa de visualização do estímulo. As frequências de evocação empregadas foram 6 Hz, 6.67 Hz, 7.5 Hz e 10 Hz, e cada um dos 18 participantes foi submetido a 10 sessões experimentais, com cada sessão retornando 5 *trials* para cada frequência de estímulo. Além disso, os EEGs em repouso dos participantes foram capturados pelos eletrodos durante 16 segundos, iniciando 20 segundos antes da primeira *trial*, conforme mostra a Figura 4.7.

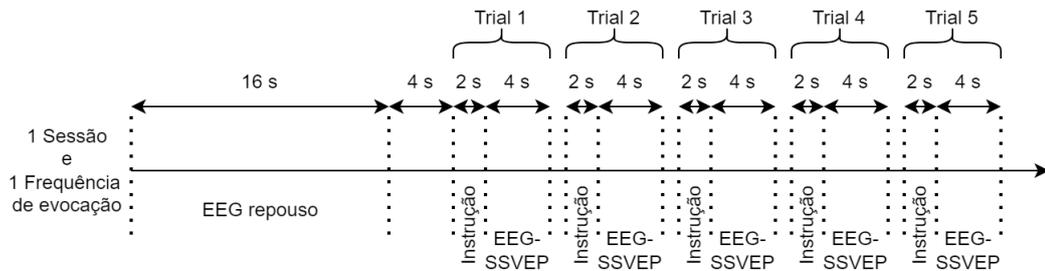


Figura 4.7 – Protocolo experimental adotado em (KWON; IM, 2022) para coleta dos EEGs de repouso e os EEGs-SSVEP para uma frequência de evocação durante uma sessão.

A Figura 4.8 apresenta as quatro redes neurais que constituem a arquitetura S2S-StarGAN. Assim como outras GANs, esse modelo possui uma rede geradora (G) e uma rede discriminadora (D), responsáveis, respectivamente, por gerar os EEGs-SSVEP artificiais e identificar se o dado fornecido é real ou sintético. No entanto, diferentemente das redes generativas tradicionais que produzem o dado artificial a partir de um código latente amostrado de alguma distribuição de probabilidade, o modelo S2S-StarGAN recebe como entradas o EEG em repouso do usuário e um código adicional que estabelece o estilo desejado, permitindo que o modelo produza o EEG-SSVEP artificial de interesse. Por sua vez, o código de estilo pode ser originado de duas formas: (i) a partir da rede codificadora de estilos (E) que extrai e codifica o estilo associado aos EEGs-SSVEP durante o processo de treinamento e (ii) a partir da rede de mapeamento F que aprende a sintetizar códigos de estilo a partir de uma distribuição aleatória. Nessa proposta, além do EEG em repouso, cada frequência de evocação representa um domínio do EEG.

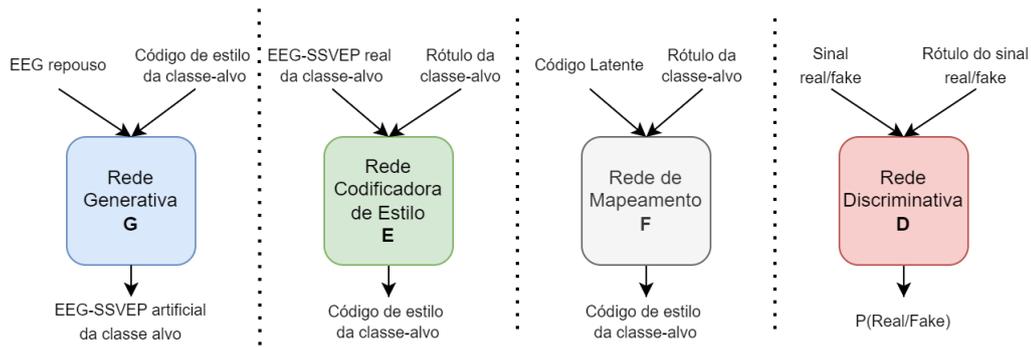


Figura 4.8 – Diagrama esquemático das redes que compõem a arquitetura S2S-StarGAN.

O treinamento da S2S-StarGAN envolve quatro funções de perda: (i) função de perda adversarial L_{ADV} vinculada ao treinamento adversário da GAN que incentiva a rede generativa a produzir sinais realistas que a rede discriminativa não consegue diferenciar dos EEGs-SSVEP reais; (ii) função de perda de reconstrução de estilo L_{STY} que busca assegurar que o código de estilo extraído da imagem sintetizada seja semelhante ao código de estilo original; (iii) a função de perda de consistência de ciclo L_{CYC} que verifica se o EEG-SSVEP sintetizado pode ser revertido ao seu estado original (EEG em repouso) após múltiplas transformações de estilo; e (iv) função de perda sensível à diversidade L_{DS} que estimula a rede generativa a produzir EEGs-SSVEP diferentes a partir de códigos de estilos diferentes. Após finalizado o treinamento, a rede generativa obtida pode ser operada de duas maneiras, como indicado na Figura 4.9.

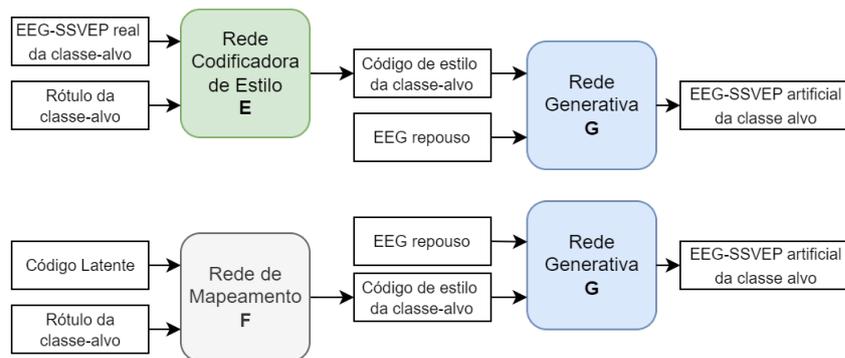


Figura 4.9 – Diagrama esquemático das duas formas de operação do modelo generativo da S2S-StarGAN após sua obtenção.

Os resultados dos experimentos demonstraram que o uso dos EEGs-SSVEP sintetizados pelo S2S-StarGAN melhorou o desempenho dos classificadores investigados em ambas as formas de operação. Diferentemente de outras arquiteturas de modelos generativos propostas até então, a S2S-StarGAN mostrou-se capaz de produzir EEGs-SSVEP artificiais sem a necessidade de utilizar os EEGs-SSVEP reais do usuário, permitindo acelerar ou até eliminar o processo de calibração da BCI para ele.

4.3 Proposta de Trabalho

Na Seção 2.3 vimos que, nas interfaces cérebro-computador baseadas no paradigma SSVEP, os usuários apresentam atividades cerebrais de interesse que são desencadeadas por estímulos visuais que oscilam em frequências específicas, denominadas frequências de evocação. As atividades cerebrais associadas a esses estímulos exibem componentes espectrais nas frequências múltiplas e sub-múltiplas das frequências de estímulo, o que torna o problema bem caracterizado no domínio da frequência.

Neste capítulo, revisamos os estudos da literatura que abordam a síntese de EEGs-SSVEP utilizando modelos generativos, e vimos como os dados artificiais produzidos por esses modelos podem ser úteis no treinamento dos classificadores das BCIs, proporcionando ganhos de desempenho. Contudo, todos os modelos generativos mencionados nesses estudos foram desenvolvidos para sintetizar EEGs-SSVEP exclusivamente no domínio do tempo, enquanto o domínio da frequência foi empregado apenas para validar visualmente a qualidade dos dados artificiais sintetizados por meio da comparação dos espectros dos EEGs reais com os espectros dos EEGs artificiais.

Diante do cenário apresentado a proposta deste trabalho é investigar como diferentes arquiteturas de modelos generativos podem ser utilizadas para a síntese de EEGs-SSVEP diretamente no domínio da frequência, que parece ser o domínio mais adequado devido à forma com que a atividade cerebral de interesse é induzida, e como o uso dos EEGs-SSVEP artificiais no domínio da frequência pode impactar o treinamento dos classificadores das BCIs.

Neste trabalho, investigamos algumas arquiteturas bem consolidadas de modelos generativos de propósito geral para a síntese de EEGs-SSVEP, incluindo DCGAN, VAE, WGAN e AC-GAN, conforme utilizado nos trabalhos (AZNAN *et al.*, 2019) e (AZNAN *et al.*, 2020), suas variantes condicionadas cDCGAN, cVAE e cWGAN, e também as duas versões de *autoencoder* adversarial, AAE e cAAE, ainda não empregadas no contexto de BCIs-SSVEP. Para classificação, utilizamos três tipos de classificadores: SVM linear, SVM não-linear e CNN-1D, com o objetivo de verificar como o uso dos EEGs-SSVEP artificiais impacta tanto classificadores tradicionais, como as SVMs, quanto arquiteturas mais complexas, como a CNN-1D.

Por fim, para comparar o uso dos EEGs-SSVEP no domínio da frequência ao uso de EEGs-SSVEP gerados no domínio do tempo, investigamos três cenários de síntese e uso dos EEGs: (i) sinais sintetizados no domínio do tempo e utilizados no treinamento dos classificadores no domínio do tempo, (ii) EEGs sintetizados no domínio do tempo e utilizados no treinamento dos classificadores no domínio da frequência, e (iii) EEGs sintetizados diretamente no domínio da frequência e empregados no treinamento

dos classificadores também no domínio da frequência. Desta forma, este trabalho busca contribuir para o estudo de *data augmentation* em sistemas BCI-SSVEP, considerando em particular a abordagem de geração de sinais artificiais diretamente no domínio da frequência.

5 Metodologia

Neste capítulo, apresentamos de forma detalhada a base de dados utilizada nos experimentos computacionais, descrevendo como as técnicas de pré-processamento foram aplicadas aos EEGs brutos, e as escolhas realizadas para a separação dos dados em conjuntos de treinamento e validação, essenciais para garantir a robustez e a validade dos resultados obtidos pelos classificadores da BCI. Além disso, discutimos as estruturas e os hiperparâmetros dos modelos de classificação empregados, incluindo a SVM linear, SVM não-linear e CNN-1D, bem como dos modelos generativos DCGAN, WGAN, AAE, VAE e AC-GAN. Por fim, descrevemos detalhadamente todos os passos executados durante as simulações computacionais, assegurando a reprodutibilidade e a clareza da metodologia aplicada.

5.1 Base de dados

Neste trabalho exploramos uma base de dados composta por EEGs-SSVEP de 32 participantes coletados pelo nosso grupo de pesquisa (CAAE0617.0.146.000-1). Como introduzido na Seção 2.3, cada participante foi submetido ao procedimento para aquisição dos EEGs, sendo que uma sessão do procedimento consiste no participante observar por 12 segundos cada um dos quatro padrões xadrez oscilando nas frequências de evocação 6 Hz, 10 Hz, 12 Hz e 15 Hz apresentados na tela de um monitor, como exemplificado na Figura 2.4, enquanto a atividade cerebral resultante dos estímulos visuais SSVEP é capturada a uma taxa de 256 Hz por um conjunto de 16 eletrodos posicionados segundo o padrão 10-10 nas regiões dos lobos occipital (O1, O2, Oz), parietal (POz, Pz, PO4, PO3, PO8, PO7, P2, P1), sulco central (Cz, C1, C2), e nos córtex motor (CPz) e sensoriomotor (FCz). Cada participante foi submetido a 8 sessões (*trials*) do procedimento experimental, resultando em um total de 32 EEGs de 12 segundos, sendo 8 EEGs para cada uma das frequências de evocação.

Para o treinamento e validação dos modelos de classificação e dos modelos generativos nos experimentos computacionais, optamos por utilizar apenas os EEGs correspondentes às frequências de evocação de 12 Hz e 15 Hz, capturadas exclusivamente pelo eletrodo Oz de 10 dos 32 participantes da base de dados (S01 a S10). A escolha por duas frequências simplifica o problema de classificação, transformando-o em um problema binário, e a escolha por apenas 10 usuários da base foi motivada por questões computacionais, pois a grande variedade de configurações levou a um número elevado de modelos de classificação e modelos generativos a serem treinados. O uso exclusivo dos EEGs do

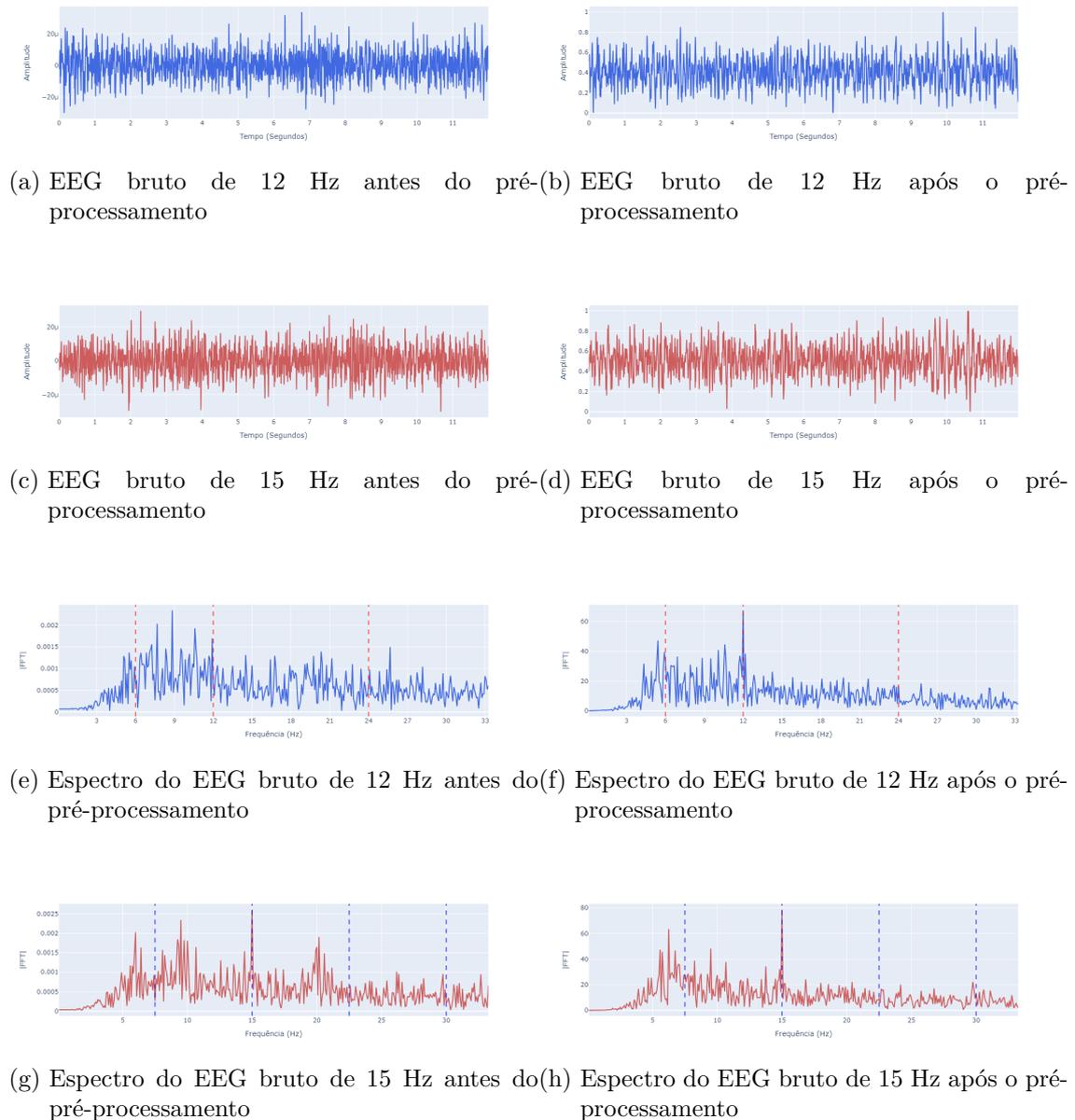


Figura 5.1 – Exemplos de EEGs no domínio do tempo e no domínio da frequência para as frequências de evocação 12 Hz e 15 Hz capturadas pelo eletrodo Oz do usuário S01. As imagens da coluna à esquerda apresentam os EEGs brutos, sem qualquer pré-processamento digital, e as imagens da coluna à direita apresentam os mesmos EEGs após o pré-processamento.

eletrodo Oz, por sua vez, permite que os modelos trabalhem com apenas um canal de entrada, simplificando a tarefa de extração de informação, embora também possa levar a uma perda por desconsiderar os outros canais. Além disso, a escolha pelo eletrodo Oz foi por ele estar localizado no centro da região occipital, apresentando uma maior chance de registrar a atividade cerebral correspondente aos estímulos visuais SSVEP observados. Contudo, é importante ressaltar que os EEGs capturados pelos demais eletrodos são descartados somente após a aplicação do filtro CAR, para a remoção dos artefatos

que aparecem igualmente em todos os eletrodos durante o registro dos EEGs, conforme explicado na Seção 2.4.1.

A Figura 5.1 apresenta dois exemplos de EEGs, um para a frequência de 12 Hz e outro para a frequência de 15 Hz, registrados com o eletrodo Oz para o participante S01, tanto no domínio do tempo quanto no domínio da frequência. Na coluna à esquerda temos os EEGs brutos sem qualquer pré-processamento digital e podemos notar que os sinais apresentados possuem bastante ruído, principalmente quando observamos os espectros desses EEGs nas Figuras 5.1(e) e 5.1(g), que não apresentam com tanta nitidez as componentes espectrais associadas aos estímulos visuais SSVEP. Já na coluna à direita temos os mesmos EEGs após a aplicação do filtro CAR para a remoção de artefatos. É possível observar que as componentes espectrais das frequências de evocação são bem mais nítidas (Figuras 5.1(f) e 5.1(h)), além de que no domínio do tempo conseguimos verificar de forma mais clara uma certa periodicidade dos sinais apresentados.

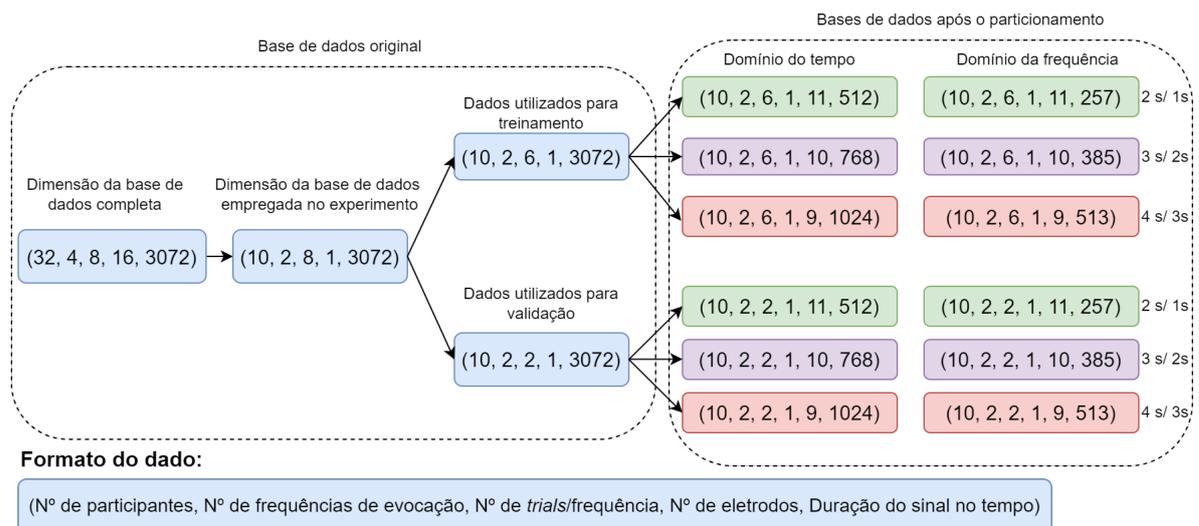
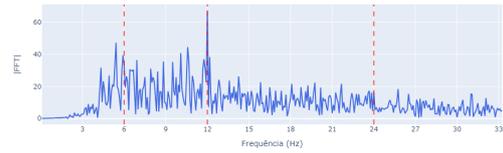
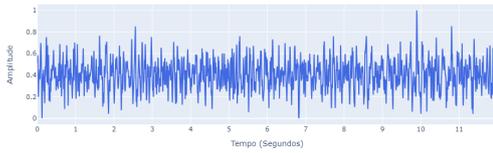


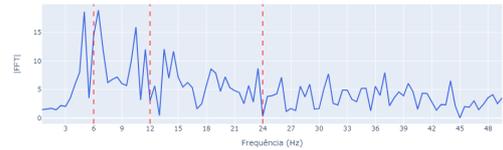
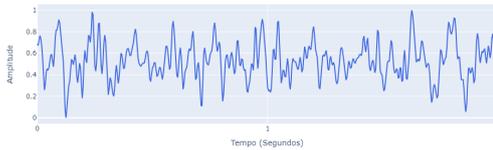
Figura 5.2 – Diagrama com as dimensões dos conjuntos de dados, desde a base de dados completa até os conjuntos de treinamento e de validação após cada configuração de particionamento.

A base de dados resultante das escolhas feitas anteriormente foi dividida em dois conjuntos: o de treinamento, constituído pelos EEGs das sessões 1, 2, 3, 4, 5 e 6, e o de validação, composto pelos EEGs das sessões 7 e 8. Como o número de EEGs por participante é extremamente reduzido, com um total de 6 amostras por frequência no conjunto de treinamento e 2 amostras por frequência no conjunto de validação, e o EEG de 12 segundos é bastante longo para ser utilizado em aplicações reais, decidimos aplicar a técnica de janelamento ou particionamento aos EEGs e trabalhar em cima dos EEGs particionados e com duração reduzida. Nesta etapa, exploramos 3 configurações distintas de particionamento com sobreposição dos EEGs existentes, aplicando janelas iguais a 2 segundos, 3 segundos e 4 segundos, e que utilizavam um passo de 1 segundo para percorrer

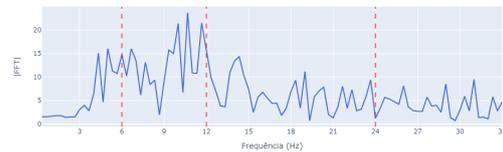
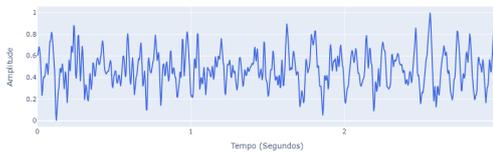
os EEGs completos. O resultado do particionamento com sobreposição dos EEGs e as dimensões dos conjuntos de dados resultantes são apresentadas na Figura 5.2, enquanto as Figuras 5.3 e 5.4 apresentam exemplos de EEGs particionados e seus espectros no domínio da frequência para as frequências de evocação 12 Hz e 15 Hz, respectivamente.



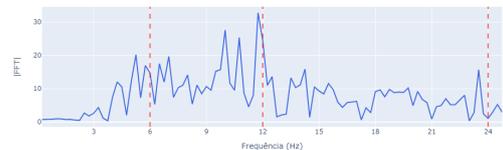
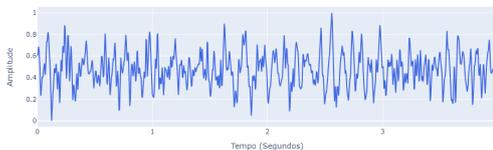
(a) EEG bruto de 12 Hz com 12 s de duração após o pré-processamento (b) Espectro do EEG bruto de 12 Hz com 12 s de duração após o pré-processamento



(c) Segmento de 2 s do EEG bruto de 12 Hz após o pré-processamento (d) Espectro do segmento de 2 s do EEG bruto de 12 Hz após o pré-processamento



(e) Segmento de 3 s do EEG bruto de 12 Hz após o pré-processamento (f) Espectro do segmento de 3 s do EEG bruto de 12 Hz após o pré-processamento



(g) Segmento de 4 s do EEG bruto de 12 Hz após o pré-processamento (h) Espectro do segmento de 4 s do EEG bruto de 12 Hz após o pré-processamento

Figura 5.3 – EEG no domínio do tempo e no domínio da frequência para a frequência de evocação de 12 Hz capturada pelo eletrodo Oz do usuário S01. As figuras (a) e (b) apresentam o EEG bruto de 12 s após o pré-processamento, as figuras (c) e (d) apresentam um exemplo de segmento de 2 s do mesmo EEG, as figuras (e) e (f) apresentam um exemplo de segmento de 3 s do mesmo EEG e as figuras (g) e (h) apresentam um exemplo de segmento de 4 s do mesmo EEG.

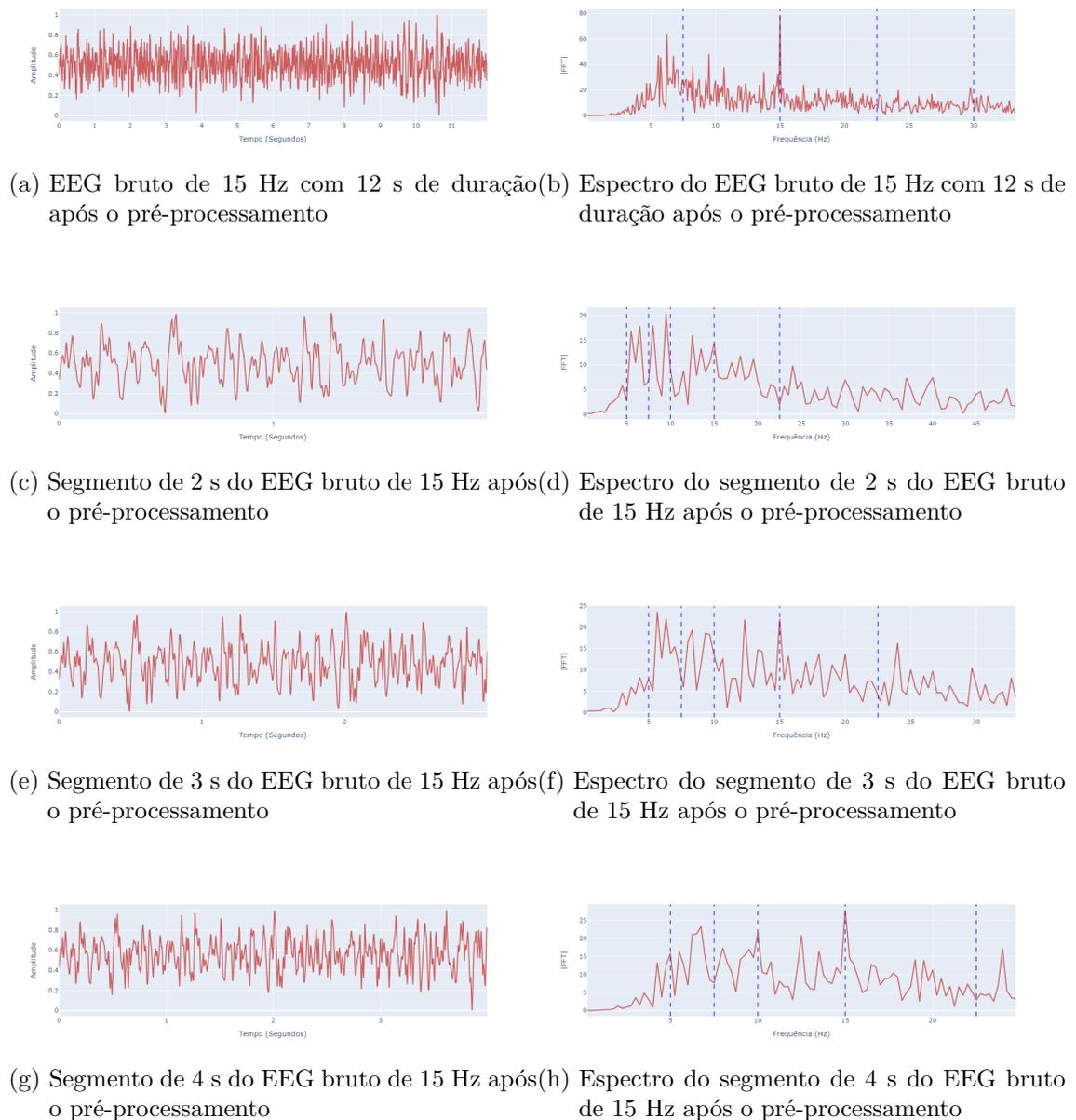


Figura 5.4 – EEG no domínio do tempo e no domínio da frequência para a frequência de evocação de 15 Hz capturada pelo eletrodo Oz do usuário S01. As figuras (a) e (b) apresentam o EEG bruto de 12 s após o pré-processamento, as figuras (c) e (d) apresentam um exemplo de segmento de 2 s do mesmo EEG, as figuras (e) e (f) apresentam um exemplo de segmento de 3 s do mesmo EEG e as figuras (g) e (h) apresentam um exemplo de segmento de 4 s do mesmo EEG.

É interessante notar nas Figuras 5.3 e 5.4 que, quanto menor o comprimento do segmento de EEG, menos nítido se torna o conteúdo espectral relacionado à frequência de evocação. Portanto, embora o número de amostras aumente quando o EEG é particionado em segmentos de 2 segundos, a qualidade desses segmentos tende a diminuir, o que pode impactar negativamente o desempenho dos classificadores durante o treinamento. Por

outro lado, quanto maior o comprimento do segmento de EEG, mais nítido se torna o conteúdo espectral relacionado à frequência de evocação. Neste trabalho faremos uma análise detalhada com relação a esse comportamento para verificar o impacto do uso de dados sintéticos com EEGs de diferentes comprimentos.

5.2 Estruturas e hiperparâmetros dos modelos de classificação

Após o detalhamento da base de dados de EEGs na Seção 5.1, apresentamos nesta seção as escolhas e a metodologia empregada no treinamento dos modelos de classificação baseados em SVM linear, SVM não-linear e rede convolucional 1D (CNN-1D) para os usuários de BCIs investigados neste estudo. Os modelos de classificação tratados são específicos para cada indivíduo, ou seja, cada classificador é treinado com amostras de um usuário específico e utilizado para classificar os EEGs desse mesmo usuário, tanto no domínio do tempo quanto no domínio da frequência e para ambos os cenários: sem e com *data augmentation*.

Um ponto importante que precisamos deixar claro é que o uso dos EEGs nos comprimentos e domínios investigados no estudo é feito de forma individualizada, ou seja, um classificador é treinado para processar EEGs de duração e domínio específicos. Por exemplo, considerando apenas os EEGs reais do usuário S01, teremos um total de 18 classificadores para esse participante, sendo um SVM linear, um SVM não-linear e uma CNN-1D para processar EEGs de: (i) 2 segundos no domínio do tempo, (ii) 3 segundos no domínio do tempo, (iii) 4 segundos no domínio do tempo, (iv) 2 segundos no domínio da frequência, (v) 3 segundos no domínio da frequência e (vi) 4 segundos no domínio da frequência.

Tabela 5.1 – Conjunto de hiperparâmetros utilizados durante o processo de treinamento e obtenção dos modelos de classificação baseados em SVM Linear e SVM Não-Linear.

Função <i>Kernel</i>	Hiperparâmetros	Conjunto de valores investigados
Linear	Regularização C	$[10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3]$
Polinomial	Regularização C	$[10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3]$
	Grau do polinômio d	$[2, 3, 4, 5, 6]$
RBF	γ	$1/(n_{atributos_X} \times var(X))$
	Regularização C	$[10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3]$
	γ	$1/(n_{atributos_X} \times var(X))$

Os classificadores baseados em SVM linear foram desenvolvidos em Python empregando o módulo SVC com função *kernel* linear disponibilizada pela biblioteca *scikit-learn*. Como vimos, esse modelo contém um hiperparâmetro de regularização C que precisa

ser ajustado para que o classificador resultante possua a maior capacidade de generalização possível, apresentando bom desempenho para dados não utilizados diretamente em seu treinamento. Para isso empregamos a técnica de validação cruzada, treinando o SVM linear com o mesmo conjunto de treinamento e testando diferentes valores para o hiperparâmetro C , conforme apresentados na Tabela 5.1. Cada modelo linear obtido foi avaliado no mesmo conjunto de validação e, ao final do processo, foi selecionado o SVM linear cujo valor do hiperparâmetro de regularização C levou ao maior desempenho junto aos dados de validação.

Tabela 5.2 – Procedimento para obtenção dos modelos de classificação baseados em SVM com EEGs no domínio do tempo para um usuário.

Procedimento para obtenção dos classificadores baseados em SVM
<p>1. Definições:</p> <ul style="list-style-type: none"> - $X_{treinamento}$: Conjunto de treinamento formado por EEGs de 12 Hz e 15 Hz das sessões 1 a 6. - $Y_{treinamento}$: Rótulos das classes dos EEGs de treinamento: 0 para frequência de estímulo de 12 Hz, e 1 para 15 Hz. - $X_{validação}$: Conjunto de validação formado por EEGs de 12 Hz e 15 Hz das sessões 7 e 8. - $Y_{validação}$: Rótulos das classes dos EEGs de validação.
<p>2. Pré-processamento:</p> <ol style="list-style-type: none"> 2.1. Aplicar o filtro CAR aos EEGs de treinamento e de validação. 2.2. Normalizar os EEGs de treinamento e de validação para o intervalo $[0, 1]$.
<p>3. Treinamento do SVM linear:</p> <ol style="list-style-type: none"> 3.1. Selecionar um valor para o hiperparâmetro de regularização C dentre os valores pré-determinados. 3.2. Empregar o conjunto $(X_{treinamento}, Y_{treinamento})$ para obter o SVM linear com o hiperparâmetro C investigado. 3.3. Utilizar os dados de validação $X_{validação}$ no classificador resultante, gerando as previsões $Y_{predição}$. 3.4. Calcular a acurácia $ACC(Y_{validação}, Y_{predição})$ do SVM linear junto aos dados de validação e guardar o resultado obtido e o hiperparâmetro C correspondente.
<p>4. Após realizar as etapas do passo 3 para todos os valores de C investigados, tomar como modelo final o SVM linear associado à maior acurácia/desempenho junto aos dados de validação.</p>
<p>5. Treinamento do SVM não-linear</p> <ol style="list-style-type: none"> 5.1. Selecionar uma combinação de função <i>kernel</i> e valores de hiperparâmetros. 5.2 Empregar o conjunto $(X_{treinamento}, Y_{treinamento})$ para obter o SVM não-linear correspondente à combinação do passo 5.1. 5.3. Utilizar os dados de validação $X_{validação}$ no classificador resultante, gerando as previsões $Y_{predição}$. 5.4. Calcular a acurácia $ACC(Y_{validação}, Y_{predição})$ do SVM não-linear junto aos dados de validação e guardar o resultado obtido e a combinação de hiperparâmetros definidos no passo 5.1.
<p>5. Após realizar as etapas do passo 5 para todas as combinações possíveis de hiperparâmetros (busca completa), tomar como modelo final o SVM não-linear associado à maior acurácia/desempenho junto aos dados de validação.</p>

Os classificadores baseados em SVM não-linear foram implementados de forma análoga ao SVM linear, empregando o módulo SVC do scikit-learn e a técnica de validação cruzada para obter o modelo com a melhor capacidade de generalização. A diferença neste caso está na necessidade de treinar um número consideravelmente maior de modelos, devido a todas as combinações possíveis dos valores de hiperparâmetros apresentados na Tabela 5.1. Além do hiperparâmetro de regularização, há o hiperparâmetro γ , responsável por controlar a influência de amostras individuais na fronteira de decisão. Para a função *kernel* polinomial, há ainda um terceiro hiperparâmetro: o grau do polinômio, que precisa

ser definido. A Tabela 5.2 apresenta com detalhes o procedimento realizado para obter os modelos baseados em SVM para EEGs no domínio do tempo. No domínio da frequência basta adicionar uma etapa de cálculo do módulo da DFT após a aplicação do filtro CAR no passo 2.

Diferente dos modelos SVM, os classificadores convolucionais 1D, cuja estrutura é apresentada na Tabela 5.3, foram implementados com a biblioteca de aprendizado profundo PyTorch. O CNN-1D utilizado neste trabalho possui o bloco SCU proposto em (AZNAN *et al.*, 2018), que inclui uma camada convolucional de entrada responsável por processar localmente os EEGs com filtros convolutivos, uma etapa de *batch normalization* para auxiliar no treinamento do modelo, especialmente para facilitar a retropropagação do erro, e por uma etapa de *max-pooling* que sumariza as informações processadas pela camada convolucional. Após o processamento pelo bloco SCU, as *feature maps* resultantes são vetorizadas e aplicadas a uma camada *fully-connected*. Por fim, os dados processados passam pela camada final de saída responsável por fornecer as probabilidades dos EEGs de entrada pertencerem a cada uma das classes do problema, com a tarefa de identificar se o EEG de entrada é resultado de um estímulo visual com frequência de oscilação igual a 12 Hz ou se o EEG de entrada é resultado de um estímulo visual com frequência de oscilação igual a 15 Hz.

Tabela 5.3 – Estrutura e hiperparâmetros do modelo de classificação CNN-1D utilizado no experimento. Alguns hiperparâmetros variam de acordo com o domínio investigado e com a duração do EEG.

Estrutura	Duração do EEG Hiperparâmetros do modelo	Domínio do Tempo			Domínio da Frequência		
		2 s	3 s	4 s	2 s	3 s	4 s
Convolucional 1D (Camada de entrada)	Nº de <i>kernels</i>	16	16	16	16	16	16
	Tamanho do <i>kernel</i>	5	5	5	5	5	5
	<i>Stride</i>	2	2	2	2	2	2
	Função de ativação	ReLU	ReLU	ReLU	ReLU	ReLU	ReLU
<i>Batch Normalization</i> 1D	Dimensão	16	16	16	16	16	16
<i>Max Pooling</i> 1D	Tamanho do <i>kernel</i>	2	2	2	2	2	2
	<i>Stride</i>	2	2	2	2	2	2
Linear 1	Nº de neurônios	100	100	100	100	100	100
	Nº de parâmetros	2032	3056	4080	1016	1528	2040
	Função de ativação	ReLU	ReLU	ReLU	ReLU	ReLU	ReLU
<i>Dropout</i>	Probabilidade	0.9	0.9	0.9	0.9	0.9	0.9
Linear 2 (Camada de Saída)	Nº de neurônios	2	2	2	2	2	2
	Nº de parâmetros	100	100	100	100	100	100
	Função de ativação	<i>Softmax</i>	<i>Softmax</i>	<i>Softmax</i>	<i>Softmax</i>	<i>Softmax</i>	<i>Softmax</i>

Para o treinamento dos classificadores CNN-1D utilizamos o critério de Entropia Cruzada como função de custo a ser minimizada, pois a camada de saída do modelo fornece duas medidas de probabilidade, e o algoritmo de otimização Adam (KINGMA; BA, 2015), com *learning rate* de 0.001, β_1 de 0.9, β_2 de 0.999, ϵ de 10^{-8} e *weight decay* igual a 0. Para a obtenção de um classificador CNN-1D, repetimos o processo de treinamento 15 vezes com o mesmo conjunto de dados, variando a inicialização dos pesos sinápticos

em cada execução. Após essa etapa, os 15 modelos resultantes foram avaliados junto ao mesmo conjunto de validação e o classificador com maior desempenho junto a esses dados foi escolhido como o classificador CNN-1D final do usuário. A Tabela 5.4 apresenta com detalhes o procedimento realizado para obtenção de classificadores CNN-1D.

Tabela 5.4 – Procedimento geral para obtenção dos classificadores CNN-1D para um usuário, duração e domínio do EEG.

Procedimento para obtenção dos classificadores CNN-1D
<p>1. Definições</p> <ul style="list-style-type: none"> - $X_{treinamento}$: Conjunto de treinamento formado por EEGs de 12 Hz e 15 Hz das sessões 1 a 6. - $Y_{treinamento}$: Rótulos das classes dos EEGs de treinamento: 0 para frequência de estímulo de 12 Hz, e 1 para 15 Hz. - $X_{validação}$: Conjunto de validação formado por EEGs de 12 Hz e 15 Hz das sessões 7 e 8. - $Y_{validação}$: Rótulos das classes dos EEGs de validação. - N_EPOCHS: Número de épocas de treinamento. Utilizamos um valor igual a 25. - $N_BATCHES$: Quantidade de <i>batches</i> apresentados ao classificador em uma época de treinamento. - $BATCH_SIZE$: Quantidade de EEGs que compõem um <i>batch</i>. (Varia com a duração do EEG) - N_CL: Número de classificadores que serão treinados e avaliados. Utilizamos um valor igual a 15. <p>2. Pré-processamento</p> <p>2.1. Aplicar o filtro CAR aos EEGs de treinamento e de validação.</p> <p>2.2. Se o problema de classificação for no domínio do tempo: normalizar os dados $X_{treinamento}$ e $X_{validação}$ para o intervalo $[0, 1]$.</p> <p>2.3. Se o problema de classificação for no domínio da frequência: calcular o módulo da DFT dos EEGs e normalizar os dados transformados para o intervalo $[0, 1]$.</p> <p>3. Treinamento dos classificadores</p> <p>Enquanto $i < N_CL$:</p> <p style="padding-left: 20px;">Inicializar o classificador C_i</p> <p style="padding-left: 20px;">Enquanto $e < N_EPOCHS$:</p> <p style="padding-left: 40px;">Enquanto $n < N_BATCHES$:</p> <p style="padding-left: 60px;">Zerar o vetor gradiente</p> <p style="padding-left: 60px;">Aplicar o n-ésimo batch ao modelo C_i, produzindo as saídas $Y_{predição,n}$</p> <p style="padding-left: 60px;">Calcular a função custo associada ao batch: $F_{EntropiaCruzada}(Y_{treinamento}, Y_{predição,n})$</p> <p style="padding-left: 60px;">Computar o vetor gradiente para todos os pesos sinápticos do modelo C_i</p> <p style="padding-left: 60px;">Atualizar os parâmetros do modelo C_i</p> <p style="padding-left: 40px;">$n = n + 1$</p> <p style="padding-left: 20px;">$e = e + 1$</p> <p style="padding-left: 20px;">Aplicar os dados de validação $X_{validação}$ no modelo C_i resultante, gerando as saídas $Y_{predição,C_i}$</p> <p style="padding-left: 20px;">Calcular a acurácia $ACC(Y_{validação}, Y_{predição,C_i})$ do modelo C_i e armazenar o resultado e o modelo</p> <p style="padding-left: 20px;">$i = i + 1$</p> <p>Após treinar os N_CL modelos, tomar como classificador final aquele associado ao maior valor de acurácia junto aos dados de validação.</p>

5.3 Estruturas e hiperparâmetros dos modelos generativos

Nesta seção apresentamos as arquiteturas, os valores de hiperparâmetros e a metodologia empregada para obtenção dos modelos generativos DCGAN, WGAN, VAE, AAE e AC-GAN, detalhando a estrutura das redes no domínio do tempo, no domínio da frequência, e também para os diferentes comprimentos de EEG investigados: 2 s, 3 s e 4 s, pois parte dos hiperparâmetros das camadas empregadas dependem da quantidade de atributos dos dados de entrada. Para permitir a reprodutibilidade deste trabalho apresentamos também na forma de pseudocódigos os algoritmos de treinamento de cada arquitetura.

A Tabela 5.5 apresenta a composição das redes generativa e discriminativa das DCGANs utilizadas para obter os modelos generativos dos usuários investigados neste trabalho. O fluxo de processamento inicia na camada *fully-connected* de entrada da rede generativa, que recebe um código latente de dimensão 64 amostrado a partir de uma distribuição normal padrão. Em seguida, a saída da camada inicial passa por um *reshape* gerando 16 *feature maps* com número de atributos reduzido, e são processados por uma sequência de 3 camadas convolucionais transpostas que aumentam gradativamente a dimensão das *feature maps* até alcançarem o número de atributos (ou tamanho) do EEG-SSVEP no domínio e comprimento especificados, e, ao final do treinamento, as saídas serão EEGs-SSVEP artificiais. Por outro lado, a rede discriminativa recebe como entrada ou EEGs reais ou EEGs artificiais sintetizados pela rede generativa, e processam esses EEGs com uma camada convolucional 1D, seguido por *batch normalization* e *max-pooling*. Por fim, os *feature maps* são vetorizados e processados por duas camadas *fully-connected*, com a segunda representando a camada de saída responsável por fornecer a probabilidade do EEG de entrada ser real ou sintético.

Tabela 5.5 – Estrutura e hiperparâmetros do modelo generativo DC-GAN utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos.

Rede	Estrutura	Duração do EEG Hiperparâmetros	Domínio do Tempo			Domínio da Frequência		
			2 s	3 s	4 s	2 s	3 s	4 s
Generativa	Linear 1 (Entrada: Código latente)	Nº de neurônios Nº de parâmetros Função de Ativação	320 64 PReLU	320 64 PReLU	320 64 PReLU	160 64 PReLU	160 64 PReLU	160 64 PReLU
	<i>Reshape</i>	(320 ou 160) para	(16, 20)	(16, 20)	(16, 20)	(16, 10)	(16, 10)	(16, 10)
	Convolução Transposta 1D 1	Nº de Kernels Tamanho do Kernel Stride Função de Ativação	32 22 3 PReLU	32 22 3 PReLU	32 22 3 PReLU	128 10 3 PReLU	128 10 3 PReLU	128 8 3 PReLU
	Convolução Transposta 1D 2	Nº de Kernels Tamanho do Kernel Stride Função de Ativação	32 18 3 PReLU	32 18 3 PReLU	32 18 3 PReLU	128 14 3 PReLU	128 16 3 PReLU	128 12 3 PReLU
	Convolução Transposta 1D 3	Nº de Kernels Tamanho do Kernel Stride Função de Ativação	1 10 2 Tanh	1 15 3 Tanh	1 20 4 Tanh	1 15 2 Tanh	1 16 3 Tanh	1 12 3 Tanh
	Convolução 1D	Nº de Kernels Tamanho do Kernel Stride Função de Ativação	16 10 2 <i>LeakyReLU(0.2)</i>	16 10 2 <i>LeakyReLU(0.2)</i>	16 10 2 <i>LeakyReLU(0.2)</i>	16 10 2 <i>LeakyReLU(0.2)</i>	16 10 2 <i>LeakyReLU(0.2)</i>	16 10 2 <i>LeakyReLU(0.2)</i>
	<i>Batch Normalization</i> 1D	Dimensão	16	16	16	16	16	16
	<i>Max-Pooling</i>	Tamanho do Kernel Stride	2 2	2 2	2 2	2 2	2 2	2 2
	Linear 1	Nº de neurônios Nº de parâmetros Função de Ativação	600 2016 <i>LeakyReLU(0.2)</i>	600 3040 <i>LeakyReLU(0.2)</i>	600 4064 <i>LeakyReLU(0.2)</i>	600 992 <i>LeakyReLU(0.2)</i>	600 1504 <i>LeakyReLU(0.2)</i>	600 2016 <i>LeakyReLU(0.2)</i>
	Linear 2	Nº de neurônios Nº de parâmetros Função de Ativação	1 600 Sigmoide	1 600 Sigmoide	1 600 Sigmoide	1 600 Sigmoide	1 600 Sigmoide	1 600 Sigmoide

O treinamento da DCGAN é descrito na Tabela 5.6. O algoritmo de otimização escolhido para o treinamento foi o algoritmo Adam com *learning rate* de 0.001, β_1 de 0.9, β_2 de 0.999, ϵ de 10^{-8} e *weight decay* igual a 0, e a função custo a ser minimizada durante o processo de otimização foi a Entropia Cruzada Binária (BCE, do inglês *Binary Cross-Entropy Loss*) que fornece uma medida do quão próximo as probabilidades previstas pela rede discriminativa estão das probabilidades reais, isto é, 0 para EEGs sintéticos e 1 para

Tabela 5.6 – Procedimento para treinamento de modelos DCGAN para um usuário utilizando EEGs com duração e domínio únicos.

Procedimento para obtenção de modelos generativos com DCGANs
<p>1. Definições</p> <ul style="list-style-type: none"> - $X_{T,12Hz}$: Conjunto de treinamento formado por EEGs de 12 Hz das sessões 1 a 6. - $X_{T,15Hz}$: Conjunto de treinamento formado por EEGs de 15 Hz das sessões 1 a 6. - $Y_{T,Real}$: Rótulos iguais a 1 para identificar os dados fornecidos à rede discriminativa como reais. - $Y_{T,Fake}$: Rótulos iguais a 0 para identificar os dados fornecidos à rede discriminativa como sintéticos. - N_EPOCHS: Número de épocas de treinamento. Utilizamos um valor igual a 400. - $LATENT_DIM$: Dimensão do códigos latentes que são gerados a partir de uma distribuição normal padrão e utilizados como entradas da rede generativa. Utilizamos um valor igual a 64. <p>2. Pré-processamento</p> <p>2.1. Aplicar o filtro CAR aos EEGs de treinamento $X_{T,12Hz}$ e $X_{T,15Hz}$.</p> <p>2.2. Se a síntese é no domínio do tempo: normalizar os dados $X_{T,12Hz}$ e $X_{T,15Hz}$ para o intervalo $[-1, 1]$.</p> <p>2.3. Se a síntese é no domínio da frequência: calcular o módulo da DFT dos EEGs dos conjuntos $X_{T,12Hz}$ e $X_{T,15Hz}$ e normalizar os dados transformados para o intervalo $[-1, 1]$.</p> <p>3. Treinamento da DC-GAN para obter um modelo generativo que sintetize EEGs-SSVEP de 12 Hz:</p> <p style="padding-left: 20px;">Criar uma instância da rede generativa G</p> <p style="padding-left: 20px;">Criar uma instância da rede discriminativa D</p> <p style="padding-left: 20px;">Inicializar os pesos sinápticos de ambas as redes com a inicialização <i>Xavier Uniform</i></p> <p style="padding-left: 20px;">Zerar todos os parâmetros de <i>bias</i></p> <p style="padding-left: 20px;">Enquanto $e < N_EPOCHS$:</p> <p style="padding-left: 40px;">Enquanto $n < N_BATCHES$:</p> <p style="padding-left: 60px;">Treinamento da rede discriminativa D:</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente - Aplicar o n-ésimo <i>batch</i> $X_{T,n,12Hz}$ de dados reais com os rótulos $Y_{T,n,Real}$ à rede discriminativa D, gerando as saídas $Y_{D,real,n}$ - Gerar um <i>batch</i> de códigos latentes Z - Aplicar o <i>batch</i> de códigos latentes Z na rede generativa G, produzindo um <i>batch</i> de dados sintéticos $X_{Fake,n,12Hz}$ - Aplicar o <i>batch</i> $X_{Fake,n,12Hz}$ de dados sintéticos com os rótulos $Y_{T,n,Fake}$ à rede discriminativa D, gerando as saídas $Y_{D,fake,n}$ - Calcular as funções custo BCE associadas aos <i>batches</i> real $X_{T,n,12Hz}$ e <i>fake</i> $X_{Fake,n,12Hz}$ - Congelar os parâmetros da rede generativa - Computar o vetor gradiente para todos os parâmetros da rede discriminativa - Atualizar os parâmetros da rede discriminativa <p style="padding-left: 60px;">Treinamento da rede generativa G:</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente - Gerar um <i>batch</i> de códigos latentes Z - Aplicar o <i>batch</i> de códigos latentes Z na rede generativa G, produzindo um <i>batch</i> de dados sintéticos $X_{Fake,n,12Hz}$ - Aplicar o <i>batch</i> $X_{Fake,n,12Hz}$ de dados sintéticos com os rótulos $Y_{T,n,Real}$ à rede discriminativa D, gerando as saídas $Y_{G,fake,n}$ - Calcular as função custo BCE associada ao <i>batch fake</i> $X_{Fake,n,12Hz}$ - Congelar os parâmetros da rede discriminativa - Computar o vetor gradiente para todos os parâmetros da rede generativa - Atualizar os parâmetros da rede generativa <p style="padding-left: 40px;">$n = n + 1$</p> <p style="padding-left: 40px;">$e = e + 1$</p> <p style="padding-left: 20px;">Ao final do treinamento, temos a rede generativa G_{12Hz} capaz de sintetizar EEGs-SSVEP de 12 Hz</p> <p>4. Realizar os passos da etapa 3 para obter a rede generativa G_{15Hz} capaz de sintetizar EEGs-SSVEP de 15 Hz.</p>

EEGs originais. Além disso, o treinamento adversário entre as duas redes, cujo critério foi apresentado na Equação (3.4), se dá ao fornecer rótulos corretos na etapa de treinamento da rede discriminativa, isto é, rótulos iguais a 1 para os dados reais e rótulos iguais a 0 para os dados sintéticos, e fornecendo rótulos incorretos na etapa de treinamento da rede

generativa, isto é, rótulos iguais a 1 para os EEGs artificiais a fim de que a rede geradora consiga enganar a rede discriminativa.

A Tabela 5.7 apresenta as estruturas das redes generativa e crítica empregadas na WGAN. O fluxo de processamento dos dados em ambas as redes é análogo ao da DCGAN, com a diferença de que na rede crítica as saídas finais deixam de representar medidas de probabilidade e passam a indicar o quão real ou falso o EEG fornecido aparenta ser. Com isso, a função de ativação da camada de saída da rede crítica passa a ser uma função linear e as saídas desejadas utilizadas para conduzir o treinamento do modelo deixam de indicar valores de probabilidade iguais a 0 e 1, e passam a apresentar valores iguais a -1 para indicar EEGs reais e valores iguais a 1 para indicar EEGs artificiais.

Tabela 5.7 – Estrutura e hiperparâmetros do modelo generativo WGAN utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos.

Rede	Estrutura	Duração do EEG Hiperparâmetros	Domínio do Tempo			Domínio da Frequência		
			2 s	3 s	4 s	2 s	3 s	4 s
Generativa	Linear 1 (Entrada: Código latente)	Nº de neurônios	320	320	320	160	160	160
		Nº de parâmetros	64	64	64	64	64	64
		Função de Ativação	PreLU	PreLU	PreLU	PreLU	PreLU	PreLU
	<i>Reshape</i>	(320 ou 160) para	16, 20	16, 20	(16, 20)	(16, 10)	(16, 10)	(16, 10)
	Convolução Transposta 1D 1	Nº de <i>Kernels</i>	32	32	32	128	128	128
		Tamanho do <i>Kernel</i>	22	22	22	10	10	8
		<i>Stride</i>	3	3	3	3	3	3
	Convolução Transposta 1D 2	Nº de <i>Kernels</i>	32	32	32	128	128	128
		Tamanho do <i>Kernel</i>	18	18	18	14	16	12
		<i>Stride</i>	3	3	3	3	3	3
Convolução Transposta 1D 3	Nº de <i>Kernels</i>	1	1	1	1	1	1	
	Tamanho do <i>Kernel</i>	10	15	20	15	16	12	
	<i>Stride</i>	2	3	4	2	3	3	
	Função de Ativação	Sigmóide	Sigmóide	Sigmóide	Sigmóide	Sigmóide	Sigmóide	
Crítica	Convolução 1D	Nº de <i>Kernels</i>	16	16	16	16	16	16
		Tamanho do <i>Kernel</i>	10	10	10	10	10	10
		<i>Stride</i>	2	2	2	2	2	2
		Função de Ativação	LeakyReLU(0.2)	LeakyReLU(0.2)	LeakyReLU(0.2)	LeakyReLU(0.2)	LeakyReLU(0.2)	LeakyReLU(0.2)
	<i>Batch Normalization</i> 1D	Dimensão	16	16	16	16	16	16
	<i>Max-Pooling</i>	Tamanho do <i>Kernel</i>	2	2	2	2	2	2
		<i>Stride</i>	2	2	2	2	2	2
	Linear 1	Nº de neurônios	600	600	600	600	600	600
		Nº de parâmetros	2016	3040	4064	992	1504	2016
		Função de Ativação	LeakyReLU(0.2)	LeakyReLU(0.2)	LeakyReLU(0.2)	LeakyReLU(0.2)	LeakyReLU(0.2)	LeakyReLU(0.2)
Linear 2	Nº de neurônios	1	1	1	1	1	1	
	Nº de parâmetros	600	600	600	600	600	600	
	Função de Ativação	Linear	Linear	Linear	Linear	Linear	Linear	

O treinamento da WGAN é descrito na Tabela 5.8. O algoritmo de otimização utilizado foi o RMSProp (TIELEMAN; HINTON, 2012) com *learning rate* de 0.00005 e *momentum* igual a zero, e a função custo a ser minimizada no processo de otimização é a função custo Wasserstein, definida na Equação (3.5). Da mesma forma que para a DCGAN, o treinamento adversário foi conduzido fornecendo saídas corretas na etapa de treinamento da rede crítica, isto é, iguais a -1 para os EEGs reais e iguais a 1 para os EEGs artificiais, e fornecendo saídas incorretas na etapa de treinamento da rede generativa, fornecendo saídas iguais a -1 ao apresentar os EEGs artificiais para a rede crítica. Para manter a estabilidade e o treinamento não divergir, os parâmetros da rede crítica são limitados ao intervalo $[-0.01, +0.01]$.

Tabela 5.8 – Procedimento para treinamento de modelos WGAN para um usuário utilizando EEGs com duração e domínio únicos.

Procedimento para obtenção de modelos generativos com WGANs
<p>1. Definições</p> <ul style="list-style-type: none"> - $X_{T,12Hz}$: Conjunto de treinamento formado por EEGs de 12 Hz das sessões 1 a 6. - $X_{T,15Hz}$: Conjunto de treinamento formado por EEGs de 15 Hz das sessões 1 a 6. - $Y_{T,Real}$: Rótulos iguais a -1 para identificar os dados fornecidos à rede crítica como reais. - $Y_{T,Fake}$: Rótulos iguais a 1 para identificar os dados fornecidos à rede crítica como sintéticos. - N_EPOCHS: Número de épocas de treinamento. Utilizamos um valor igual a 1000. - $LATENT_DIM$: Dimensão do códigos latentes que são gerados a partir de uma distribuição normal padrão e utilizados como entradas da rede generativa. Utilizamos um valor igual a 64. - N_CRITIC: Número de vezes que a rede crítica é treinada para cada vez que a rede generativa é treinada. Utilizamos um valor igual a 5. - $CLIP_VALUE$: Constante definida para limitar os valores dos parâmetros do modelo. Utilizamos um valor igual a 0.01. <p>2. Pré-processamento</p> <p>2.1. Aplicar o filtro CAR aos EEGs de treinamento $X_{T,12Hz}$ e $X_{T,15Hz}$.</p> <p>2.2. Se a síntese é no domínio do tempo: normalizar os dados $X_{T,12Hz}$ e $X_{T,15Hz}$ para o intervalo [-1, 1].</p> <p>2.3. Se a síntese é no domínio da frequência: calcular o módulo da DFT dos EEGs dos conjuntos $X_{T,12Hz}$ e $X_{T,15Hz}$ e normalizar os dados transformados para o intervalo [0, 1].</p> <p>3. Treinamento da WGAN para obter um modelo generativo que sintetize EEGs-SSVEP de 12 Hz:</p> <p style="padding-left: 20px;">Criar uma instância da rede generativa G</p> <p style="padding-left: 20px;">Criar uma instância da rede crítica C</p> <p style="padding-left: 20px;">Inicializar os pesos sinápticos de ambas as redes com a inicialização <i>Xavier Uniform</i></p> <p style="padding-left: 20px;">Zerar todos os parâmetros de <i>bias</i></p> <p style="padding-left: 20px;">Enquanto $e < N_EPOCHS$:</p> <p style="padding-left: 40px;">Enquanto $n < N_BATCHES$:</p> <p style="padding-left: 60px;">Treinamento da rede crítica C:</p> <p style="padding-left: 60px;">Enquanto $i < N_CRITIC$:</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente - Aplicar o i-ésimo <i>batch</i> $X_{T,i,12Hz}$ de dados reais com os rótulos $Y_{T,i,Real}$ à rede crítica C, gerando as saídas $Y_{C,real,i}$ - Gerar um <i>batch</i> de códigos latentes Z - Aplicar o <i>batch</i> de códigos latentes Z na rede generativa G, produzindo um <i>batch</i> de dados sintéticos $X_{Fake,i,12Hz}$ - Aplicar o <i>batch</i> $X_{Fake,i,12Hz}$ de dados sintéticos com os rótulos $Y_{T,i,Fake}$ à rede crítica C, gerando as saídas $Y_{C,fake,i}$ - Calcular as funções custo Wasserstein associadas aos <i>batches</i> real $X_{T,i,12Hz}$ e <i>fake</i> $X_{Fake,i,12Hz}$ - Congelar os parâmetros da rede generativa - Computar o vetor gradiente para todos os parâmetros da rede crítica - Atualizar os parâmetros da rede crítica <p style="padding-left: 60px;">$i = i + 1$</p> <p style="padding-left: 40px;">Limita todos os parâmetros para o intervalo $[-CLIP_VALUE, +CLIP_VALUE]$</p> <p style="padding-left: 60px;">Treinamento da rede generativa G:</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente - Gerar um <i>batch</i> de códigos latentes Z - Aplicar o <i>batch</i> de códigos latentes Z na rede generativa G, produzindo um <i>batch</i> de dados sintéticos $X_{Fake,n,12Hz}$ - Aplicar o <i>batch</i> $X_{Fake,n,12Hz}$ de dados sintéticos com os rótulos $Y_{T,n,Real}$ à rede crítica C, gerando as saídas $Y_{G,fake,n}$ - Calcular a função custo Wasserstein associada ao <i>batch fake</i> $X_{Fake,n,12Hz}$ - Congelar os parâmetros da rede crítica - Computar o vetor gradiente para todos os parâmetros da rede generativa - Atualizar os parâmetros da rede generativa <p style="padding-left: 40px;">$n = n + 1$</p> <p style="padding-left: 20px;">$e = e + 1$</p> <p>Ao final do treinamento, temos a rede generativa G_{12Hz} capaz de sintetizar EEGs-SSVEP de 12 Hz</p> <p>4. Realizar os passos da etapa 3 para obter a rede generativa G_{15Hz} capaz de sintetizar EEGs de 15 Hz.</p>

A Tabela 5.9 apresenta a composição das redes codificadora e decodificadora dos *autoencoders* variacionais VAE. O fluxo de processamento inicia na rede codificadora, que recebe como entrada os EEGs-SSVEP reais da base de dados. Esses EEGs são processados por uma camada convolucional 1D, seguido por *batch normalization* e *max-pooling*, resultando em um conjunto de *feature maps* transformados. Em seguida, os *feature maps* são vetorizados acoplando uma *feature map* ao final de outra e então processadas por uma camada *fully-connected* cuja saída alimenta as duas camadas de saída da rede codificadora, uma responsável por fornecer um vetor de médias e outra responsável por fornecer um vetor de variâncias. Finalizado o fluxo na rede codificadora, os vetores de média e variância são combinados a uma variável amostrada a partir de uma distribuição normal padrão, explorando o truque da reparametrização explicado na Seção 3.1, de forma que os códigos latentes são finalmente produzidos e servem como entradas da rede decodificadora, cuja missão é reconstruir os EEGs-SSVEP fornecidos inicialmente na entrada do modelo.

Tabela 5.9 – Estrutura e hiperparâmetros do modelo generativo VAE utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos.

Rede	Estrutura	Duração do EEG Hiperparâmetros	Domínio do Tempo			Domínio da Frequência		
			2 s	3 s	4 s	2 s	3 s	4 s
Codificadora	Convolução 1D	Nº de <i>Kernels</i>	16	16	16	16	16	16
		Tamanho do <i>Kernel Stride</i>	10	10	10	10	10	10
		Função de Ativação	2	2	2	2	2	2
	<i>Batch Normalization</i> 1D	Dimensão	<i>LeakyReLU</i> (0.2)					
	<i>Max-Pooling</i>	Tamanho do <i>Kernel Stride</i>	16	16	16	16	16	16
	Linear 1	Nº de neurônios	2	2	2	2	2	2
		Nº de parâmetros	2	2	2	2	2	2
Linear (Médias)	Nº de neurônios	600	600	600	600	600	600	
	Nº de parâmetros	2016	3040	4064	992	1504	2016	
	Função de Ativação	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	
Linear (Variâncias)	Nº de neurônios	64	64	64	64	64	64	
	Nº de parâmetros	600	600	600	600	600	600	
	Função de Ativação	Linear	Linear	Linear	Linear	Linear	Linear	
Decodificadora	Linear 1 (Entrada: Código latente)	Nº de neurônios	64	64	64	64	64	64
		Nº de parâmetros	600	600	600	600	600	600
		Função de Ativação	Linear	Linear	Linear	Linear	Linear	Linear
	<i>Reshape</i>	(320 ou 160) para	320	320	320	160	160	160
	Convolução Transposta 1D 1	Nº de <i>Kernels</i>	64	64	64	64	64	64
		Tamanho do <i>Kernel Stride</i>	PReLU	PReLU	PReLU	PReLU	PReLU	PReLU
		Função de Ativação	16, 20	16, 20	(16, 20)	(16, 10)	(16, 10)	(16, 10)
Convolução Transposta 1D 2	Nº de <i>Kernels</i>	32	32	32	128	128	128	
	Tamanho do <i>Kernel Stride</i>	22	22	22	10	10	8	
	Função de Ativação	3	3	3	3	3	3	
Convolução Transposta 1D 3	Nº de <i>Kernels</i>	PReLU	PReLU	PReLU	PReLU	PReLU	PReLU	
	Tamanho do <i>Kernel Stride</i>	32	32	32	128	128	128	
	Função de Ativação	18	18	18	14	16	12	
Convolução Transposta 1D 3	Nº de <i>Kernels</i>	3	3	3	3	3	3	
	Tamanho do <i>Kernel Stride</i>	PReLU	PReLU	PReLU	PReLU	PReLU	PReLU	
	Função de Ativação	1	1	1	1	1	1	
Convolução Transposta 1D 3	Nº de <i>Kernels</i>	10	15	20	15	16	12	
	Tamanho do <i>Kernel Stride</i>	2	3	4	2	3	3	
	Função de Ativação	Sigmoíde	Sigmoíde	Sigmoíde	Sigmoíde	Sigmoíde	Sigmoíde	

O treinamento do VAE é mais simples que o dos modelos baseados em GANs e está resumido na Tabela 5.10. Também utilizamos o algoritmo Adam com *learning rate* de 0.001, β_1 de 0.9, β_2 de 0.999, ϵ de 10^{-8} e *weight decay* igual a zero para o treinamentos desse modelo e, como função custo a ser minimizada temos (i) a função L_{1Loss} para a perda de reconstrução dos EEGs, que computa a soma dos módulos dos erros associados a cada par de atributos correspondentes dos EEGs de entrada e reconstruído, e (ii) o divergente de

Kullback-Leibler para a função custo de regularização, que busca aproximar a distribuição de probabilidade gerada pela rede codificadora à distribuição normal padrão.

Tabela 5.10 – Procedimento para treinamento de modelos VAE para um usuário utilizando EEGs com duração e domínio únicos.

Procedimento para obtenção de modelos generativos com VAEs
<p>1. Definições</p> <ul style="list-style-type: none"> - $X_{T,12Hz}$: Conjunto de treinamento formado por EEGs de 12 Hz das sessões 1 a 6. - $X_{T,15Hz}$: Conjunto de treinamento formado por EEGs de 15 Hz das sessões 1 a 6. - N_EPOCHS: Número de épocas de treinamento. Utilizamos um valor igual a 3000. <p>2. Pré-processamento</p> <p>2.1. Aplicar o filtro CAR aos EEGs de treinamento $X_{T,12Hz}$ e $X_{T,15Hz}$.</p> <p>2.2. Se a síntese é no domínio do tempo: normalizar os dados $X_{T,12Hz}$ e $X_{T,15Hz}$ para o intervalo $[0, 1]$.</p> <p>2.3. Se a síntese é no domínio da frequência: calcular o módulo da DFT dos EEGs dos conjuntos $X_{T,12Hz}$ e $X_{T,15Hz}$ e normalizar os dados transformados para o intervalo $[0, 1]$.</p> <p>3. Treinamento do VAE para obter um modelo generativo que sintetize EEGs-SSVEP de 12 Hz:</p> <ul style="list-style-type: none"> Criar uma instância da rede codificadora E Criar uma instância da rede decodificadora D Inicializar os pesos sinápticos de ambas as redes com a inicialização <i>Xavier Uniform</i> Zerar todos os parâmetros de <i>bias</i> Enquanto $e < N_EPOCHS$: <ul style="list-style-type: none"> Enquanto $n < N_BATCHES$: <ul style="list-style-type: none"> - Zerar o vetor gradiente - Aplicar o n-ésimo <i>batch</i> $X_{T,n,12Hz}$ de dados reais à rede codificadora E, gerando os vetores de média e variância - Gerar um <i>batch</i> de códigos latentes Z a partir dos vetores de média e variância e com o truque da reparametrização - Aplicar o <i>batch</i> de códigos latentes Z na rede decodificadora D, reconstruindo o <i>batch</i> de dados reais $X_{Reconstrução,n,12Hz}$ - Calcular as funções custo de reconstrução (L1) e de regularização (Divergente de Kullback-Leibler) - Computar o vetor gradiente para todos os parâmetros das redes codificadora e decodificadora - Atualizar os parâmetros das redes codificadora e decodificadora $n = n + 1$ $e = e + 1$ Ao final do treinamento, a rede decodificada pode ser tomada como a rede generativa G_{12Hz} capaz de sintetizar EEGs-SSVEP de 12 Hz <p>4. Realizar os passos da etapa 3 para obter a rede generativa G_{15Hz} capaz de sintetizar EEGs-SSVEP de 15 Hz.</p>

Passamos agora para a abordagem AAE, que combina elementos de modelos baseados nas GANs com elementos de *autoencoders* variacionais, cuja estrutura é apresentada na Tabela 5.11. Assim como no VAE, a rede codificadora recebe como entrada EEGs-SSVEP originais da base de dados e produz nas camadas de saída os vetores de médias e variâncias. Posteriormente, os vetores de média e variância gerados são combinados a uma variável amostrada a partir de uma distribuição normal padrão, produzindo códigos latentes que são utilizados tanto como entrada da rede decodificadora para reconstruir os EEG-SSVEP originais fornecidos na entrada, quanto como entrada da rede discriminativa, que busca identificar a origem do código latente fornecido. Assim, a rede discriminativa recebe como entrada ou um código latente real amostrado a partir de uma distribuição normal padrão ou um código latente artificial obtido com as saídas da rede codificadora que, nesta abordagem, faz o papel de rede generativa. O código latente é,

então, processado por uma sequência de camadas *fully-connected* até produzir como saída as probabilidades de o código latente apresentado ser real ou falso.

Tabela 5.11 – Estrutura e hiperparâmetros do modelo generativo AAE utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos.

Rede	Estrutura	Duração do EEG Hiperparâmetros	Domínio do Tempo			Domínio da Frequência		
			2 s	3 s	4 s	2 s	3 s	4 s
Codificadora	Convolução 1D	Nº de <i>Kernels</i>	16	16	16	16	16	16
		Tamanho do <i>Kernel</i>	10	10	10	10	10	10
		<i>Stride</i>	2	2	2	2	2	2
		Função de Ativação	<i>LeakyReLU</i> (0.2)					
	<i>Batch Normalization</i> 1D	Dimensão	16	16	16	16	16	16
	Max-Pooling	Tamanho do <i>Kernel</i>	2	2	2	2	2	2
		<i>Stride</i>	2	2	2	2	2	2
Linear 1	Nº de neurônios	600	600	600	600	600	600	
	Nº de parâmetros	2016	3040	4064	992	1504	2016	
	Função de Ativação	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	<i>LeakyReLU</i> (0.2)	
Linear (Médias)	Nº de neurônios	64	64	64	64	64	64	
	Nº de parâmetros	600	600	600	600	600	600	
	Função de Ativação	Linear	Linear	Linear	Linear	Linear	Linear	
Linear (Variâncias)	Nº de neurônios	64	64	64	64	64	64	
	Nº de parâmetros	600	600	600	600	600	600	
	Função de Ativação	Linear	Linear	Linear	Linear	Linear	Linear	
Decodificadora	Linear 1	Nº de neurônios	320	320	320	160	160	160
		Nº de parâmetros	64	64	64	64	64	64
		Função de Ativação	PReLU	PReLU	PReLU	PReLU	PReLU	PReLU
		<i>Reshape</i>	(320) para	(16, 20)	(16, 20)	(16, 20)	(16, 10)	(16, 10)
	Convolação Transposta 1D 1	Nº de <i>Kernels</i>	32	32	32	128	128	128
		Tamanho do <i>Kernel</i>	22	22	22	10	10	8
		<i>Stride</i>	3	3	3	3	3	3
		Função de Ativação	PReLU	PReLU	PReLU	PReLU	PReLU	PReLU
	Convolação Transposta 1D 2	Nº de <i>Kernels</i>	32	32	32	128	128	128
		Tamanho do <i>Kernel</i>	18	18	18	14	16	12
<i>Stride</i>		3	3	3	3	3	3	
	Função de Ativação	PReLU	PReLU	PReLU	PReLU	PReLU	PReLU	
Convolação Transposta 1D 3	Nº de <i>Kernels</i>	1	1	1	1	1	1	
	Tamanho do <i>Kernel</i>	10	15	20	15	16	12	
	<i>Stride</i>	2	3	4	2	3	3	
	Função de Ativação	Sigmóide	Sigmóide	Sigmóide	Sigmóide	Sigmóide	Sigmóide	
Discriminativa	Linear 1	Nº de neurônios	100	100	100	100	100	100
		Nº de parâmetros	64	64	64	64	64	64
		Função de Ativação	ReLU	ReLU	ReLU	ReLU	ReLU	ReLU
	Linear 2	Nº de neurônios	1000	1000	1000	1000	1000	1000
		Nº de parâmetros	1000	1000	1000	1000	1000	1000
		Função de Ativação	ReLU	ReLU	ReLU	ReLU	ReLU	ReLU
Linear 3	Nº de neurônios	1	1	1	1	1	1	
	Nº de parâmetros	1000	1000	1000	1000	1000	1000	
	Função de Ativação	Sigmóide	Sigmóide	Sigmóide	Sigmóide	Sigmóide	Sigmóide	

O procedimento de treinamento do AAE é descrito na Tabela 5.12. O algoritmo de otimização escolhido para o treinamento dos modelos baseados em VAE foi o algoritmo Adam com *learning rate* de 0.001, β_1 de 0.9, β_2 de 0.999, ϵ de 10^{-8} e *weight decay* igual a 0. Assim como no VAE, a função custo de reconstrução do EEG corresponde à L_{1Loss} , que computa a soma dos módulos dos erros associados a cada par de atributos reconstruído dos EEGs de entrada. Contudo, na etapa de regularização o divergente de Kullback-Leibler é substituído pelo treinamento adversário usual da GAN. Ao final do treinamento, a rede decodificadora se torna a rede generativa capaz de sintetizar os EEGs no domínio e com a duração especificados.

A Tabela 5.13 apresenta a composição das redes generativa e discriminativa da abordagem com AC-GAN utilizada para obter os modelos generativos dos usuários investigados neste estudo. O fluxo de processamento dos dados nas redes generativa e discriminativa é muito semelhante ao da DC-GAN, com a diferença de que é fornecido à

Tabela 5.12 – Procedimento para treinamento de modelos AAE para um usuário utilizando EEGs com duração e domínio únicos.

Procedimento para obtenção de modelos generativos com AAEs
<p>1. Definições</p> <ul style="list-style-type: none"> - $X_{T,12Hz}$: Conjunto de treinamento formado por EEGs de 12 Hz das sessões 1 a 6. - $X_{T,15Hz}$: Conjunto de treinamento formado por EEGs de 15 Hz das sessões 1 a 6. - $Y_{T,Real}$: Rótulos iguais a 1 para identificar os dados fornecidos à rede discriminativa como reais. - $Y_{T,Fake}$: Rótulos iguais a 0 para identificar os dados fornecidos à rede discriminativa como sintéticos. - N_EPOCHS: Número de épocas de treinamento. Utilizamos um valor igual a 1000. <p>2. Pré-processamento</p> <p>2.1. Aplicar o filtro CAR aos EEGs de treinamento $X_{T,12Hz}$ e $X_{T,15Hz}$.</p> <p>2.2. Se a síntese é no domínio do tempo: normalizar os dados $X_{T,12Hz}$ e $X_{T,15Hz}$ para o intervalo $[0, 1]$.</p> <p>2.3. Se a síntese é no domínio da frequência: calcular o módulo da DFT dos EEGs dos conjuntos $X_{T,12Hz}$ e $X_{T,15Hz}$ e normalizar os dados transformados para o intervalo $[0, 1]$.</p> <p>3. Treinamento do AAE para obter um modelo generativo que sintetize EEGs-SSVEP de 12 Hz:</p> <p>Criar uma instância da rede codificadora E, rede decodificadora D e rede discriminativa C</p> <p>Inicializar os pesos sinápticos de todas as redes com a inicialização <i>Xavier Uniform</i> e zerar os <i>bias</i></p> <p>Enquanto $e < N_EPOCHS$:</p> <p> Enquanto $n < N_BATCHES$:</p> <p> Treinamento da etapa de reconstrução:</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente e aplicar o n-ésimo <i>batch</i> $X_{T,n,12Hz}$ de dados reais à rede codificadora E, gerando os vetores de média e variância - Gerar um <i>batch</i> de códigos latentes Z_n a partir dos vetores de média e variância e com o truque da reparametrização - Aplicar o <i>batch</i> de códigos latentes Z_n na rede decodificadora, reconstruindo o <i>batch</i> de dados reais $X_{Reconstrução,n,12Hz}$ - Calcular a função custo de reconstrução (L1) - Computar o vetor gradiente para todos os parâmetros das redes codificadora e decodificadora - Atualizar os parâmetros das redes codificadora e decodificadora <p> Treinamento da etapa de regularização (rede discriminativa C):</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente e gerar um <i>batch</i> de códigos latentes Z a partir de uma distribuição normal - Aplicar o <i>batch</i> de códigos Z com os rótulos $Y_{T,n,Real}$ à rede discriminativa, gerando as saídas $Y_{C,real}$ - Aplicar o n-ésimo <i>batch</i> $X_{T,n,12Hz}$ de dados reais à rede codificadora E, gerando os vetores de média e variância - Gerar um <i>batch</i> de códigos latentes <i>fake</i> $Z_{fake,n}$ a partir dos vetores de média e variância e com o truque da reparametrização - Aplicar o <i>batch</i> de códigos latentes $Z_{fake,n}$ com os rótulos $Y_{T,n,Fake}$ à rede discriminativa, gerando as saídas $Y_{C,fake,n}$ - Calcular as funções custo BCE associadas aos <i>batches</i> de códigos latentes Z e $Z_{fake,n}$ - Congelar os parâmetros da rede codificadora - Computar o vetor gradiente para todos os parâmetros da rede discriminativa - Atualizar os parâmetros da rede discriminativa <p> Treinamento da etapa de regularização (rede codificadora E):</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente e aplicar o n-ésimo <i>batch</i> $X_{T,n,12Hz}$ de dados reais à rede codificadora E, gerando os vetores de média e variância - Gerar um <i>batch</i> de códigos latentes $Z_{fake,n}$ a partir dos vetores de média e variância e com o truque da reparametrização - Aplicar o <i>batch</i> de códigos latentes $Z_{fake,n}$ com os rótulos $Y_{T,n,Real}$ à rede discriminativa C, gerando as saídas $Y_{E,fake,n}$ - Calcular a função custo BCE associada ao <i>batch</i> de códigos latente $Z_{fake,n}$ - Congelar os parâmetros da rede discriminativa - Computar o vetor gradiente para todos os parâmetros da rede codificadora - Atualizar os parâmetros da rede codificadora <p> $n = n + 1$</p> <p> $e = e + 1$</p> <p>Ao final do treinamento, a rede decodificadora pode ser tomada como a rede generativa G_{12Hz} capaz de sintetizar EEGs-SSVEP de 12 Hz</p> <p>4. Realizar os passos da etapa 3 para obter a rede generativa G_{15Hz} capaz de sintetizar EEGs-SSVEP de 15 Hz.</p>

rede generativa uma entrada adicional responsável por controlar a classe do EEG artificial sintetizado. Essa entrada de controle passa inicialmente por uma camada de *embedding*, que produz novas componentes que são acopladas como canais adicionais aos *feature maps* resultantes do processamento do código latente. Além disso, a rede discriminativa da AC-GAN apresenta duas camadas de saída, uma responsável por identificar se o EEG fornecido é real ou falso e outra responsável por identificar a frequência de evocação do EEG apresentado.

Tabela 5.13 – Estrutura e hiperparâmetros do modelo generativo AC-GAN utilizado nos experimentos computacionais para síntese de EEGs no domínio do tempo e no domínio da frequência para três comprimentos distintos.

Rede	Estrutura	Duração do EEG Hiperparâmetros	Domínio do Tempo			Domínio da Frequência		
			2 s	3 s	4 s	2 s	3 s	4 s
Generativa	<i>Embedding</i> 1 (Paralelo a Linear 1)	Nº de Classes Dimensão	2 64	2 64	2 64	2 64	2 64	2 64
	Linear 1 (entrada: código latente)	Nº de neurônios Nº de parâmetros Função de Ativação	320 64 PReLU	320 64 PReLU	320 64 PReLU	160 64 PReLU	160 64 PReLU	160 64 PReLU
	Linear 2 (saída do <i>embedding</i>)	Nº de neurônios Nº de parâmetros Função de Ativação	80 64 Linear	80 64 Linear	80 64 Linear	40 64 Linear	40 64 Linear	40 64 Linear
	<i>Reshape</i> (saída do linear 1)	(320 ou 160) para	(16, 20)	(16, 20)	(16, 20)	(16, 10)	(16, 10)	(16, 10)
	<i>Reshape</i> (saída do linear 2)	(80 ou 40) para	(4, 20)	(4, 20)	(4, 20)	(4, 10)	(4, 10)	(4, 10)
	<i>Stack</i>	<i>Stack</i>	(20, 20)	(20, 20)	(20, 20)	(20, 10)	(20, 10)	(20, 10)
	Convolução Transposta 1D 1	Nº de <i>Kernels</i> Tamanho do <i>Kernel</i> <i>Stride</i> Função de Ativação	32 22 3 PReLU	32 22 3 PReLU	32 22 3 PReLU	128 10 3 PReLU	128 10 3 PReLU	128 8 3 PReLU
	Convolução Transposta 1D 2	Nº de <i>Kernels</i> Tamanho do <i>Kernel</i> <i>Stride</i> Função de Ativação	32 18 3 PReLU	32 18 3 PReLU	32 18 3 PReLU	128 14 3 PReLU	128 16 3 PReLU	128 12 3 PReLU
	Convolução Transposta 1D 3	Nº de <i>Kernels</i> Tamanho do <i>Kernel</i> <i>Stride</i> Função de Ativação	1 10 2 Sigmoidé	1 15 3 Sigmoidé	1 20 4 Sigmoidé	1 15 2 Sigmoidé	1 16 3 Sigmoidé	1 12 3 Sigmoidé
	Discriminativa e Classificadora Auxiliar	Convolução 1D	Nº de <i>Kernels</i> Tamanho do <i>Kernel</i> <i>Stride</i> Função de Ativação	16 10 2 <i>LeakyReLU</i> (0.2)				
Batch Normalization 1D		Dimensão	16	16	16	16	16	16
Max-Pooling		Tamanho do <i>Kernel</i> <i>Stride</i>	2 2	2 2	2 2	2 2	2 2	2 2
Linear 1		Nº de neurônios Nº de parâmetros Função de Ativação	600 2016 <i>LeakyReLU</i> (0.2)	600 3040 <i>LeakyReLU</i> (0.2)	600 4064 <i>LeakyReLU</i> (0.2)	600 992 <i>LeakyReLU</i> (0.2)	600 1504 <i>LeakyReLU</i> (0.2)	600 2016 <i>LeakyReLU</i> (0.2)
Linear 2 (saída discriminativa)		Nº de neurônios Nº de parâmetros Função de Ativação	1 600 Sigmoidé	1 600 Sigmoidé	1 600 Sigmoidé	1 600 Sigmoidé	1 600 Sigmoidé	1 600 Sigmoidé
Linear 3 (saída da classe)		Nº de neurônios Nº de parâmetros Função de Ativação	2 600 Softmax	2 600 Softmax	2 600 Softmax	2 600 Softmax	2 600 Softmax	2 600 Softmax

O treinamento da AC-GAN é descrito com detalhes na Tabela 5.15. O algoritmo de otimização escolhido para o treinamento dos modelos DC-GAN foi o algoritmo Adam com *learning rate* de 0.001, β_1 de 0.9, β_2 de 0.999, ϵ de 10^{-8} e *weight decay* igual a 0, e a funções custo a serem minimizadas durante o processo de otimização são: (i) a Entropia Cruzada Binária para o problema de identificar se o EEG apresentado é real ou falso e (ii) uma segunda função de Entropia Cruzada para o problema de identificar a frequência de evocação, 12 Hz ou 15 Hz, do EEG analisado. Do mesmo modo que foi realizado no treinamento da DCGAN, o treinamento adversarial do modelo é conduzido fornecendo as saídas corretas na etapa de treinamento da rede discriminativa e rótulos

incorretos na etapa de treinamento da rede generativa. Diferente das técnicas apresentadas anteriormente, a AC-GAN resultante é capaz de sintetizar de forma controlada tanto EEGs-SSVEP de 12 Hz quanto EEGs-SSVEP de 15 Hz, não sendo necessário o treinamento de dois modelos separadamente.

Por fim, neste trabalho também estudamos o uso das técnicas DCGAN, WGAN, VAE e AAE em suas versões condicionadas para obter modelos generativos capazes de sintetizar EEGs-SSVEP nas frequências de evocação de 12 Hz e 15 Hz de forma controlável, da mesma forma que o modelo generativo resultante do treinamento da AC-GAN. A Tabela 5.14 contém as alterações nas camadas de entrada das redes generativa, discriminativa, crítica, codificadora e decodificadora de todas as redes apresentadas anteriormente nesta seção. Além da camada de entrada original de cada rede, responsável ou por processar códigos latentes ou por processar EEGs-SSVEP, uma camada de entrada secundária é inserida em paralelo à original com a função de receber e processar o rótulo da classe da frequência desejada, como indicado na Figura 3.9, para, por fim, anexar a informação do rótulo como canais ou *feature maps* adicionais. Para treinar os modelos condicionados, basta seguir os procedimentos de treinamento dos respectivos modelos não condicionados, e utilizando como bases de treinamento os mesmos conjuntos de dados e rótulos empregados no treinamento da AC-GAN, pois é necessário utilizar os EEGs de 12 Hz e 15 Hz conjuntamente.

Tabela 5.14 – Alterações na estrutura dos modelos generativos para incorporação da informação da classe para o treinamento dos modelos condicionados cDCGAN, cWGAN, cVAE e cAAE.

Rede	Estrutura	Duração do EEG Hiperparâmetros	Domínio do Tempo			Domínio da Frequência		
			2 s	3 s	4 s	2 s	3 s	4 s
Generativa e Decodificadora	<i>Embedding</i> 1 (Paralelo a Linear 1)	Nº de Classes	2	2	2	2	2	2
		Dimensão	64	64	64	64	64	64
	Linear 1 (entrada: código latente)	Nº de neurônios	320	320	320	160	160	160
		Nº de parâmetros	64	64	64	64	64	64
		Função de Ativação	PReLU	PReLU	PReLU	PReLU	PReLU	PReLU
	Linear 2 (saída do <i>embedding</i>)	Nº de neurônios	80	80	80	40	40	40
		Nº de parâmetros	64	64	64	64	64	64
Função de Ativação		Linear	Linear	Linear	Linear	Linear	Linear	
<i>Reshape</i> (saída do linear 1)	(320 ou 160) para	(16, 20)	(16, 20)	(16, 20)	(16, 10)	(16, 10)	(16, 10)	
<i>Reshape</i> (saída do linear 2)	(80 ou 40) para	(4, 20)	(4, 20)	(4, 20)	(4, 10)	(4, 10)	(4, 10)	
<i>Stack</i>	<i>Stack</i>	(20, 20)	(20, 20)	(20, 20)	(20, 10)	(20, 10)	(20, 10)	
Discriminativa, Crítica e Codificadora	<i>Embedding</i> 1 (Paralelo a Linear 1)	Nº de Classes	2	2	2	2	2	2
		Dimensão	64	64	64	64	64	64
	Linear 1 (saída do <i>embedding</i>)	Nº de neurônios	4 × 512	4 × 768	4 × 1024	4 × 257	4 × 385	4 × 513
		Nº de parâmetros	64	64	64	64	64	64
<i>Reshape</i>	<i>Reshape</i>	Linear	Linear	Linear	Linear	Linear	Linear	
<i>Stack</i> com EEG	<i>Stack</i>	(4, 512)	(4, 768)	(4, 1024)	(4, 257)	(4, 385)	(4, 513)	
			(5, 512)	(5, 768)	(5, 1024)	(5, 257)	(5, 385)	(5, 513)

Tabela 5.15 – Procedimento para treinamento de modelos AC-GAN para um usuário utilizando EEGs com duração e domínio únicos.

Procedimento para obtenção de modelos generativos com AC-GANs
<p>1. Definições</p> <ul style="list-style-type: none"> - $X_{Treinamento}$: Conjunto de treinamento formado por EEGs de 12 Hz e 15 Hz das sessões 1 a 6. - $Y_{T,Real}$: Rótulos iguais a 1 para identificar os dados fornecidos à rede discriminativa como reais. - $Y_{T,Fake}$: Rótulos iguais a 0 para identificar os dados fornecidos à rede discriminativa como sintéticos. - $Y_{Treinamento}$: Rótulos das classes dos EEGs de treinamento. Rótulo igual a 0 para 12 Hz e 1 para 15 Hz. - $Y_{12Hz,Fake}$: Rótulos iguais a 0 para os EEGs artificiais de 12 Hz. - $Y_{15Hz,Fake}$: Rótulos iguais a 1 para os EEGs artificiais de 15 Hz. - N_EPOCHS: Número de épocas de treinamento. Utilizamos um valor igual a 1000. - $LATENT_DIM$: Dimensão do códigos latentes que são gerados a partir de uma distribuição normal padrão e utilizados como entradas da rede generativa. Utilizamos um valor igual a 64. <p>2. Pré-processamento</p> <p>2.1. Aplicar o filtro CAR aos EEGs de treinamento $X_{Treinamento}$.</p> <p>2.2. Se a síntese é no domínio do tempo: normalizar os dados $X_{Treinamento}$ para o intervalo $[0, 1]$.</p> <p>2.3. Se a síntese é no domínio da frequência: calcular o módulo da DFT dos EEGs dos conjuntos $X_{Treinamento}$ e normalizar os dados transformados para o intervalo $[0, 1]$.</p> <p>3. Treinamento da AC-GAN para obter um modelo generativo que sintetize EEGs-SSVEP de 12 Hz e 15 Hz:</p> <p style="padding-left: 20px;">Criar uma instância da rede generativa G</p> <p style="padding-left: 20px;">Criar uma instância da rede discriminativa D</p> <p style="padding-left: 20px;">Inicializar os pesos sinápticos de ambas as redes com a inicialização <i>Xavier Uniform</i></p> <p style="padding-left: 20px;">Zerar todos os parâmetros de <i>bias</i></p> <p style="padding-left: 20px;">Enquanto $e < N_EPOCHS$:</p> <p style="padding-left: 40px;">Enquanto $n < N_BATCHES$:</p> <p style="padding-left: 60px;">Treinamento da rede discriminativa D:</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente - Aplicar o n-ésimo <i>batch</i> $X_{Treinamento,n}$ de dados reais com os rótulos $Y_{T,n,Real}$ e $Y_{Treinamento}$ à rede discriminativa D, gerando as saídas $Y_{D,real,n}$ e $Y_{D,real,classe,n}$ - Gerar dois <i>batches</i> de códigos latentes Z_{12Hz} e Z_{15Hz} - Aplicar os <i>batches</i> de códigos latentes Z_{12Hz} e Z_{15Hz} e os rótulos de classe correspondentes $Y_{12Hz,Fake}$ e $Y_{15Hz,Fake}$ na rede generativa G, produzindo os <i>batches</i> de dados sintéticos $X_{Fake,n,12Hz}$ e $X_{Fake,n,15Hz}$ - Aplicar os <i>batches</i> $X_{Fake,n,12Hz}$ e $X_{Fake,n,15Hz}$ de dados sintéticos com os rótulos $Y_{T,n,Fake}$, $Y_{12Hz,Fake}$ e $Y_{15Hz,Fake}$ à rede discriminativa D, gerando as saídas $Y_{D,fake,n}$ e $Y_{D,fake,classe,n}$ - Calcular as funções custo BCE associadas aos <i>batches</i> real $X_{Treinamento}$ e <i>fakes</i> $X_{Fake,n,12Hz}$ e $X_{Fake,n,15Hz}$ na tarefa de identificar os dados como reais ou sintéticos - Calcular as funções custo de entropia cruzada associadas aos <i>batches</i> real $X_{Treinamento}$ e <i>fakes</i> $X_{Fake,n,12Hz}$ e $X_{Fake,n,15Hz}$ na tarefa de identificar a frequência de evocação dos dados reais e sintéticos - Congelar os parâmetros da rede generativa - Computar o vetor gradiente para todos os parâmetros da rede discriminativa - Atualizar os parâmetros da rede discriminativa <p style="padding-left: 60px;">Treinamento da rede generativa G:</p> <ul style="list-style-type: none"> - Zerar o vetor gradiente - Gerar dois <i>batches</i> de códigos latentes Z_{12Hz} e Z_{15Hz} - Aplicar os <i>batches</i> de códigos latentes Z_{12Hz} e Z_{15Hz} e os rótulos de classe correspondentes $Y_{12Hz,Fake}$ e $Y_{15Hz,Fake}$ na rede generativa G, produzindo os <i>batches</i> de dados sintéticos $X_{Fake,n,12Hz}$ e $X_{Fake,n,15Hz}$ - Aplicar os <i>batches</i> $X_{Fake,n,12Hz}$ e $X_{Fake,n,15Hz}$ de dados sintéticos com os rótulos $Y_{T,n,Real}$, $Y_{12Hz,Fake}$ e $Y_{15Hz,Fake}$ à rede discriminativa D, gerando as saídas $Y_{G,fake,n}$ e $Y_{G,fake,classe,n}$ - Calcular as funções custo BCE associadas aos <i>batches</i> <i>fakes</i> $X_{Fake,n,12Hz}$ e $X_{Fake,n,15Hz}$ na tarefa de identificar os dados como reais ou sintéticos - Calcular as funções custo de entropia cruzada associadas aos <i>batches</i> <i>fakes</i> $X_{Fake,n,12Hz}$ e $X_{Fake,n,15Hz}$ na tarefa de identificar a frequência de evocação dos sintéticos - Congelar os parâmetros da rede discriminativa - Computar o vetor gradiente para todos os parâmetros da rede generativa - Atualizar os parâmetros da rede generativa <p style="padding-left: 40px;">$n = n + 1$</p> <p style="padding-left: 40px;">$e = e + 1$</p> <p>Ao final do treinamento, temos a rede generativa G capaz de sintetizar EEGs-SSVEP de 12 Hz e 15 Hz.</p>

5.4 Procedimento experimental

Após concluirmos a apresentação sobre a composição da base de dados de EEGs-SSVEP coletados dos pacientes em laboratório, bem como das estruturas e metodologias de treinamento utilizadas para desenvolver os modelos de classificação para detecção da frequência de evocação associada aos EEGs, e sobre as escolhas e metodologia empregadas para obtenção dos modelos generativos baseados em redes adversárias generativas e *autoencoders* variacionais capazes de sintetizar EEGs-SSVEP artificiais para os indivíduos investigados, apresentamos nesta seção a sequência de experimentos computacionais conduzidos durante a pesquisa, cujos resultados serão discutidos posteriormente no capítulo 6.

Como vimos na Seção 4.3, há atualmente uma ausência de trabalhos na literatura que tratam da síntese de EEGs-SSVEP no domínio da frequência. Com isso, uma das propostas deste trabalho é investigar como diferentes arquiteturas de modelos generativos podem ser utilizadas para sintetizar EEGs-SSVEP diretamente nesse domínio, e como o uso desses dados artificiais pode impactar no treinamento e, conseqüentemente, no desempenho dos classificadores das BCIs. Além disso, buscamos também verificar se a abordagem do problema de classificação em BCIs com paradigma SSVEP no domínio da frequência traz ganhos quando comparado ao mesmo problema abordado no domínio do tempo.

Para permitir as comparações de desempenho dos classificadores entre domínios do tempo e da frequência, com e sem o uso de dados artificiais, os experimentos computacionais foram divididos em cinco etapas:

1. Treinamento de classificadores no domínio do tempo sem o uso de EEGs-SSVEP artificiais, utilizando apenas as amostras reais contidas na base de dados coletada em laboratório.
2. Treinamento de classificadores no domínio da frequência sem o uso de dados artificiais, utilizando apenas os espectros dos EEGs-SSVEP reais da base de dados coletada em laboratório.
3. Treinamento de modelos generativos para sintetizar EEGs-SSVEP no domínio do tempo, seguido pelo treinamento de classificadores no domínio do tempo com o uso de base de dados aumentada formada por amostras reais e por amostras artificiais sintetizadas pelos modelos generativos. Durante o treinamento dos classificadores foi avaliado duas proporções de amostras reais e sintéticas, 54:54 e 54:108 amostras.
4. Treinamento de classificadores no domínio da frequência com o uso de base de dados aumentada formada pelos espectros dos EEGs-SSVEP reais e pelos espectros

das amostras sintetizadas no domínio do tempo pelos modelos generativos obtidos na etapa anterior. Assim como no domínio do tempo, durante o treinamento dos classificadores foi avaliado duas proporções de amostras reais e sintéticas, 54:54 e 54:108 amostras.

5. Treinamento de modelos generativos para sintetizar diretamente os EEGs-SSVEP, seguido pelo treinamento de classificadores também no domínio da frequência com o uso de base de dados aumentada formada por espectros dos EEGs-SSVEP reais e pelos espectros sintetizados diretamente no domínio da frequência pelos modelos generativos. Durante o treinamento dos classificadores foi avaliado duas proporções de amostras reais e sintéticas, 54:54 e 54:108 amostras.

Utilizando como exemplo o participante S01, no cenário em que seus EEGs de 12 segundos foram divididos em segmentos de 4 segundos de duração com sobreposição de 3 segundos, a sequência de experimentos computacionais teve início com o treinamento dos classificadores SVM linear, SVM não-linear e CNN-1D no domínio do tempo apenas com os EEGs-SSVEP originais de 4 segundos das sessões 1 a 6. Em seguida, as mesmas amostras reais do participante foram transformadas para o domínio da frequência através do cálculo do módulo da DFT, e os respectivos espectros dos EEGs foram utilizados no treinamento de novos classificadores SVM linear, SVM não-linear e CNN-1D, conforme apresentado no diagrama da Figura 5.5.

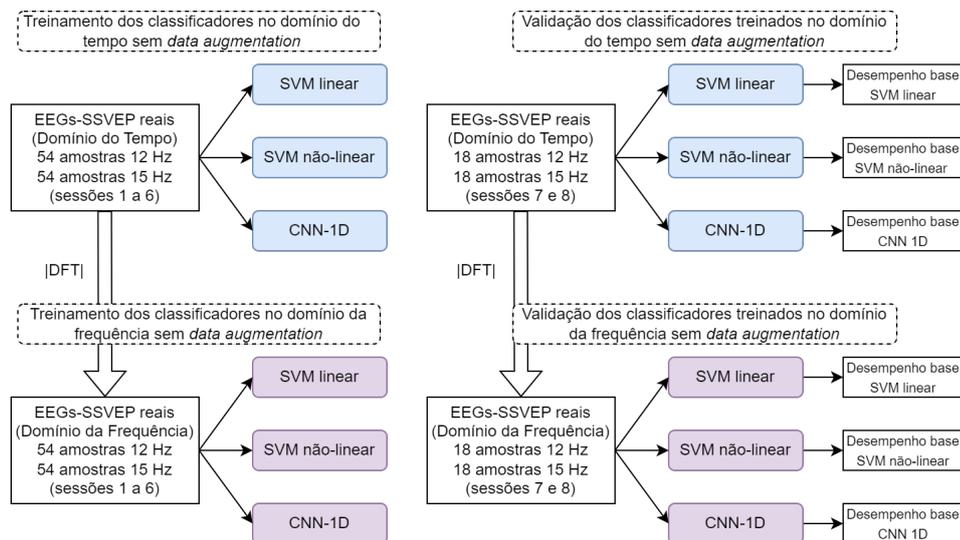


Figura 5.5 – Diagrama esquemático dos treinamentos sem *data augmentation* dos classificadores SVM linear, SVM não-linear e CNN-1D nos domínios do tempo e da frequência com EEGs-SSVEP de 4 segundos de duração do participante S01.

Finalizado o procedimento para obtenção dos 3 classificadores no domínio do tempo e dos 3 classificadores no domínio da frequência utilizando apenas dados reais do

participante S01, os mesmos conjuntos de dados foram empregados no treinamento dos modelos generativos DCGAN, WGAN, VAE, AAE, AC-GAN, cDCGAN, cWGAN, cVAE e cAAE no domínio do tempo, para os modelos resultantes serem capazes de sintetizar EEGs-SSVEP artificiais, e depois no domínio da frequência, para os modelos resultantes serem capazes de sintetizar diretamente os espectros dos EEGs.

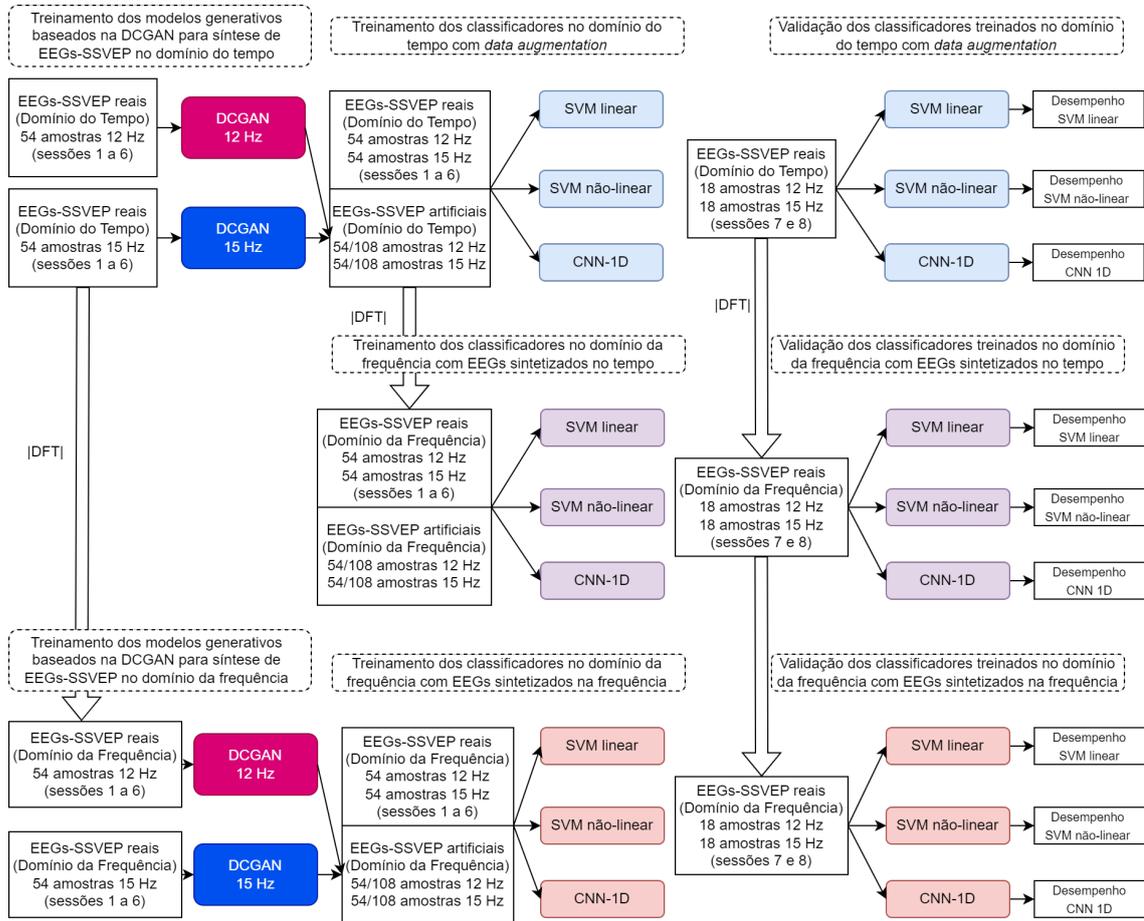


Figura 5.6 – Diagrama esquemático dos treinamentos com *data augmentation* dos classificadores SVM linear, SVM não-linear e CNN-1D nos domínios do tempo e da frequência com EEGs-SSVEP de 4 segundos de duração do participante S01. Neste diagrama utilizamos como exemplo os modelos generativos obtidos a partir do treinamento da DC-GAN.

Com os 9 modelos generativos capazes de sintetizar EEGs-SSVEP no domínio do tempo, foram treinados um SVM linear, um SVM não-linear e um CNN-1D para cada base de dados aumentada formada pelos EEGs-SSVEP reais do participante S01 e pelos EEGs artificiais sintetizados por cada um dos modelos generativos da etapa anterior, resultando em um total de 9 classificadores SVM linear, 9 classificadores SVM não-linear e 9 classificadores CNN-1D obtidos com *data augmentation* no domínio do tempo.

Em seguida, com os mesmos 9 modelos generativos capazes de sintetizar EEGs-SSVEP no domínio do tempo foram treinados um SVM linear, um SVM não-linear e um

CNN-1D para cada base de dados aumentada formada pelos espectros dos EEGs-SSVEP reais do participante S01 e pelos espectros dos EEGs-SSVEP sintetizados artificialmente no domínio do tempo, totalizando 9 classificadores SVM linear, 9 classificadores SVM não-linear e 9 classificadores CNN-1D obtidos com *data augmentation* no domínio da frequência com o uso dos EEGs-SSVEP sintetizados no tempo.

Por fim, com os 9 modelos generativos capazes de sintetizar EEGs-SSVEP no domínio da frequência, foram treinados um SVM linear, um SVM não-linear e um CNN-1D para cada base de dados aumentada formada pelos espectros dos EEGs-SSVEP reais do participante S01 e pelos espectros dos EEGs-SSVEP sintetizados diretamente no domínio da frequência pelos modelos generativos, totalizando 9 classificadores SVM linear, 9 classificadores SVM não-linear e 9 classificadores CNN-1D obtidos com *data augmentation* no domínio da frequência com o uso dos EEGs-SSVEP sintetizados na frequência.

É importante destacar que para a obtenção dos modelos de classificação com *data augmentation* em todos os cenários investigados (usuário, modelo de classificação, comprimento de EEG e modelo generativo), foram utilizadas duas combinações de dados reais e sintéticos, sendo a primeira formada pelo mesmo número de amostras reais e sintéticas (1:1) e a segunda com o número de amostras sintéticas igual ao dobro do número de amostras reais (1:2). Para o problema de classificação com EEGs de 2 segundos, foram utilizadas 66 amostras reais e 66 amostras sintéticas na primeira combinação, e 66 amostras reais e 132 amostras sintéticas na segunda combinação. Para o problema de classificação com EEGs de 3 segundos, foram utilizadas 60 amostras reais e 60 amostras sintéticas na primeira combinação, e 60 amostras reais e 120 amostras sintéticas na segunda combinação. Por fim, para o problema de classificação com EEGs de 4 segundos, foram utilizadas 54 amostras reais e 54 amostras sintéticas na primeira combinação, e 54 amostras reais e 108 amostras sintéticas na segunda combinação.

A Figura 5.6 ilustra esquematicamente o treinamento dos classificadores utilizando os EEGs-SSVEP de 4 segundos sintetizados apenas pela DCGAN, pois para os outros modelos o procedimento a ser seguido é análogo. Após a obtenção dos classificadores em todos os cenários apresentados, os modelos de classificação foram avaliados utilizando o mesmo conjunto de validação, composto pelos EEGs-SSVEP das sessões 7 e 8 coletados do participante S01, permitindo a comparação dos desempenhos de cada modelo.

5.5 Fechamento

Apresentamos neste capítulo uma análise detalhada da base de dados utilizada nos experimentos computacionais, explicando como as técnicas de pré-processamento foram aplicadas aos EEGs brutos e as decisões tomadas para separar os dados em conjuntos de treinamento e validação. Discutimos também as estruturas, os hiperparâmetros e as metodologias empregadas para obtenção dos modelos de classificação baseados em SVM linear, SVM não-linear e CNN-1D, além dos modelos generativos DCGAN, WGAN, AAE, VAE e AC-GAN, apresentando todos os passos realizados durante o treinamento dos modelos citados. Por fim, descrevemos detalhadamente o procedimento experimental empregado para a geração dos resultados e posterior análise comparativa. Dessa forma, temos o terreno preparado para a apresentação dos resultados acompanhada das discussões e principais conclusões, o que será feito no próximo capítulo.

6 Resultados e discussões

Neste capítulo, apresentamos os resultados dos experimentos computacionais realizados de acordo com a metodologia descrita no Capítulo 5. Para cada um dos 10 participantes, S01 a S10, foram treinados classificadores baseados em SVM linear, SVM não-linear e CNN-1D em diferentes cenários, com o objetivo de identificar a frequência de evocação dos EEGs-SSVEP apresentados aos modelos. Esses cenários incluem o uso de EEGs-SSVEP no domínio do tempo ou no domínio da frequência, diferentes tamanhos de janela dos EEGs (2 s, 3 s e 4 s), e o emprego de bases de dados aumentadas artificialmente com amostras sintéticas geradas por cada um dos nove modelos generativos investigados: DCGAN, WGAN, VAE, AAE, AC-GAN, cDCGAN, cWGAN, cVAE e cAAE.

Na Seção 6.1, reportamos os desempenhos dos classificadores obtidos apenas com amostras reais de treinamento no domínio do tempo, estabelecendo assim as referências no tempo para cada classificador e participante. Além disso, apresentamos também os desempenhos dos mesmos classificadores quando treinados com os espectros dos EEGs-SSVEP reais, viabilizando, assim, uma análise inicial do impacto do domínio de entrada (tempo ou frequência) sobre os modelos. Para nos auxiliar na comparação entre as duas abordagens, utilizaremos um indicador que reflete o percentual de cenários em que um classificador foi capaz de atingir um mínimo de 70% de acurácia, sendo que cada cenário é dado pela combinação de usuário, comprimento de EEG e classificador empregado na tarefa.

Cada seção subsequente apresenta separadamente os resultados referentes aos classificadores treinados com bases de dados aumentadas, inicialmente no domínio do tempo, com EEGs-SSVEP sintetizados no domínio do tempo (Seção 6.2), no domínio da frequência, com os espectros de EEGs-SSVEP gerados diretamente no domínio da frequência (Seção 6.3), e no domínio da frequência, mas com os espectros calculados a partir de EEGs-SSVEP gerados no domínio do tempo (Seção 6.4). Nestas seções, as comparações entre os desempenhos com e sem *data augmentation* são feitas exclusivamente dentro do mesmo domínio.

Com todos os resultados apresentados nas Seções 6.2-6.4, comparamos na Seção 6.5 os desempenhos alcançados pelos classificadores treinados com as bases de dados aumentadas, principalmente analisando os ganhos em relação aos desempenhos base no domínio do tempo somente com dados reais, que comumente servem como ponto de partida no processo de obtenção de classificadores em projetos de BCI. Além disso, são apresentadas visões que sumarizam como os EEGs-SSVEP sintetizados por cada modelo

generativo influenciaram os desempenhos dos classificadores, seja promovendo ganhos, seja promovendo perdas.

6.1 Resultados dos classificadores treinados somente com EEGs-SSVEP reais

A Tabela 6.1 apresenta para cada participante do estudo os desempenhos dos classificadores obtidos exclusivamente com EEGs-SSVEP reais no domínio do tempo, provenientes da base de dados formada por EEGs capturados em laboratório, quando avaliados junto aos seus respectivos dados de validação. Os resultados evidenciam uma variação significativa no desempenho dos classificadores em BCIs-SSVEP, e também que apenas o uso de amostras reais no domínio do tempo para treinamento dos classificadores geralmente resulta em modelos com performance insatisfatória, já que apenas 28,8% dos desempenhos alcançados são superiores a 70%, sendo que um terço desses resultados pertence apenas ao participante S08.

Tabela 6.1 – Desempenhos dos classificadores treinados somente com EEGs-SSVEP reais no domínio do tempo quando avaliados utilizando os dados de validação.

Usuário	SVM linear			SVM não-linear			CNN-1D		
	2 s	3 s	4 s	2 s	3 s	4 s	2 s	3 s	4 s
1	0,5000	0,5250	0,5556	0,5682	0,5750	0,5833	0,6136	0,6000	0,6667
2	0,6818	0,6000	0,6389	0,6818	0,6500	0,6667	0,7500	0,7250	0,8056
3	0,5000	0,5750	0,6667	0,5455	0,6500	0,7222	0,6136	0,6250	0,6944
4	0,5909	0,6750	0,7222	0,6591	0,7250	0,6944	0,5909	0,6000	0,7500
5	0,5909	0,6000	0,6389	0,5909	0,6000	0,6111	0,6818	0,7250	0,7222
6	0,5909	0,6250	0,6667	0,6136	0,7250	0,6667	0,7273	0,8000	0,8056
7	0,5909	0,7000	0,7222	0,5909	0,7250	0,7500	0,6364	0,6750	0,6944
8	0,9091	0,9750	1,0000	0,9318	0,9750	1,0000	0,9545	0,9750	1,0000
9	0,4773	0,4750	0,4167	0,5455	0,5000	0,5000	0,5909	0,6500	0,6111
10	0,4318	0,4750	0,5278	0,4773	0,5750	0,6667	0,5455	0,5500	0,6667
Média	0,5864	0,6225	0,6556	0,6205	0,6700	0,6861	0,6705	0,6925	0,7417

A variação nos desempenhos citada anteriormente é observada ao alterar o comprimento dos EEGs, ao mudar o modelo de classificação empregado e, principalmente, ao analisar o mesmo tipo de classificador para BCIs de diferentes usuários. Por exemplo, para o participante S10, o uso de amostras de 4 segundos resultou em um ganho de 40% no desempenho do SVM não-linear em comparação com o uso de amostras de 2 segundos. Para o participante S09, o modelo convolucional CNN-1D apresentou um ganho de 47% em relação ao SVM linear na mesma tarefa de identificar a frequência de evocação dos EEGs-SSVEP de 4 segundos. Por fim, o SVM linear foi capaz de atingir o desempenho máximo de 100% para o usuário S08, enquanto chegou a apenas 41,67% para o usuário

S09, evidenciando que as características dos EEGs dos usuários são cruciais para a boa performance do modelo, principalmente quando utilizamos um único canal, como o Oz.

A Tabela 6.2 apresenta os desempenhos dos classificadores treinados com os espectros dos EEGs-SSVEP reais da mesma base de dados formada por EEGs capturados em laboratório, quando avaliados junto aos seus respectivos EEGs-SSVEP de validação também no domínio da frequência. Embora o uso exclusivo de EEGs-SSVEP reais no domínio da frequência ainda possa levar a desempenhos não muito bons (e.g., S09 com janelas menores que 4 s), os resultados indicam um aprimoramento da capacidade de distinguir as frequências de estímulo por parte dos classificadores, com 55,5% dos desempenhos alcançados superando a acurácia mínima de 70%, e com os usuários S01, S06, S08 e S10 chegando a excelentes níveis de desempenho na maior parte dos cenários estudados.

Tabela 6.2 – Desempenhos dos classificadores treinados somente com espectros dos EEGs-SSVEP reais no domínio da frequência quando avaliados utilizando os espectros dos EEGs-SSVEP do conjunto de validação.

Usuário	SVM Linear			SVM não-linear			CNN-1D		
	2 s	3 s	4 s	2 s	3 s	4 s	2 s	3 s	4 s
1	0,9318	0,9500	1,0000	0,9318	0,9500	1,0000	0,9545	0,9750	1,0000
2	0,6591	0,4750	0,5833	0,6591	0,5750	0,6111	0,6818	0,7000	0,7222
3	0,5909	0,6500	0,6111	0,7045	0,6750	0,7222	0,7045	0,7250	0,8056
4	0,6136	0,3500	0,4722	0,5455	0,5500	0,5278	0,5682	0,5750	0,6667
5	0,5455	0,4500	0,3889	0,5227	0,6000	0,5833	0,5682	0,5000	0,5556
6	0,9318	0,8250	0,8056	0,9091	0,9000	0,7778	1,0000	0,9750	1,0000
7	0,6591	0,6750	0,6944	0,7045	0,7250	0,6389	0,7955	0,8000	0,7500
8	0,9545	0,9500	1,0000	0,9545	0,9500	1,0000	0,9773	0,9500	1,0000
9	0,5000	0,4500	0,6944	0,5909	0,5500	0,7778	0,6364	0,5750	0,7500
10	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000
Média	0,7386	0,6775	0,7250	0,7523	0,7475	0,7639	0,7886	0,7775	0,8250

A Figura 6.1 nos mostra graficamente como a abordagem no domínio da frequência para o problema de classificação em BCIs impactou os desempenhos de cada usuário, considerando cada comprimento de EEG e modelo de classificador, com cada célula da Figura 6.1 representando a divisão dos desempenhos correspondentes dos classificadores das Tabelas 6.2 e 6.1. Por exemplo, para o usuário 1 com classificador SVM Linear e EEGs de 2 segundos de comprimento, o desempenho de 93,18% no domínio da frequência representou um ganho de 86,36% quando comparado ao mesmo cenário no domínio do tempo, isto é, $Ganho = 100 \times ((0,9318/0,5) - 1)\%$. Essa comparação nos permite separar os participantes em três grupos. O primeiro, formado pelos usuários S01, S06 e S10, mostra que a abordagem no domínio da frequência foi extremamente eficaz, praticamente resolvendo o problema de detectar a frequência de evocação dos EEGs-SSVEP gerados por esses usuários. O segundo, formado pelos pacientes S03, S07 e S09, não apresentou

o predomínio no domínio da frequência, mas os ganhos obtidos por essa abordagem em alguns dos cenários resultaram nas melhores performances sem *data augmentation* para eles. O terceiro grupo, formado pelos usuários S02, S04 e S05, tiveram seus desempenhos degradados no domínio da frequência, com perdas de até 48%, indicando que para esses usuários e com esses classificadores, o uso exclusivo dos EEGs-SSVEP do eletrodo Oz no domínio do tempo é mais eficiente.

Usuário	SVM linear			SVM não-linear			CNN-1D		
	2s	3s	4s	2s	3s	4s	2s	3s	4s
1	86,36%	80,95%	79,99%	63,99%	65,22%	71,44%	55,56%	62,50%	49,99%
2	-3,33%	-20,83%	-8,70%	-3,33%	-11,54%	-8,34%	-9,09%	-3,45%	-10,35%
3	18,18%	13,04%	-8,34%	29,15%	3,85%	0,00%	14,81%	16,00%	16,01%
4	3,84%	-48,15%	-34,62%	-17,24%	-24,14%	-23,99%	-3,84%	-4,17%	-11,11%
5	-7,68%	-25,00%	-39,13%	-11,54%	0,00%	-4,55%	-16,66%	-31,03%	-23,07%
6	57,69%	32,00%	20,83%	48,16%	24,14%	16,66%	37,49%	21,88%	24,13%
7	11,54%	-3,57%	-3,85%	19,22%	0,00%	-14,81%	25,00%	18,52%	8,01%
8	4,99%	-2,56%	0,00%	2,44%	-2,56%	0,00%	2,39%	-2,56%	0,00%
9	4,76%	-5,26%	66,64%	8,32%	10,00%	55,56%	7,70%	-11,54%	22,73%
10	131,59%	110,53%	89,47%	109,51%	73,91%	49,99%	83,32%	81,82%	49,99%
Média	25,95%	8,84%	10,59%	21,24%	11,57%	11,34%	17,61%	12,27%	11,23%

Figura 6.1 – Mapa de calor com os ganhos e as perdas de desempenho quando comparamos às performances dos classificadores no domínio da frequência com seus respectivos desempenhos base no domínio do tempo.

Ao analisarmos os espectros médios dos usuários na Figura 6.2, que foram calculados a partir da média dos módulos dos espectros dos EEGs-SSVEP do conjunto de treinamento, podemos verificar que para os usuários S01 e S10 as atividades cerebrais induzidas pelos estímulos visuais nas frequências de evocação de 12 Hz e 15 Hz apresentam picos muito nítidos nas frequências de estímulo e nas harmônicas, o que provavelmente facilitou a tarefa dos classificadores. Por outro lado, para os usuários S02 e S05, que tiveram perdas consideráveis quando o problema foi abordado no domínio da frequência, os espectros médios são menos intuitivos, pois não se vê picos próximos às frequências de estímulo, o que provavelmente dificultou a identificação das classes. Contudo, o usuário S06 nos mostra que mesmo apresentando espectros pouco nítidos, o domínio da frequência ainda pode promover ganhos de desempenho ao classificador da BCI e ser mais eficiente que a abordagem no domínio do tempo. Desses resultados podemos concluir que a análise dos espectros dos EEGs-SSVEP logo após a coleta dos dados é importante, pois se os espectros pertencentes ao paciente forem bem comportados e com picos nítidos nas frequências de estímulo, a abordagem no domínio da frequência pode se mostrar mais adequada e eficiente para eles.

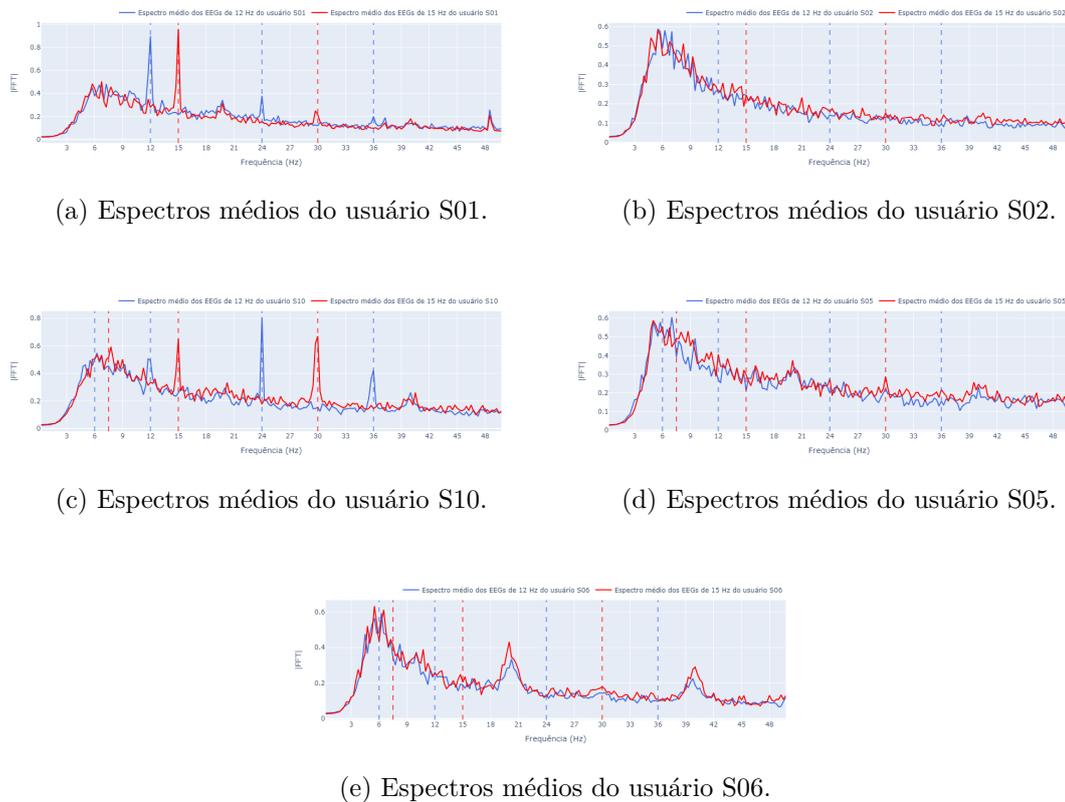


Figura 6.2 – Média dos espectros dos EEGs-SSVEP de 12 Hz e 15 Hz do conjunto de treinamento dos usuários S01, S02, S05, S06 e S10 .



Figura 6.3 – Número de vezes que cada um dos classificadores treinados sem *data augmentation* alcançou os maiores desempenhos de validação para cada cenário abordado da tarefa de identificar a frequência de evocação dos EEGs-SSVEP.

Além de permitir analisar a efetividade das abordagens nos domínios do tempo e da frequência para a classificação da BCI, os resultados possibilitam resumir os desempenhos dos modelos, verificando tanto a eficácia dos classificadores quanto o impacto do

comprimento dos EEGs utilizados. A Figura 6.3 mostra o número de usuários para os quais cada classificador, SVM linear, SVM não-linear e CNN-1D, atingiu o melhor desempenho nos diferentes domínios e tamanhos analisados. Observa-se que o classificador convolucional CNN-1D, baseado no bloco SCU proposto por (AZNAN *et al.*, 2018), foi o mais efetivo entre os três modelos, atingindo a melhor performance sem *data augmentation* para 8 dos 10 participantes em todos os cenários, exceto quando o problema foi abordado no domínio do tempo com EEGs de 3 segundos de duração, onde alcançou a melhor marca para 6 usuários. Em contrapartida, os classificadores baseados em SVM linear mostraram-se os menos eficientes na tarefa de identificar os EEGs brutos ou seus espectros, indicando que a maior flexibilidade presente em modelos com estruturas mais complexas, como nas redes convolucionais, oferece uma vantagem significativa neste contexto.

A Figura 6.4 mostra como os classificadores desempenharam em função do comprimento dos EEGs. Podemos observar que, no domínio do tempo, o uso de janelas de 4 s se mostrou a melhor opção, enquanto no domínio da frequência, ainda que tenha levado a resultados um pouco melhores, teve impacto realmente significativo apenas para a CNN-1D. Esses resultados confirmam uma tendência já esperada de que quanto maior o tamanho da janela, mais informações existem no sinal para a identificação da frequência de estímulo. Contudo, o aumento do comprimento dos EEGs reduz a quantidade de amostras para treinamento e aumenta o tempo de resposta da BCI, o que pode ser um impeditivo dependendo da aplicação. Por fim, podemos concluir que no cenário sem *data augmentation* a abordagem no domínio da frequência com EEGs de 4 segundos e utilizando classificadores CNN-1D tende a fornecer desempenhos melhores.

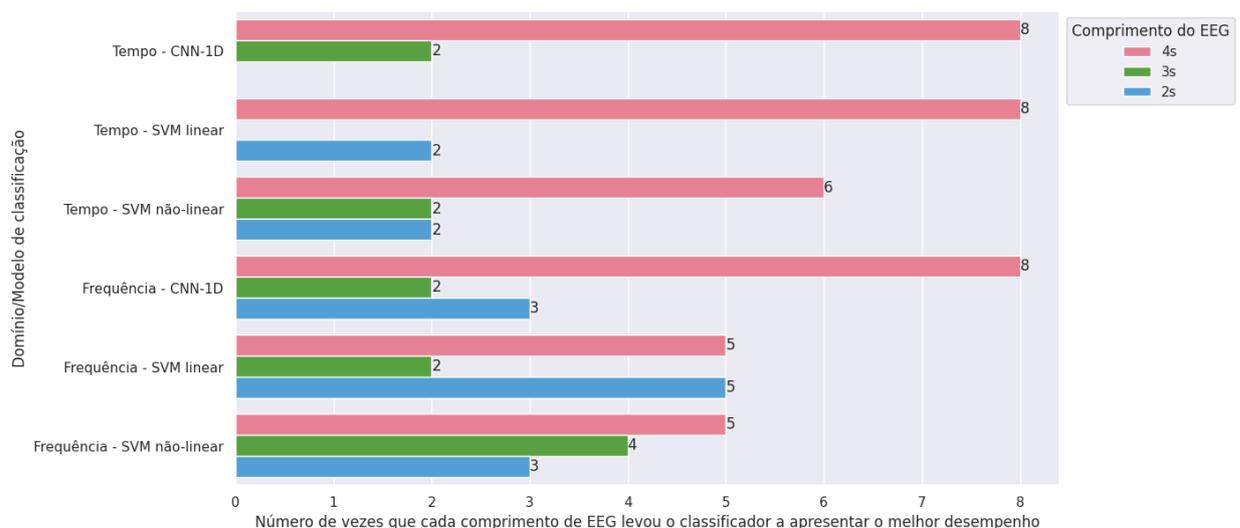


Figura 6.4 – Quantidade de vezes que cada comprimento de EEG investigado levou os classificadores treinados sem *data augmentation* a alcançar os maiores desempenhos de validação nos domínios do tempo e da frequência para a tarefa de identificar a frequência de evocação dos EEGs-SSVEP.

6.2 Resultados dos classificadores treinados com bases de dados aumentadas artificialmente no domínio do tempo

Concluídas as análises do problema de classificação da BCI sem *data augmentation*, apresentamos nesta seção os resultados atingidos no domínio do tempo por classificadores treinados com bases aumentadas formadas pelos EEGs-SSVEP reais dos usuários e por EEGs-SSVEP artificiais produzidos pelos modelos generativos estudados.

A Tabela 6.3 contém para cada participante os máximos desempenhos alcançados nos dados de validação pelos classificadores treinados com amostras sintéticas. Para conseguir esses resultados, seguimos o procedimento experimental da Seção 5.4 obtendo 9 modelos generativos distintos para cada comprimento de EEG do usuário, treinamos 9 classificadores para cada arquitetura e comprimento de EEG com as bases de dados aumentadas sinteticamente e tomamos como melhor performance para cada cenário o máximo desempenho atingido. Nessa abordagem, podemos observar que o percentual de classificadores que apresentaram desempenhos superiores a 70% no domínio do tempo foi de 61,1%, mais que dobrando esse número quando comparado ao caso sem *data augmentation*. Além disso, os classificadores CNN-1D treinados na tarefa de identificar EEGs-SSVEP de 4 segundos de comprimento alcançaram desempenhos maiores que 70% para todos os participantes analisados e um desempenho médio de 81,67%, indicando um ganho de 7,5 pontos percentuais com relação ao mesmo problema na abordagem sem o uso dos dados artificiais.

Tabela 6.3 – Maiores desempenhos para os modelos de classificação obtidos com o uso de EEGs-SSVEP artificiais sintetizados no domínio do tempo para o problema de classificar os sinais no domínio do tempo.

Usuário	SVM Linear			SVM não-linear			CNN-1D		
	2 s	3 s	4 s	2 s	3 s	4 s	2 s	3 s	4 s
1	0,6136	0,6500	0,6389	0,6818	0,7000	0,6667	0,6818	0,7000	0,7222
2	0,6818	0,7500	0,7222	0,7045	0,7750	0,7500	0,7727	0,8250	0,9167
3	0,6364	0,6000	0,6944	0,6818	0,7000	0,7778	0,6591	0,6500	0,7778
4	0,6818	0,8000	0,7778	0,7273	0,7750	0,7778	0,7500	0,7750	0,8056
5	0,7045	0,6500	0,6389	0,7045	0,6500	0,6667	0,7273	0,7750	0,8056
6	0,6591	0,8250	0,8056	0,6818	0,8000	0,8333	0,7727	0,8250	0,8333
7	0,6591	0,7250	0,7778	0,7045	0,7250	0,7778	0,7273	0,7750	0,7778
8	0,9545	0,9750	1,0000	0,9545	1,0000	1,0000	0,9545	0,9750	1,0000
9	0,6136	0,6000	0,6944	0,6364	0,6250	0,7500	0,6591	0,6500	0,7222
10	0,6364	0,6000	0,6667	0,6591	0,6750	0,6944	0,6818	0,7250	0,8056
Média	0,6841	0,7175	0,7417	0,7136	0,7425	0,7695	0,7386	0,7675	0,8167

A Figura 6.5 apresenta os ganhos alcançados no domínio do tempo referentes aos modelos treinados com *data augmentation*, com cada célula da Figura 6.5 represen-

tando a divisão dos desempenhos correspondentes dos classificadores das Tabelas 6.3 e 6.1. Em 88,8% dos casos observamos um aumento de performance com relação aos respectivos desempenhos-base sem *data augmentation*, o que demonstra a capacidade da abordagem com dados artificiais em melhorar a qualidade dos classificadores da BCI. Nos demais cenários, os resultados não foram degradados, de modo que, no pior caso, os classificadores preservam suas capacidades de generalização iniciais. Dentre os resultados alcançados, podemos destacar o ganho de 66,64% para o SVM linear com EEGs de 4 segundos do usuário S09, bem como o ganho de 47,38% do mesmo tipo de classificador para os EEGs de 2 segundos do usuário S10.

Usuário	SVM linear			SVM não-linear			CNN-1D		
	2s	3s	4s	2s	3s	4s	2s	3s	4s
1	22,72%	23,81%	14,99%	19,99%	21,74%	14,30%	11,11%	16,67%	8,32%
2	0,00%	25,00%	13,04%	3,33%	19,23%	12,49%	3,03%	13,79%	13,79%
3	27,28%	4,35%	4,15%	24,99%	7,69%	7,70%	7,42%	4,00%	12,01%
4	15,38%	18,52%	7,70%	10,35%	6,90%	12,01%	26,93%	29,17%	7,41%
5	19,22%	8,33%	0,00%	19,22%	8,33%	9,10%	6,67%	6,90%	11,55%
6	11,54%	32,00%	20,83%	11,11%	10,34%	24,99%	6,24%	3,12%	3,44%
7	11,54%	3,57%	7,70%	19,22%	0,00%	3,71%	14,28%	14,81%	12,01%
8	4,99%	0,00%	0,00%	2,44%	2,56%	0,00%	0,00%	0,00%	0,00%
9	28,56%	26,32%	66,64%	16,66%	25,00%	50,00%	11,54%	0,00%	18,18%
10	47,38%	26,32%	26,32%	38,09%	17,39%	4,15%	24,99%	31,82%	20,83%
Média	16,66%	15,26%	13,13%	15,01%	10,82%	12,15%	10,16%	10,83%	10,11%

Figura 6.5 – Ganhos de desempenho no domínio do tempo promovidos pelo uso de dados artificiais para obtenção dos classificadores da BCI.

Com o objetivo de comparar a eficiência dos modelos generativos empregados apresentamos na Figura 6.6 o número de vezes que cada modelo generativo produziu sinais artificiais que, ao serem utilizados no treinamento dos modelos de classificação, levaram a desempenhos superiores aos valores base do caso sem *data augmentation*. Nessa figura apresentamos uma visão agrupada de todos os participantes e comprimentos de EEGs, de modo que para cada classificador o indicador associado a cada modelo generativo varia entre 0 e 30, com valor máximo dado pela multiplicação da quantidade de participantes (dez), multiplicado pelo número de tamanhos distintos de janela (três).

Podemos observar que para o SVM linear e CNN-1D o modelo generativo mais efetivo foi o cVAE, enquanto para o SVM não-linear os modelos cDCGAN, WGAN e VAE obtiveram números próximos. De modo geral, porém, não verificamos o predomínio de uma técnica em relação às outras no domínio do tempo, o que mostra a necessidade de testar vários modelos generativos. Por fim, nota-se que para o classificador CNN-1D o número de modelos que o levou a atingir desempenhos maiores foi menor que para os modelos baseados em SVM, e um dos motivos para esse comportamento está no fato de que as performances dos classificadores CNN-1D na abordagem sem *data augmentation* são maiores que dos outros dois classificadores, reduzindo, assim, a margem para melhorias

com o acréscimo de novos dados.

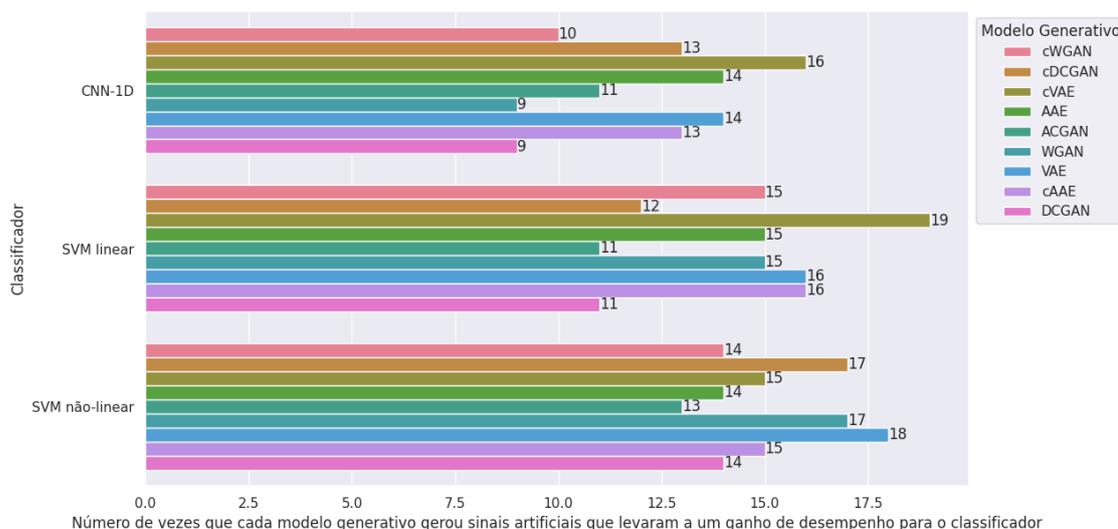


Figura 6.6 – Quantidade de cenários que o uso dos EEGs-SSVEP artificiais sintetizados por cada modelo generativo investigado levou a ganhos de desempenho para os classificadores SVM linear, SVM não-linear e CNN-1D no domínio do tempo.

6.3 Resultados dos classificadores treinados com bases de dados aumentadas artificialmente no domínio da frequência com EEGs sintetizados no domínio da frequência

Passamos agora para o domínio da frequência, em que a Tabela 6.4 apresenta para cada participante os máximos desempenhos alcançados nos dados de validação pelos classificadores treinados com amostras sintéticas produzidas no próprio domínio da frequência. Nessa abordagem, podemos observar que o percentual de classificadores que apresentaram desempenhos superiores a 70% foi de 76,6%, um aumento de 22,2 pontos percentuais quando comparado ao caso sem *data augmentation* no mesmo domínio. Ao analisarmos os desempenhos médios dos classificadores ao longo dos usuários, vemos que a CNN-1D apresentou desempenho acima de 83% para os 3 comprimentos de EEGs. Além disso, para os casos de 2 s e 4 s, essa arquitetura ultrapassou o desempenho de referência de 70% para 9 participantes.

Embora o CNN-1D com 4 segundos apresente o maior valor médio, para alguns participantes o máximo desempenho alcançado foi obtido ou por outro modelo ou por outro comprimento de EEG. Para o usuário S02 o maior desempenho foi alcançado com o SVM não-linear com EEGs de 2 segundos, enquanto para os usuários S05 e S07 as máximas performances foram atingidas pela CNN-1D com EEGs de 2 segundos, que também alcançou o desempenho máximo de 100% para os usuários S06, S08 e S10. Esses

casos são particularmente interessantes porque definem sistemas BCIs com tempos de resposta menores, o que pode ser um requisito em aplicações que se aproximem do tempo real.

Tabela 6.4 – Maiores desempenhos para os modelos de classificação obtidos com o uso de EEGs-SSVEP artificiais sintetizados no domínio da frequência para o problema de classificar os sinais no domínio da frequência.

Usuário	SVM Linear			SVM não-linear			CNN-1D		
	2 s	3 s	4 s	2 s	3 s	4 s	2 s	3 s	4 s
1	0,9545	0,9500	1,0000	0,9545	0,9500	1,0000	0,9773	1,0000	1,0000
2	0,7045	0,6500	0,6944	0,7955	0,6750	0,7222	0,7045	0,7250	0,7778
3	0,6818	0,7500	0,7778	0,7045	0,7250	0,7778	0,7955	0,8000	0,8333
4	0,6591	0,5500	0,7222	0,6136	0,5750	0,7222	0,6818	0,6250	0,7778
5	0,6591	0,6250	0,6111	0,6818	0,6500	0,6389	0,7045	0,6000	0,6667
6	0,9318	0,9250	0,8611	0,9318	0,8750	0,8333	1,0000	1,0000	1,0000
7	0,7500	0,7750	0,7500	0,7727	0,7750	0,8056	0,8182	0,8000	0,7500
8	0,9773	1,0000	1,0000	0,9773	1,0000	1,0000	1,0000	1,0000	1,0000
9	0,6364	0,6750	0,8056	0,7045	0,6750	0,8611	0,7045	0,8000	0,8056
10	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000
Média	0,7955	0,7900	0,8222	0,8136	0,7900	0,8361	0,8386	0,8350	0,8611

A Figura 6.7 apresenta os ganhos alcançados no domínio da frequência referentes aos modelos treinados com bases de dados aumentadas artificialmente por EEGs sintetizados na frequência, com cada célula da Figura 6.7 representando a divisão dos desempenhos correspondentes dos classificadores das Tabelas 6.4 e 6.2. Em 73,3% dos casos observamos um aumento de performance com relação aos respectivos desempenhos-base sem *data augmentation* na frequência, com destaque para os ganhos para os usuários S02, S04 e S05, em que a adição de amostras sintéticas permitiu que parte do desempenho degradado ao migrar do tempo para a frequência fosse recuperada para esses pacientes. Para os casos em que não houve aumento no desempenho, a maior parcela foi consequência dos classificadores já apresentarem o desempenho base no valor máximo de 100% no problema de classificação. Ainda, diferente da abordagem no domínio do tempo em que as performances ou aumentaram ou se mantiveram inalteradas para todos os cenários, o uso de dados artificiais na frequência provocou queda no desempenho em 1 dos 90 cenários, para o SVM não-linear com EEGs de 3 segundos do participante S03.

Usuário	SVM linear			SVM não-linear			CNN-1D		
	2s	3s	4s	2s	3s	4s	2s	3s	4s
1	2,44%	0,00%	0,00%	2,44%	0,00%	0,00%	2,39%	2,56%	0,00%
2	6,89%	36,84%	19,05%	20,69%	17,39%	18,18%	3,33%	3,57%	7,70%
3	15,38%	15,38%	27,28%	0,00%	7,41%	7,70%	12,92%	10,34%	3,44%
4	7,42%	57,14%	52,94%	12,48%	4,55%	36,83%	19,99%	8,70%	16,66%
5	20,82%	38,89%	57,14%	30,44%	8,33%	9,53%	23,99%	20,00%	20,00%
6	0,00%	12,12%	6,89%	2,50%	-2,78%	7,14%	0,00%	2,56%	0,00%
7	13,79%	14,81%	8,01%	9,68%	6,90%	26,09%	2,85%	0,00%	0,00%
8	2,39%	5,26%	0,00%	2,39%	5,26%	0,00%	2,32%	5,26%	0,00%
9	27,28%	50,00%	16,01%	19,22%	22,73%	10,71%	10,70%	39,13%	7,41%
10	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Média	7,70%	16,61%	13,41%	8,15%	5,69%	9,45%	6,34%	7,40%	4,38%

Figura 6.7 – Ganhos de desempenho no domínio da frequência promovidos pelo uso dos espectros de EEGs sintetizados diretamente no domínio da frequência para obtenção dos classificadores da BCI.

A Figura 6.8 exibe o número de vezes que cada modelo generativo produziu espectros artificiais de EEGs que, ao serem utilizados no treinamento dos classificadores, resultou em desempenhos superiores aos valores base da abordagem sem *data augmentation* no domínio da frequência. Podemos observar que para o SVM não-linear o modelo generativo mais efetivo foi a DCGAN, para o SVM linear foram os modelos VAE e cWGAN, enquanto para o CNN-1D também foi a DCGAN, em suas versões sem e com condicionamento. Assim como no domínio do tempo, não verificamos o predomínio de uma técnica específica em relação às outras. Por fim, verificamos que o uso das amostras sintetizadas pela WGAN se mostrou a menos efetiva aos modelos, pois dentre todas as técnicas investigadas ela foi a que apresentou os piores indicadores nos classificadores.

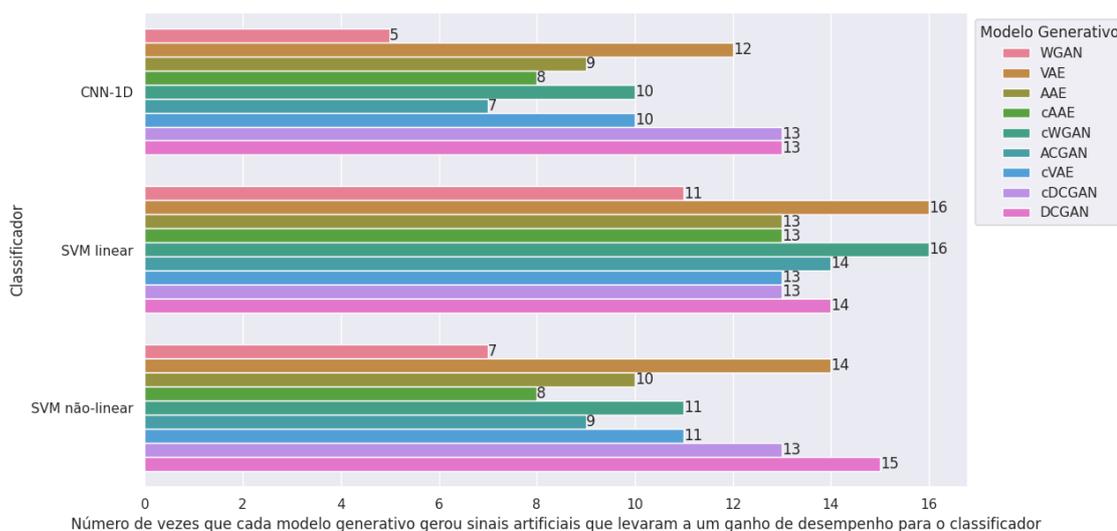


Figura 6.8 – Quantidade de cenários que o uso dos espectros dos EEGs-SSVEP artificiais sintetizados no domínio da frequência por cada modelo generativo investigado levou a ganhos de desempenho para os classificadores SVM linear, SVM não-linear e CNN-1D.

6.4 Resultados dos classificadores treinados com bases de dados aumentadas artificialmente no domínio da frequência com EEGs sintetizados no domínio do tempo

A última abordagem investigada neste trabalho relacionada ao problema de classificação de EEGs-SSVEP em BCI mescla o uso dos EEGs em ambos os domínios. Inicialmente os EEGs-SSVEP reais no domínio do tempo são empregados no treinamento de cada modelo generativo e, em seguida, tanto os dados reais quanto os EEGs-SSVEP sintetizados artificialmente no tempo pelos modelos resultantes são transformados para o domínio da frequência por meio do cálculo do módulo de suas respectivas DFTs para, por fim, serem utilizados no treinamento dos classificadores que tratam o problema de classificação no domínio da frequência. Antes de partirmos para a análise dos resultados, é importante destacar que aproveitamos os mesmos modelos generativos obtidos para a abordagem com *data augmentation* no domínio do tempo (Seção 6.2) para sintetizar os EEGs-SSVEP artificiais que foram convertidos na abordagem da presente seção.

Tabela 6.5 – Maiores desempenhos para os modelos de classificação obtidos com o uso de EEGs-SSVEP artificiais sintetizados no domínio do tempo para o problema de classificar os sinais no domínio da frequência.

Usuário	SVM Linear			SVM não-linear			CNN-1D		
	2 s	3 s	4 s	2 s	3 s	4 s	2 s	3 s	4 s
1	0,9773	1,0000	1,0000	0,9773	1,0000	1,0000	0,9773	1,0000	1,0000
2	0,6591	0,6750	0,8056	0,7273	0,6750	0,7222	0,7273	0,7250	0,7778
3	0,7955	0,8250	0,7500	0,7273	0,8500	0,7500	0,7955	0,8750	0,8611
4	0,6818	0,6000	0,7222	0,5909	0,7000	0,7778	0,6818	0,6750	0,8056
5	0,6364	0,6000	0,6944	0,7045	0,6500	0,6389	0,7727	0,6500	0,7222
6	0,9545	0,9750	1,0000	0,9773	0,9750	0,9167	1,0000	1,0000	1,0000
7	0,8182	0,8000	0,7500	0,8409	0,9000	0,7778	0,8864	0,9000	0,9167
8	1,0000	0,9750	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000
9	0,6818	0,6750	0,7500	0,7045	0,6250	0,8056	0,6591	0,7000	0,8333
10	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000
Média	0,8205	0,8125	0,8472	0,8250	0,8375	0,8389	0,8500	0,8525	0,8917

A Tabela 6.5 apresenta os maiores desempenhos alcançados nos dados de validação pelos classificadores treinados com amostras sintéticas produzidas no domínio do tempo e convertidos para o domínio da frequência. Podemos observar que o percentual de classificadores que apresentaram desempenhos superiores a 70% foi de 80%, um aumento de 25,6 pontos percentuais quando comparado ao caso sem *data augmentation* no mesmo domínio, e 3,4 pontos percentuais a mais que a abordagem com dados artificiais sintetizados na frequência (Seção 6.3). Com relação aos classificadores, apenas a CNN-1D aplica a EEGs de 4 segundos foi capaz de ultrapassar o limiar de 70% de acurácia para todos os

10 participantes do estudo, resultado que não havia sido alcançado pela primeira abordagem no domínio da frequência, além de apresentar um desempenho médio superior em 3,06 pontos percentuais. Outro número interessante, agora com relação aos desempenhos médios, é que em todos os casos esse indicador ultrapassou a marca dos 80% (ver última linha da Tabela 6.5), o que reforça a pertinência dessa abordagem.

Usuário	SVM linear			SVM não-linear			CNN-1D		
	2s	3s	4s	2s	3s	4s	2s	3s	4s
1	4,88%	5,26%	0,00%	4,88%	5,26%	0,00%	2,39%	2,56%	0,00%
2	0,00%	42,11%	38,11%	10,35%	17,39%	18,18%	6,67%	3,57%	7,70%
3	34,63%	26,92%	22,73%	3,24%	25,93%	3,85%	12,92%	20,69%	6,89%
4	11,11%	71,43%	52,94%	8,32%	27,27%	47,37%	19,99%	17,39%	20,83%
5	16,66%	33,33%	78,55%	34,78%	8,33%	9,53%	35,99%	30,00%	29,99%
6	2,44%	18,18%	24,13%	7,50%	8,33%	17,86%	0,00%	2,56%	0,00%
7	24,14%	18,52%	8,01%	19,36%	24,14%	21,74%	11,43%	12,50%	22,23%
8	4,77%	2,63%	0,00%	4,77%	5,26%	0,00%	2,32%	5,26%	0,00%
9	36,36%	50,00%	8,01%	19,22%	13,64%	3,57%	3,57%	21,74%	11,11%
10	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Média	11,08%	19,93%	16,86%	9,66%	12,04%	9,82%	7,79%	9,65%	8,08%

Figura 6.9 – Ganhos de desempenho no domínio da frequência promovidos pelo uso dos espectros de EEGs sintetizados no domínio do tempo para obtenção dos classificadores da BCI.

A Figura 6.9 apresenta os ganhos alcançados no domínio da frequência referentes aos modelos treinados com bases de dados aumentadas artificialmente por EEGs sintetizados no tempo, com cada célula da Figura 6.9 representando a divisão dos desempenhos correspondentes dos classificadores das Tabelas 6.5 e 6.2. Em 80% dos casos observamos um aumento de performance com relação aos respectivos desempenhos-base sem *data augmentation* na frequência, novamente com destaque para os ganhos referentes aos usuários S02, S04 e S05, em que o uso dos dados artificiais permitiu recuperar parte do desempenho degradado ao passar para o domínio da frequência. Para os 20% dos casos restantes em que as performances se mantiveram, o que corresponde a 18 cenários, 17 deles foi consequência dos classificadores já apresentarem o desempenho base no valor máximo de 100% no problema de classificação. Somente para a SVM linear com EEGs de 2 segundos e para o usuário S02 é que o emprego de *data augmentation* não trouxe ganhos de desempenho. É interessante notar que o mesmo classificador treinado com dados sintéticos no domínio do tempo também não apresentou aumento de desempenho, indicando uma possível dificuldade dos modelos generativos em aprender a sintetizar EEGs-SSVEP artificiais no domínio do tempo para esse participante quando utilizamos seus EEGs-SSVEP registrados pelo eletrodo Oz.

A Figura 6.10 mostra o número de vezes que cada modelo generativo produziu EEGs-SSVEP artificiais que, ao serem convertidos para o domínio da frequência e utilizados no treinamento dos classificadores, levaram a desempenhos superiores aos valores de referência (caso sem *data augmentation* no domínio da frequência). Podemos observar

que para o SVM linear o modelo generativo mais efetivo foi o VAE seguido pelo AAE, enquanto para a SVM não-linear foi o AAE em sua versão condicionada, seguido pelo VAE. Assim como nas abordagens anteriores com *data augmentation*, não verificamos o predomínio de uma única técnica para todos os classificadores, reforçando mais uma vez a necessidade de testar várias técnicas para síntese dos EEGs artificiais e escolher aquela que leva ao melhor desempenho.

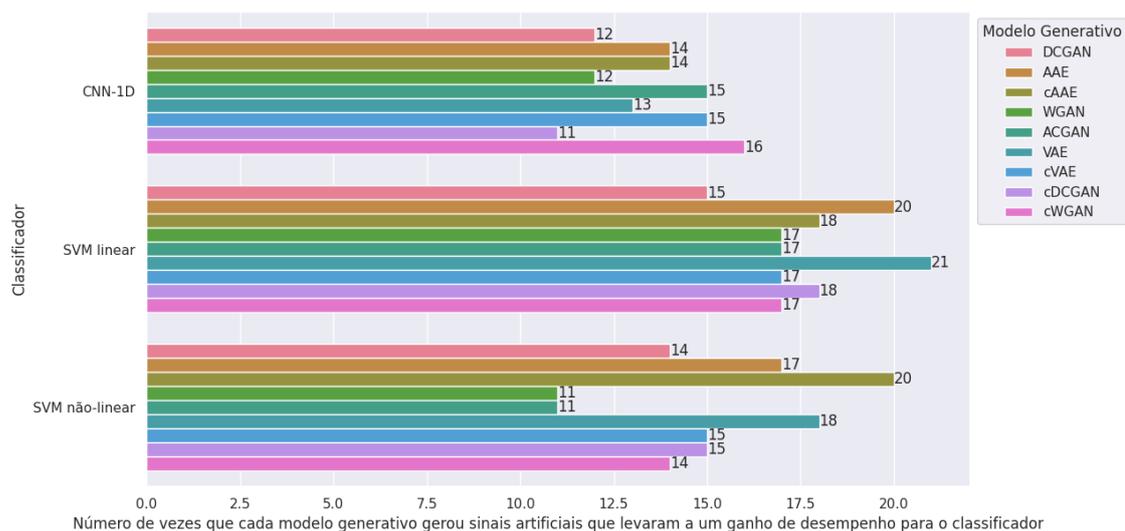


Figura 6.10 – Quantidade de cenários que o uso dos espectros dos EEGs-SSVEP artificiais sintetizados no domínio do tempo por cada modelo generativo investigado levou a ganhos de desempenho para os classificadores SVM linear, SVM não-linear e CNN-1D.

6.5 Resultados finais das abordagens com *data augmentation* nos domínios do tempo e da frequência

Nesta seção apresentamos os resultados e considerações finais com relação ao uso de dados artificiais sintetizados por modelos generativos de propósito geral para o treinamento de classificadores em BCIs baseadas no paradigma SSVEP. As Figuras 6.11, 6.12 e 6.13 apresentam, respectivamente, para os comprimentos de EEGs iguais a 2 s, 3 s e 4 s, uma comparação entre o melhor desempenho sem *data augmentation* e o melhor desempenho com *data augmentation* alcançado por cada modelo generativo investigado neste estudo. As colunas à esquerda na imagem indicam, respectivamente, o usuário, o domínio que os dados artificiais foram gerados, o domínio de classificação e o maior desempenho alcançado pelo classificador quando treinado apenas com dados reais. Já as células do mapa de calor nos fornece os ganhos correspondentes ao uso dos EEGs produzidos por cada modelo generativo (indicados na parte inferior da imagem). Por exemplo, a primeira célula do mapa da Figura 6.11 contém um valor igual a 1,111, o que indica que para o problema de classificação de EEGs-SSVEP de 2 segundos de duração do usuário S01 no

domínio do tempo, o uso das amostras geradas artificialmente pelo *autoencoder* adversarial (AAE) promoveu um ganho de 11,1% em relação ao melhor desempenho atingido (61,36%) quando somente os dados reais foram utilizados na obtenção dos classificadores (note que a comparação não leva em consideração um classificador específico, mas somente o máximo ganho obtido por algum deles).

Para os classificadores que processam EEGs-SSVEP com 2 segundos de comprimento (Figura 6.11), podemos observar que o maior ganho alcançado foi de 36% para o participante S05 no domínio da frequência com EEGs-SSVEP artificiais sintetizados pelo modelo generativo obtido a partir da técnica DCGAN. Por outro lado, a maior queda de desempenho também foi ocasionada pelo mesmo modelo generativo, mas para o usuário S03 no domínio do tempo e com EEGs sintetizados no tempo. Dentre os participantes do experimento, observa-se que os classificadores do usuário S06 foram os que encontraram maior dificuldade em superar o melhor desempenho do cenário sem *data augmentation*, pois o desempenho de 72,73% no domínio do tempo só foi ultrapassado pelo classificador obtido com o uso dos EEGs-SSVEP produzidos pela WGAN. Por outro lado, para o participante S07 todos os modelos generativos foram capazes de promover ganhos no domínio da frequência com EEGs sintetizados no tempo.

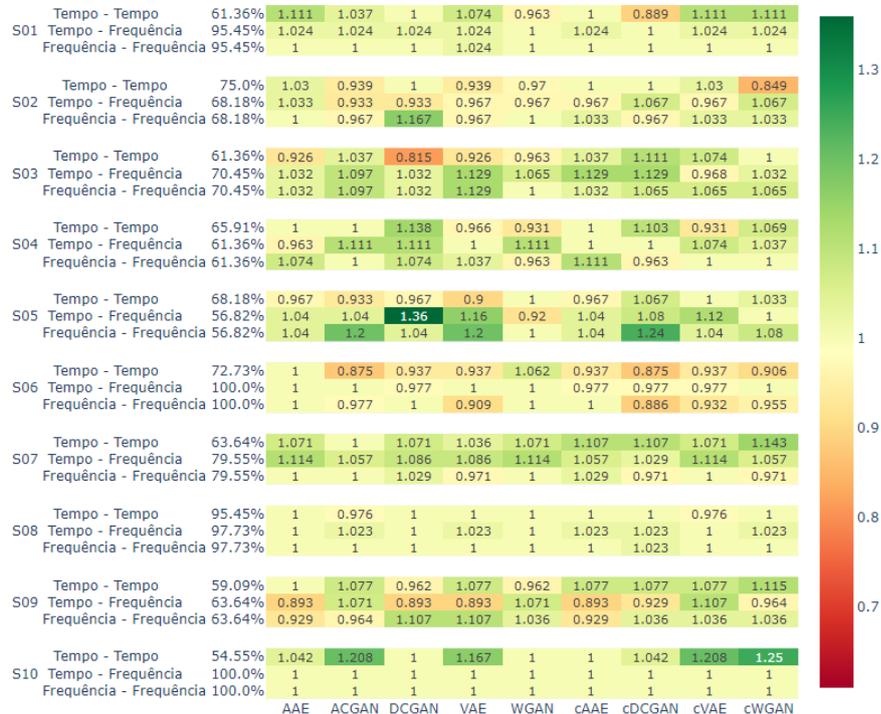


Figura 6.11 – Ganhos de desempenho proporcionados pelo uso dos EEGs de 2 segundos sintetizados por cada modelo generativo quando comparamos aos maiores desempenhos sem *data augmentation*.

Para os classificadores que processam EEGs-SSVEP com 3 segundos de comprimento (Figura 6.12), observamos um ganho de performance máxima de 39,1% para o

participante S09 no domínio da frequência com EEGs-SSVEP artificiais sintetizados na frequência pelo *autoencoder* variacional, enquanto a maior perda de desempenho ocorreu no domínio do tempo com o uso dos EEGs artificiais do usuário S04 sintetizados pela DCGAN. Quando comparamos os desempenhos dos classificadores treinados com dados artificiais aos máximos desempenhos sem *data augmentation*, nota-se uma maior dificuldade dos classificadores em alcançar performances superiores, como pode ser observado para os participantes S03, S08 e S09 quando o problema é tratado no domínio do tempo e para os participantes S02, S05, S06 e S07 quando o problema é abordado no domínio da frequência, em que ou nenhum ou apenas um modelo foi capaz de promover ganhos almejados.



Figura 6.12 – Ganhos de desempenho proporcionados pelo uso dos EEGs de 3 segundos sintetizados por cada modelo generativo quando comparamos aos maiores desempenhos sem *data augmentation*.

Para os classificadores que processam EEGs-SSVEP com 4 segundos de comprimento (Figura 6.13), observamos um ganho máximo de desempenho de 24%, que ocorreu para o participante S05 no domínio do tempo, com EEGs sintetizados no tempo pela cWGAN. Esse percentual é inferior aos obtidos para as janelas de menor duração (2 s e 3 s). Além disso, a maior queda de desempenho acontece justamente para EEGs de 4 s, novamente para o usuário S04 com os EEGs gerados pela DCGAN. As 3 visões apresentadas nesta seção reforçam que, embora seja possível obter melhorias na performance dos classificadores da BCI utilizando EEGs artificiais tanto no domínio do tempo quanto no domínio da frequência por arquiteturas de modelos generativos de propósito geral, não

há uma garantia de que os EEGs sintetizados por todos os modelos proporcionarão os ganhos de desempenho desejados, sendo recomendado investigar diferentes técnicas.

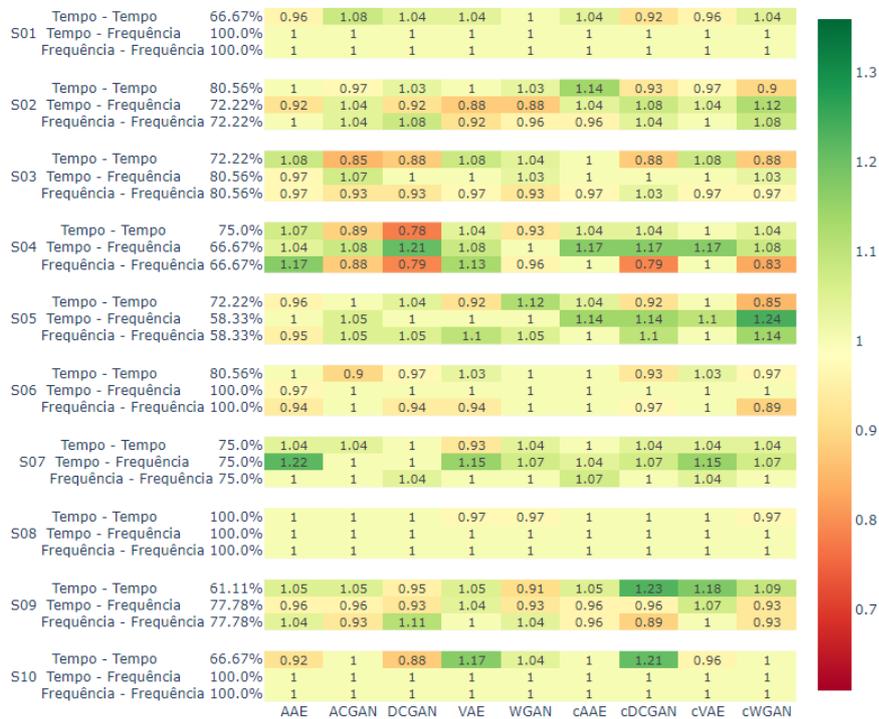


Figura 6.13 – Ganhos de desempenho proporcionados pelo uso dos EEGs de 4 segundos sintetizados por cada modelo generativo quando comparamos aos maiores desempenhos sem *data augmentation*.

Por fim, a Figura 6.14 apresenta uma média global dos desempenhos dos 10 usuários investigados para cada domínio, comprimento de EEGs e uso ou não de dados sintéticos no treinamento dos classificadores, com os valores apresentados extraídos das linhas que indicam os valores médios para cada tabela de desempenhos 6.1, 6.2, 6.3, 6.4 e 6.5. A análise dos resultados apresentados na Figura 6.14 mostra que o modelo convolucional CNN-1D possui os maiores desempenhos médios em todas as abordagens estudadas, destacando-se como a escolha mais assertiva para a tarefa de classificação em BCIs-SSVEP que processam ou sinais brutos de EEGs ou os espectros desses sinais. Comparando as abordagens investigadas, podemos observar que o uso de EEGs de 2 e 3 segundos no domínio da frequência e sem *data augmentation* supera o desempenho médio da abordagem com dados artificiais no domínio do tempo para EEGs de 4 segundos, indicando a possibilidade de utilizar nas BCIs classificadores que processam EEGs mais rápidos, mantendo ou até superando o desempenho dos EEGs de 4 s de duração no tempo. Além disso, ao comparar as abordagens com *data augmentation* na frequência utilizando EEGs artificiais sintetizados no tempo, observa-se um desempenho médio superior em relação à abordagem em que os EEGs são sintetizados diretamente no domínio da frequência, mostrando que mesmo o domínio da frequência sendo o mais adequado para o problema

de classificação abordado, a etapa de treinamento dos modelos generativos e posterior geração dos dados artificiais no domínio do tempo se mostrou mais efetivo do que treinar as mesmas técnicas generativas para produzir os espectros dos EEGs. Assim, o cenário ideal para obter os melhores desempenhos em sistemas BCIs é aquele que utiliza espectros de EEGs de 4 segundos de comprimento e classificadores convolucionais CNN-1D treinados com bases de dados aumentadas constituídas pelos espectros dos EEGs reais do usuário e por espectros de EEGs-SSVEP artificiais que são sintetizados no domínio do tempo.

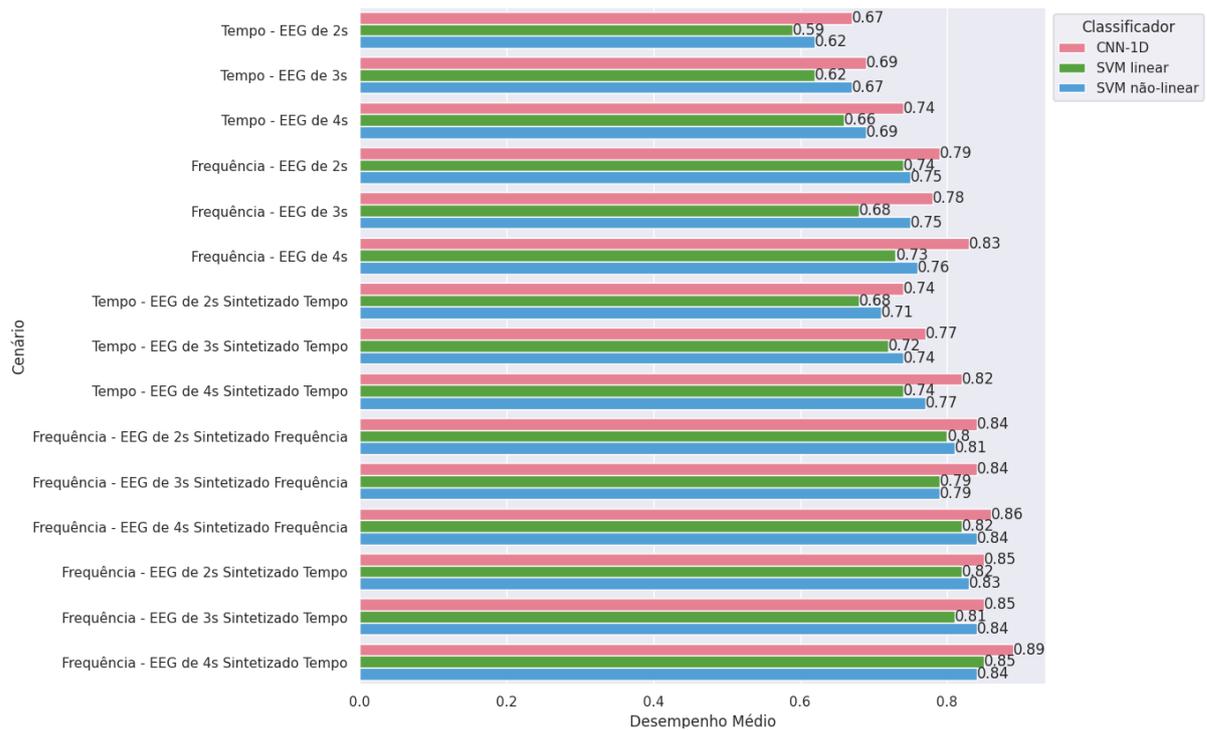
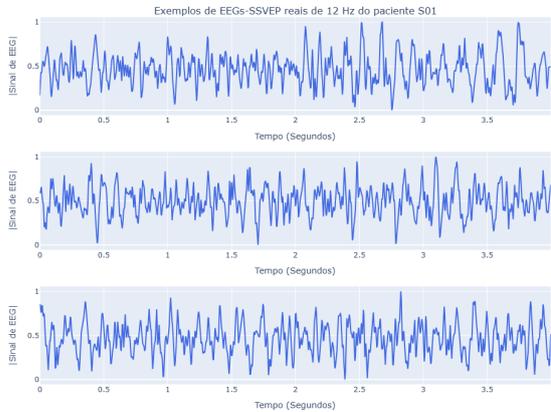


Figura 6.14 – Comparativo dos desempenhos médios dos classificadores SVM linear, SVM não-linear e CNN-1D sobre os 10 participantes da base de dados em função das abordagens sem e com *data augmentation* investigadas.

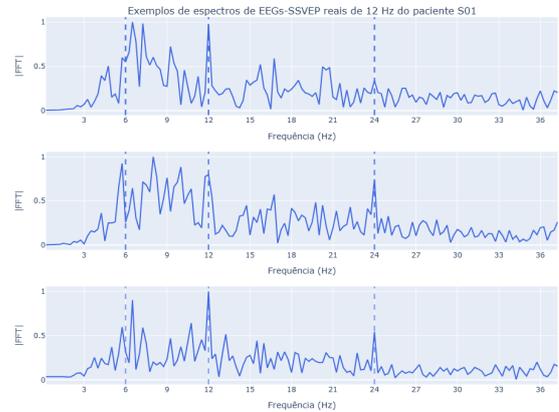
6.6 Exemplos de amostras de EEGs-SSVEP artificiais

Nesta seção apresentamos exemplos de EEGs-SSVEP artificiais de 4 s que foram sintetizados por alguns dos modelos generativos obtidos neste trabalho, a fim de reforçar a capacidade desses modelos genéricos em sintetizar EEGs-SSVEP coerentes, principalmente quando analisamos os espectros desses EEGs. A Figura 6.15 apresenta algumas amostras reais de EEGs-SSVEP, para referência, tanto no domínio do tempo quanto no domínio da frequência para as frequências de evocação iguais a 12 Hz e 15 Hz capturadas do participante S01. Podemos observar nos espectros dos EEGs-SSVEP relacionados ao estímulo de 12 Hz, que surgem picos em 12 Hz e 24 Hz, além de uma concentração de energia em torno da frequência de 6 Hz. Já para os espectros correspon-

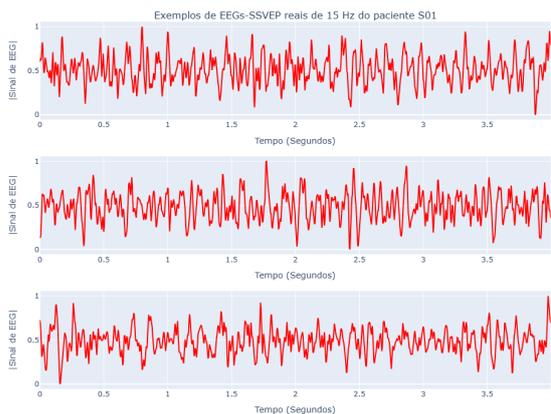
dentos aos estímulos de 15 Hz do mesmo usuário, aparecem picos nítidos na frequência de evocação, 15 Hz.



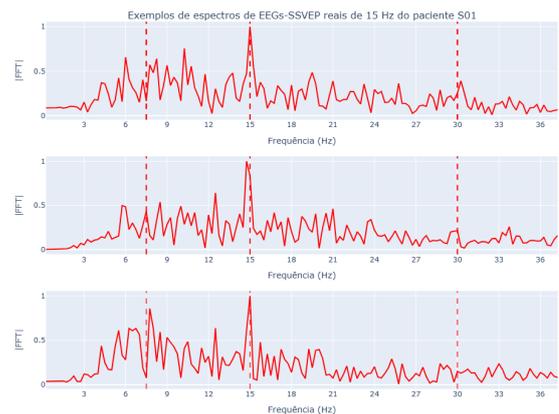
(a) EEGs-SSVEP de 12 Hz reais do usuário S01.



(b) Espectros dos EEGs-SSVEP de 12 Hz reais.



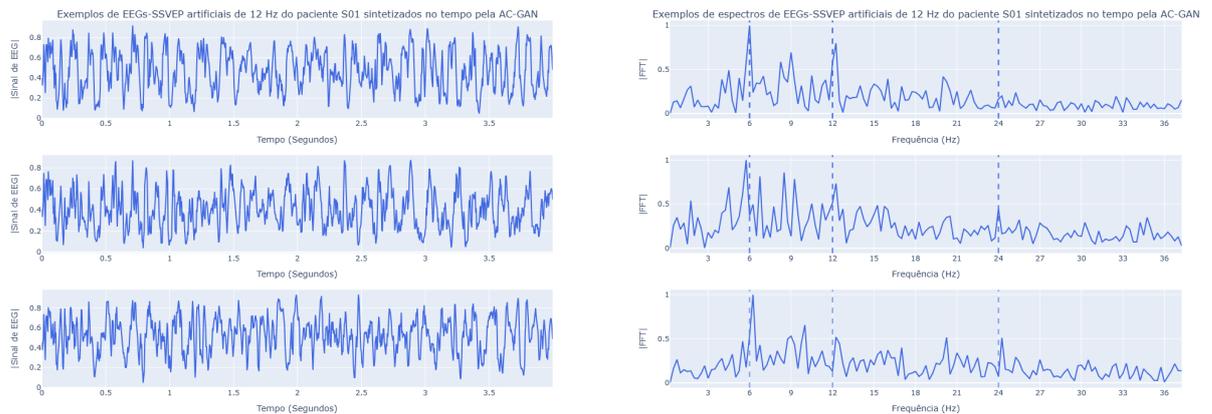
(c) EEGs-SSVEP de 15 Hz reais do usuário S01.



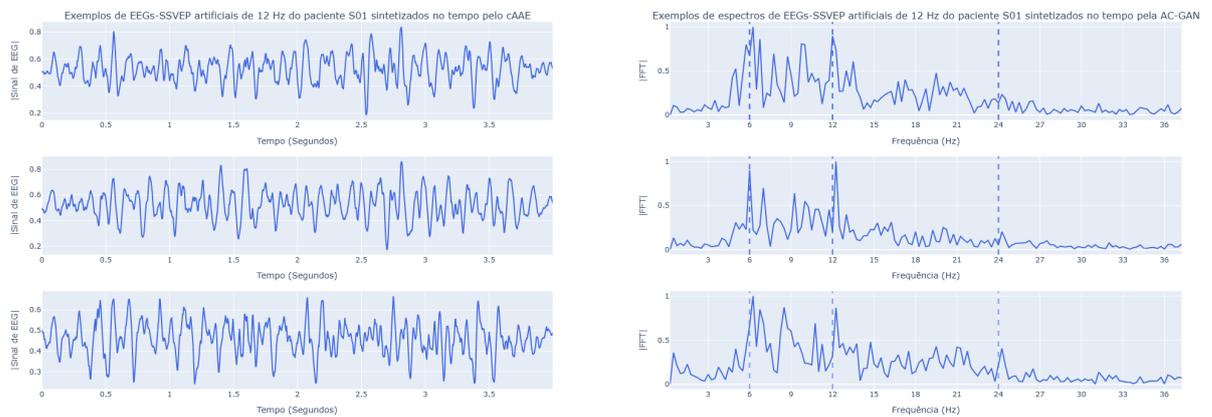
(d) Espectros dos EEGs-SSVEP de 15 Hz reais.

Figura 6.15 – Exemplos de amostras reais do paciente S01 nos domínios do tempo e da frequência para as frequências de evocação de 12 Hz e 15 Hz.

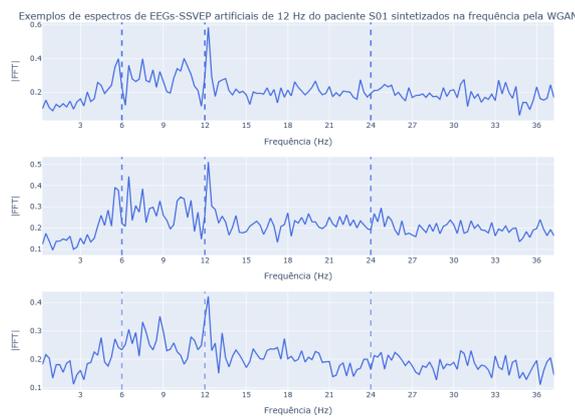
A Figura 6.16 apresenta exemplos de amostras artificiais de 12 Hz sintetizadas pelos modelos generativos treinados com as amostras reais do usuário S01. A Figura 6.16a apresenta três EEGs-SSVEP de 12 Hz sintetizados pelo modelo generativo AC-GAN no domínio do tempo, enquanto a Figura 6.16b apresenta os respectivos espectros desses sinais. Assim como os sinais reais, as amostras de 12 Hz sintetizadas pela AC-GAN apresentam picos na frequência de evocação e também na frequência de 6 Hz, mas os picos na frequência de 24 Hz, que estão bem presentes nos sinais reais, foram gerados mas com uma amplitude bem menor. Outra diferença que pode ser observada entre os EEGs-SSVEP de 12 Hz sintetizados pela AC-GAN em relação aos sinais reais é que as amostras artificiais apresentam componentes significativas para frequências inferiores à 3 Hz.



(a) EEGs-SSVEP de 12 Hz sintetizados no tempo pela ACGAN obtida para o usuário S01. (b) Espectros dos EEGs-SSVEP de 12 Hz sintetizados no tempo pela ACGAN obtida para o usuário S01.

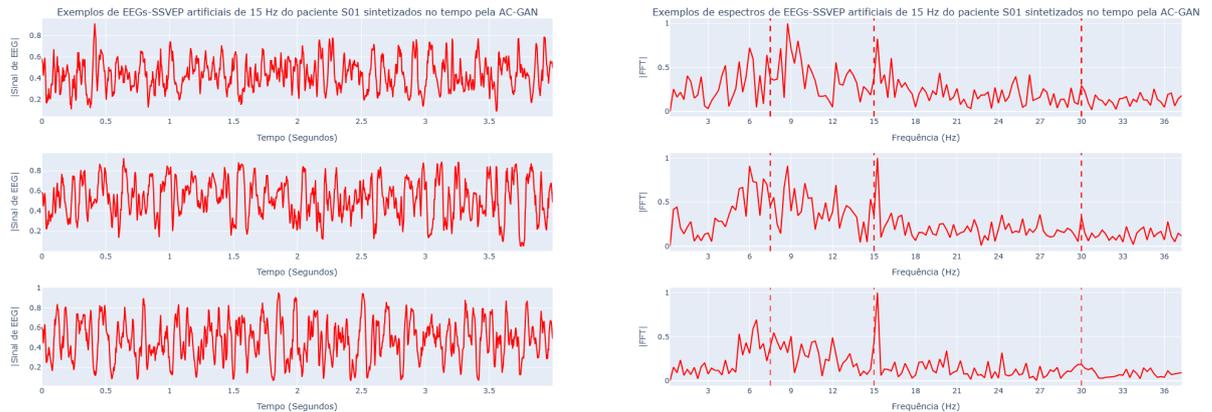


(c) EEGs-SSVEP de 12 Hz sintetizados no tempo pelo cAAE obtido para o usuário S01. (d) Espectros dos EEGs-SSVEP de 12 Hz sintetizados no tempo pelo cAAE obtido para o usuário S01.

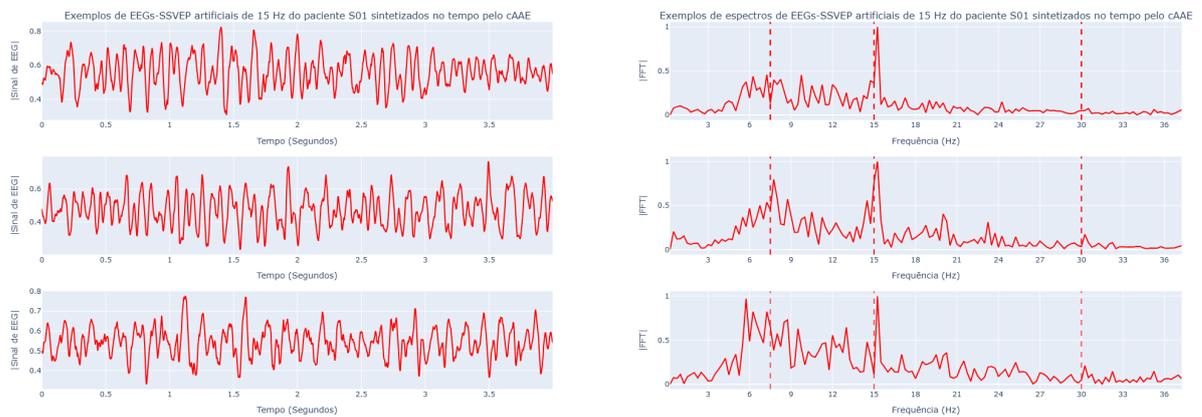


(e) Espectros dos EEGs-SSVEP de 12 Hz sintetizados na frequência pela WGAN obtida para o usuário S01.

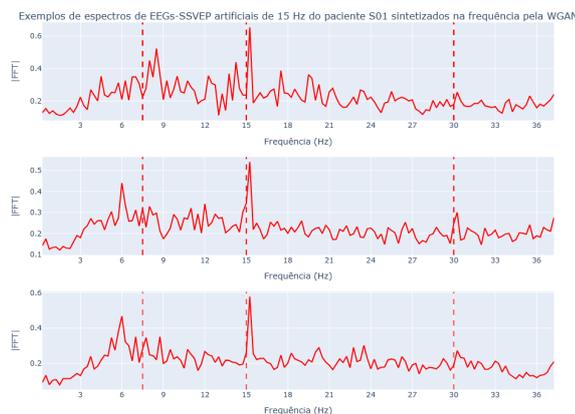
Figura 6.16 – Exemplos de amostras artificiais do paciente S01 nos domínios do tempo e da frequência para a frequência de evocação de 12 Hz.



(a) EEGs-SSVEP de 15 Hz sintetizados no tempo pela ACGAN obtida para o usuário S01. (b) Espectros dos EEGs-SSVEP de 15 Hz sintetizados no tempo pela ACGAN obtida para o usuário S01.



(c) EEGs-SSVEP de 15 Hz sintetizados no tempo pelo cAAE obtido para o usuário S01. (d) Espectros dos EEGs-SSVEP de 15 Hz sintetizados no tempo pelo cAAE obtido para o usuário S01.



(e) Espectros dos EEGs-SSVEP de 15 Hz sintetizados na frequência pela WGAN obtida para o usuário S01.

Figura 6.17 – Exemplos de amostras artificiais do paciente S01 nos domínios do tempo e da frequência para a frequência de evocação de 15 Hz.

A Figura 6.16c exibe três EEGs-SSVEP artificiais de 12 Hz sintetizados pelo modelo generativo cAAE no domínio do tempo, ao lado de seus respectivos espectros (Figura 6.16d). Assim como os sinais gerados pelo modelo anterior, as amostras de 12 Hz sintetizadas pela cAAE apresentam picos na frequência de evocação e na frequência de 6 Hz, e os picos na frequência de 24 Hz também foram gerados com amplitude reduzida. Com relação aos dados reais, o cAAE conseguiu produzir EEGs-SSVEP de 12 Hz com as componentes espectrais menores que 3 Hz bem atenuadas, mas para as frequências superiores a 24 Hz os sinais apresentaram componentes praticamente iguais a zero, enquanto nos sinais reais ainda existia uma certa variação de amplitude nessa faixa de frequências.

A Figura 6.16e apresenta 3 EEGs-SSVEP artificiais de 12 Hz sintetizados pelo modelo generativo WGAN diretamente no domínio da frequência. Assim como nos casos anteriores, as amostras artificiais apresentaram o pico na frequência de evocação. Contudo, neste caso o modelo apresentou dificuldade em produzir o pico presente em 6 Hz e, além disso, podemos observar também uma dificuldade em gerar as variações de maior amplitude entre picos e vales no espectro, que são menores no espectro sintetizado quando comparamos com os espectros reais.

Por fim, a Figura 6.17 apresenta exemplos de amostras artificiais de 15 Hz sintetizadas pelos mesmos modelos generativos obtidos para o usuário S01. Todas as observações feitas para as características dos EEGs-SSVEP artificiais de 12 Hz se estendem para as amostras sintéticas de 15 Hz, com a diferença de que para este caso os picos de amplitude apareceram na frequência de evocação de 15 Hz.

Além do participante S01, também geramos amostras de EEGs-SSVEP artificiais para o participante S10 da base de dados. A Figura 6.18 apresenta algumas amostras reais de EEGs-SSVEP, para referência, tanto no domínio do tempo quanto no domínio da frequência para as frequências de evocação iguais a 12 Hz e 15 Hz. Diferente das respostas geradas pelo primeiro usuário, o participante S10 apresenta como característica picos bem acentuados nas frequências múltiplas das frequências de evocação, isto é, em 30 Hz para as amostras correspondentes à frequência de evocação de 15 Hz, e em 24 Hz para os sinais correspondentes à frequência de evocação de 12 Hz.

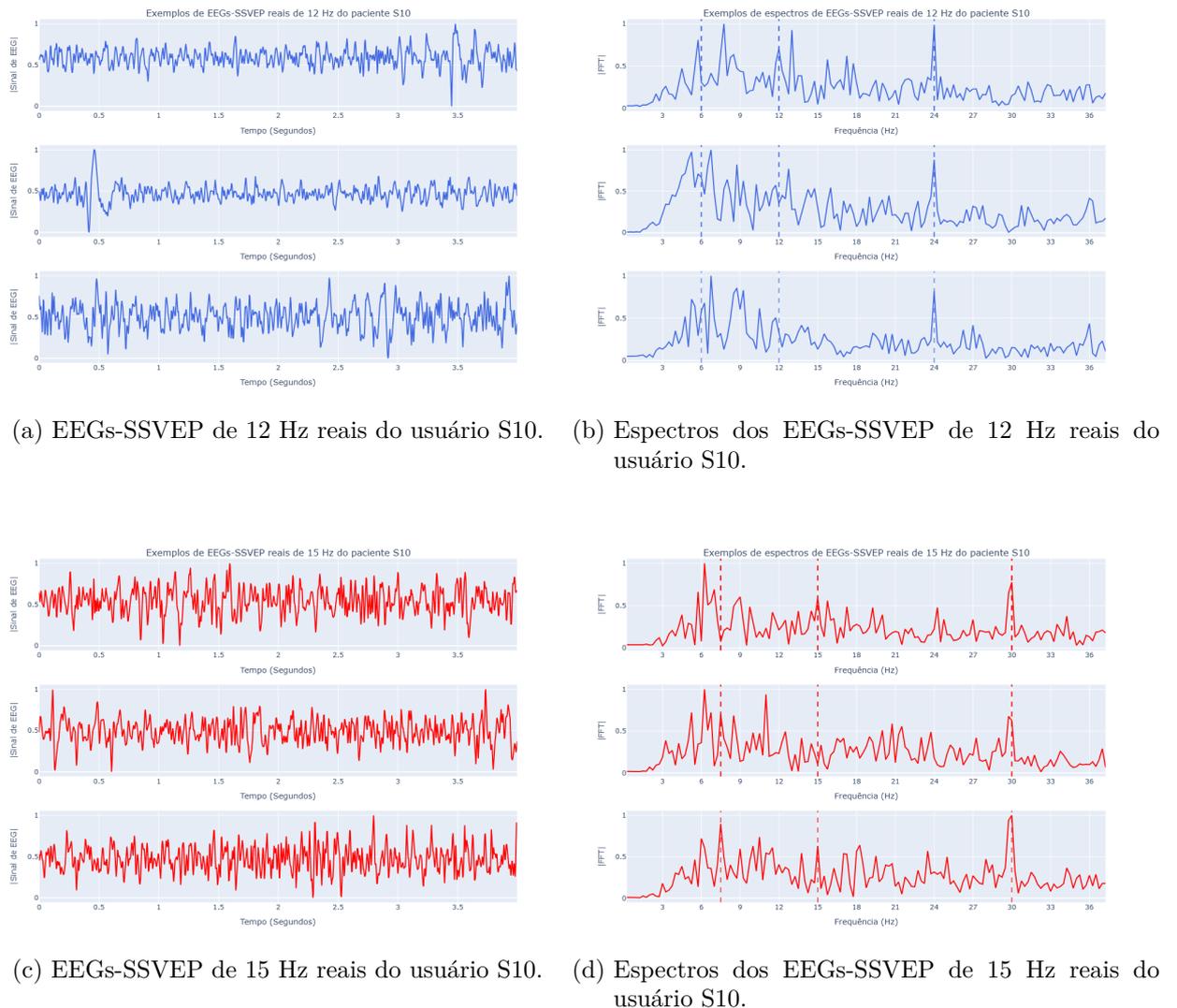
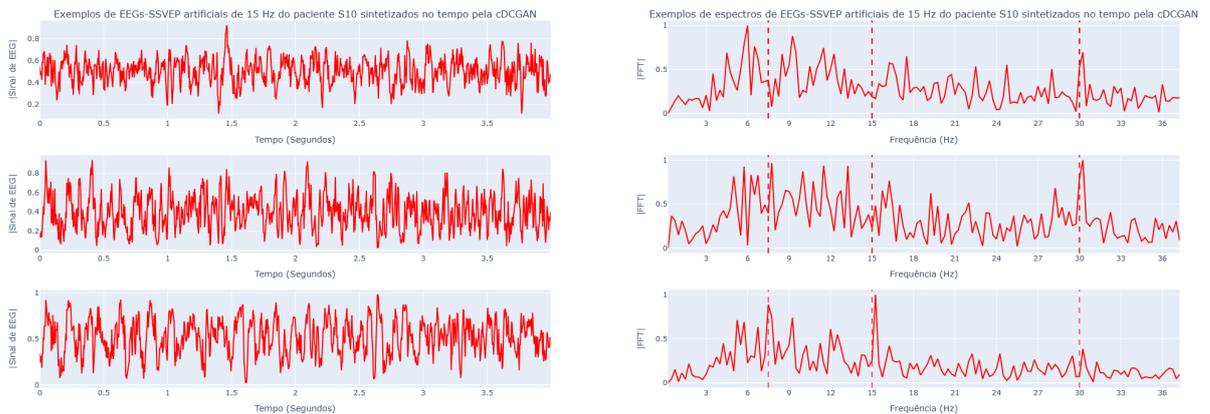
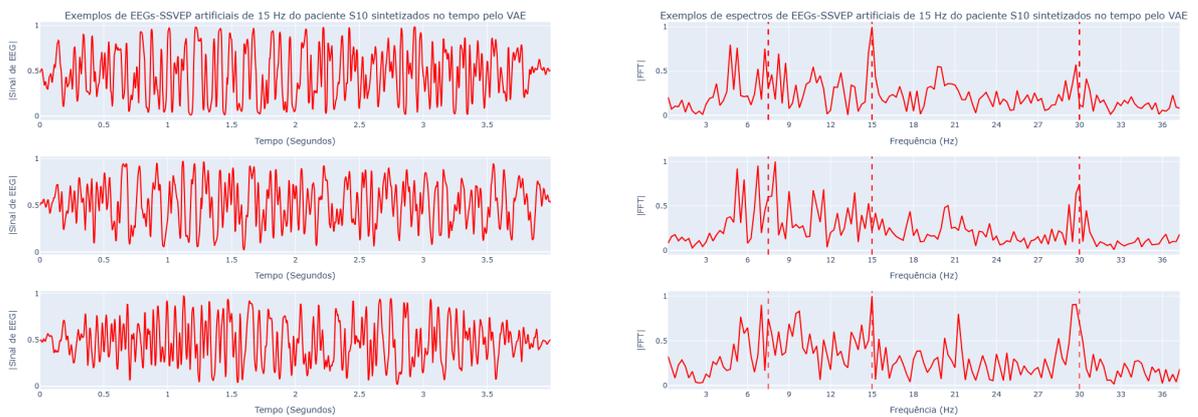


Figura 6.18 – Exemplos de amostras reais do paciente S10 nos domínios do tempo e da frequência para as frequências de evocação de 12 Hz e 15 Hz.

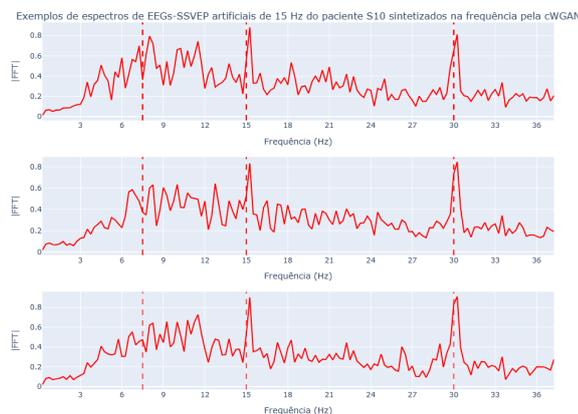
A Figura 6.19 apresenta exemplos de amostras artificiais de 15 Hz sintetizadas pelos modelos generativos treinados com as amostras reais do usuário S10. A Figura 6.19a apresenta três EEGs-SSVEP artificiais de 15 Hz sintetizados pelo modelo generativo cDC-GAN no domínio do tempo, enquanto a Figura 6.19b apresenta os respectivos espectros na frequência desses sinais. Podemos observar que os dois primeiros sinais sintetizados apresentam o pico de amplitude na frequência de 30 Hz assim como as amostras reais, enquanto o terceiro exemplo apresenta picos na frequência de evocação de 15 Hz e na frequência sub-múltipla de 7,5 Hz, que pode ser devido à presença de sinais reais com esse comportamento no conjunto de treinamento.



(a) EEGs-SSVEP de 15 Hz sintetizados no tempo pela (b) Espectros dos EEGs-SSVEP de 15 Hz sintetizados no tempo pela cDCGAN obtida para o usuário S10.

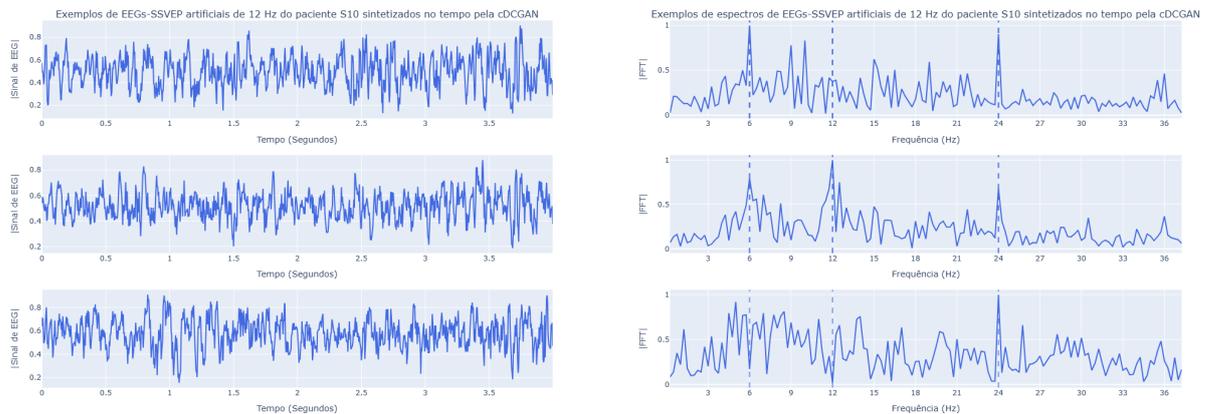


(c) EEGs-SSVEP de 15 Hz sintetizados no tempo pelo (d) Espectros dos EEGs-SSVEP de 15 Hz sintetizados no tempo pelo VAE obtido para o usuário S10.

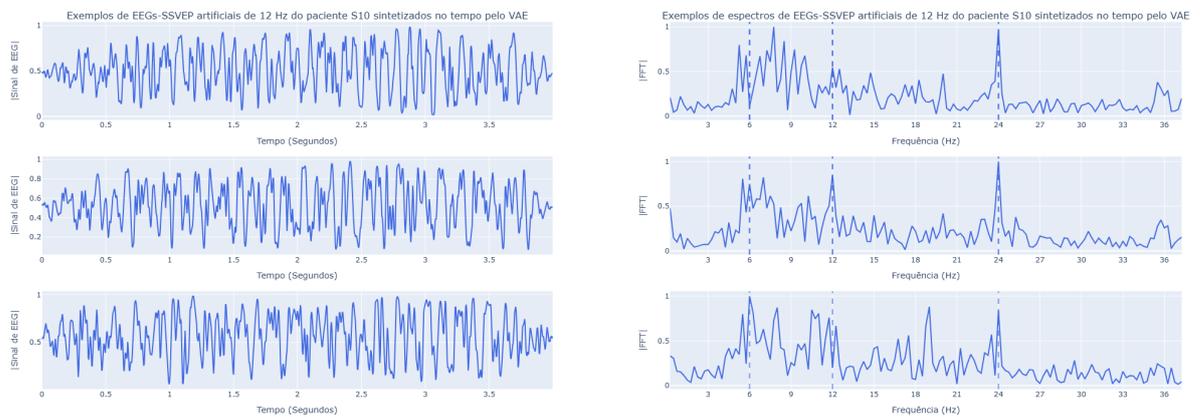


(e) Espectros dos EEGs-SSVEP de 15 Hz sintetizados na frequência pela cWGAN obtida para o usuário S10.

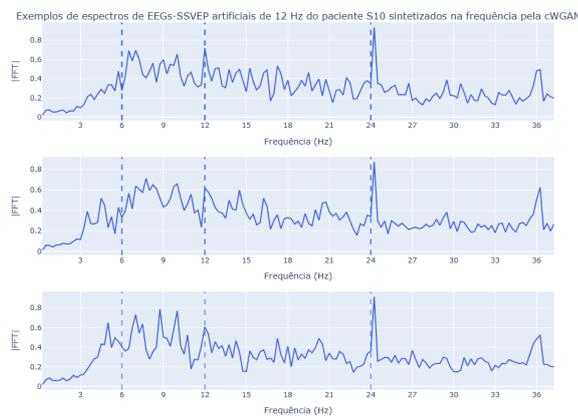
Figura 6.19 – Exemplos de amostras artificiais do paciente S10 nos domínios do tempo e da frequência para a frequência de evocação de 15 Hz.



(a) EEGs-SSVEP de 12 Hz sintetizados no tempo pela cDCGAN obtida para o usuário S10. (b) Espectros dos EEGs-SSVEP de 12 Hz sintetizados no tempo pela cDCGAN obtida para o usuário S10.



(c) EEGs-SSVEP de 12 Hz sintetizados no tempo pelo VAE obtido para o usuário S10. (d) Espectros dos EEGs-SSVEP de 12 Hz sintetizados no tempo pelo VAE obtido para o usuário S10.



(e) Espectros dos EEGs-SSVEP de 12 Hz sintetizados na frequência pela cWGAN obtida para o usuário S10.

Figura 6.20 – Exemplos de amostras artificiais do paciente S10 nos domínios do tempo e da frequência para a frequência de evocação de 12 Hz.

Por sua vez, a Figura 6.19c mostra três EEGs-SSVEP gerados pelo VAE no domínio do tempo para uma frequência de estímulo de 15 Hz, juntamente com os respectivos espectros (Figura 6.19d). Como podemos observar, assim como o modelo anterior, os EEGs sintetizados apresentam pico na frequência de 30 Hz e na frequência de evocação de 15 Hz, indicando que o EEG-SSVEP no tempo possui essas frequências que compõem as respostas geradas pelo usuário. Contudo, quando observamos o próprio EEG no domínio do tempo, conseguimos verificar visualmente que ele apresenta um perfil muito diferente dos EEGs-SSVEP reais de 15 Hz do usuário S10.

Já a Figura 6.19e exibe três espectros de EEGs-SSVEP para 15 Hz que foram diretamente sintetizados na frequência pelo modelo cWGAN. Da mesma forma que para os modelos anteriores, a cWGAN foi capaz de aprender a gerar os espectros com os picos nas frequências de 15 Hz e 30 Hz, e assim como ocorreu para as amostras artificiais do usuário S01 sintetizadas pela WGAN no domínio da frequência, o modelo condicionado também teve dificuldades em reproduzir as variações mais abruptas de amplitude no espectro fora das frequências múltiplas da frequência de evocação, o que pode indicar uma limitação na resposta desse modelo.

Por fim, a Figura 6.20 apresenta exemplos de amostras artificiais de 12 Hz sintetizadas pelos mesmos modelos generativos para o usuário S10. Todas as observações feitas anteriormente se estendem para as amostras sintéticas de 12 Hz, com a diferença de que para este caso os picos de amplitude apareceram nas frequências de 12 Hz, 24 Hz e também em 36 Hz.

7 Conclusão

Este trabalho de mestrado apresentou um amplo estudo sobre a aplicabilidade de modelos generativos de propósito geral para a síntese de EEGs em sistemas BCI-SSVEP, investigando não apenas a abordagem no domínio do tempo, como realizado pelos trabalhos anteriores e discutidos na seção de revisão bibliográfica, mas também no domínio da frequência com o uso de espectros artificiais de EEGs-SSVEP que foram ou sintetizados diretamente na frequência ou a partir do cálculo do módulo da DFT dos EEGs-SSVEP sintéticos gerados no domínio do tempo. Para permitir a comparação entre as abordagens estudadas, todos os modelos generativos e os classificadores da BCI foram obtidos com o uso dos dados reais dos mesmos conjuntos de treinamento, correspondentes às sessões 1, 2, 3, 4, 5 e 6 da base de dados, enquanto as amostras das sessões 7 e 8 foram utilizadas para a avaliação dos modelos, com os desempenhos de validação sendo apresentados na seção de resultados e discussões.

Os resultados mostram que, mesmo sem o uso de dados sintéticos para o treinamento dos classificadores, o tratamento do problema de classificação no domínio da frequência tende a apresentar melhores desempenhos que os mesmos classificadores no domínio do tempo, principalmente quando os espectros dos EEGs dos usuários apresentam picos nítidos nas frequências múltiplas e sub-múltiplas às frequências de estímulo. No entanto, quando os espectros não apresentam esse tipo de comportamento, o uso exclusivo dos dados reais no domínio da frequência pode resultar em degradação na performance dos modelos. Com isso, a análise dos espectros dos EEGs-SSVEP logo após a coleta dos dados é importante, pois se os espectros pertencentes ao paciente forem bem comportados e com picos nítidos nas frequências de estímulo, a abordagem no domínio da frequência pode se mostrar mais adequada e eficiente.

Com relação às abordagens com *data augmentation* seja no domínio do tempo seja no domínio da frequência, verificamos que em 80,7% das comparações houve ganho de desempenho resultante do uso de dados artificiais, enquanto que em 18,8% os desempenhos se mantiveram inalterados e em apenas 0,5% (1 caso) ocorreu uma redução na performance do modelo devido à incorporação de dados sintéticos em seu treinamento. Outro ponto que podemos concluir é que embora consigamos obter ganhos de desempenho com os EEGs-SSVEP sintetizados por modelos generativos de propósito geral, nem todos os modelos são capazes de promover esses ganhos e nem em todos os cenários, o que reforça a necessidade de avaliar um número significativo de arquiteturas de modelos generativos até a obtenção do modelo cujos dados artificiais promovam aumento na performance do classificador

da BCI. Ainda com relação aos modelos generativos, nosso estudo indicou que não há uma técnica específica que predomina sobre as demais, o que, mais uma vez, enfatiza a importância de se estudar variados modelos.

As duas abordagens no domínio da frequência com *data augmentation* apresentaram os maiores desempenhos médios para todos os classificadores e comprimentos de EEG, indicando que a abordagem no domínio da frequência que utiliza espectros de EEGs tende a ser mais adequado que a abordagem no domínio do tempo que utiliza apenas os EEGs-brutos diretamente como entradas dos classificadores da BCI. Apesar disso, observou-se desempenhos superiores quando os EEGs artificiais foram sintetizados no domínio do tempo e depois transformados para o domínio da frequência. Outro resultado interessante é que mesmo com os classificadores CNN-1D apresentando os maiores ganhos quando são projetados para processar os EEGs de 4 segundos de duração, a abordagem no domínio da frequência com dados artificiais diminui consideravelmente a diferença de desempenho desses classificadores com relação ao comprimento do EEG, permitindo assim o uso de EEGs de duração reduzida, como os de 2 segundos, para muitos dos usuários sem que ocorra queda na performance ou que a perda de desempenho seja muito pequena, possivelmente não provocando grandes impactos no uso da BCI.

Por fim, os resultados nos permite concluir que dentre os cenários investigados no escopo deste trabalho, o domínio da frequência se mostra como o mais adequado para o problema de classificação em BCIs, principalmente quando os EEGs do paciente apresentam espectros bem comportados, que a escolha mais segura para o modelo de classificação é a rede convolucional CNN-1D, e que, caso seja essencial ter o máximo desempenho de classificação, deve-se utilizar janelas de 4 segundos de duração e explorar dados artificiais que foram gerados no domínio do tempo e depois convertidos para o domínio da frequência.

Como perspectivas futuras, pode-se expandir a análise de geração e uso de EEGs-SSVEP artificiais explorando eletrodos adicionais para aproveitar o caráter espacial da tarefa, pois neste trabalho utilizamos os EEGs capturados exclusivamente pelo eletrodo Oz como dado de entrada dos modelos. Além disso, uma abordagem *few-shot* poderia ser investigada aplicando uma pequena porção de sinais reais no domínio da frequência para ajustar um classificador treinado apenas com amostras sintéticas, com o objetivo de estabelecer uma BCI-SSVEP invariante ao sujeito. Por fim, dado que os modelos probabilísticos de difusão para remoção de ruído (DDPM) têm apresentado resultados impressionantes em tarefas de geração de imagens, superando modelos baseados em GANs em alguns casos, eles também poderiam ser avaliados no contexto de BCI-SSVEP.

Referências

- AGGARWAL, S.; CHUGH, N. Review of machine learning techniques for eeg based brain computer interface. *Archives of Computational Methods in Engineering*, v. 29, 01 2022. Citado na página 23.
- ARJOVSKY, M.; CHINTALA, S.; BOTTOU, L. Wasserstein generative adversarial networks. In: PRECUP, D.; TEH, Y. W. (Ed.). *Proceedings of the 34th International Conference on Machine Learning*. [S.l.: s.n.], 2017. (Proceedings of Machine Learning Research, v. 70), p. 214–223. Citado 2 vezes nas páginas 42 e 44.
- ASANZA, V.; PELAEZ, C.; LOAYZA, F.; LORENTE, L.; PELUFFO-ORDÓÑEZ, D. Identification of lower-limb motor tasks via brain–computer interfaces: A topical overview. *Sensors*, v. 22, p. 2028, 03 2022. Citado 2 vezes nas páginas 7 e 24.
- AZNAN, N.; ABARGHOUEI, A. A.; BONNER, S.; CONNOLLY, J.; BRECKON, T. On the classification of ssvep-based dry-eeg signals via convolutional neural networks. 2018. Citado 5 vezes nas páginas 23, 37, 56, 76 e 99.
- AZNAN, N.; ABARGHOUEI, A. A.; BONNER, S.; CONNOLLY, J.; BRECKON, T. Leveraging synthetic subject invariant eeg signals for zero calibration bci. 2020. Citado 7 vezes nas páginas 14, 19, 50, 51, 54, 58 e 67.
- AZNAN, N.; ATAPOUR-ABARGHOUEI, A.; BONNER S., J. D.; MOUBAYED, N. A.; BRECKON, T. Simulating brain signals: Creating synthetic eeg data via neural-based generative models for improved ssvep classification. *2019 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2019. Citado 7 vezes nas páginas 14, 19, 50, 54, 58, 59 e 67.
- BELLMAN, R.; CORPORATION, R.; COLLECTION, K. M. R. *Dynamic Programming*. Princeton University Press, 1957. (Rand Corporation research study). ISBN 9780691079516. Disponível em: <<https://books.google.com.br/books?id=wdtoPwAACAAJ>>. Citado na página 50.
- BERTRAND, O.; PERRIN, F.; PERNIER, J. A theoretical justification of the average reference in topographic evoked potential studies. *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, v. 62, n. 6, p. 462–464, 1985. ISSN 0168-5597. Citado na página 28.
- BISHOP, C. M. *Neural Networks for Pattern Recognition*. USA: Oxford University Press, Inc., 1995. ISBN 0198538642. Citado na página 29.
- BISHOP, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. 1. ed. [S.l.]: Springer, 2007. ISBN 0387310738. Citado 4 vezes nas páginas 18, 29, 30 e 51.
- BROWNLEE, J. *How to Develop an Auxiliary Classifier GAN (AC-GAN) From Scratch with Keras*. 2021. Accessed on 10 13, 2024. Disponível em: <<https://machinelearningmastery.com/how-to-develop-an-auxiliary-classifier-gan-ac-gan-from-scratch-with-keras/>>. Citado na página 48.

- CARVALHO, S. N. *Contribuições ao desenvolvimento de interfaces cérebro-computador baseadas em potenciais evocados visualmente em regime estacionário*. Tese (Doutorado) — Faculdade de Engenharia Elétrica e de Computação, UNICAMP, 2016. Citado 9 vezes nas páginas 18, 20, 22, 23, 24, 25, 27, 29 e 51.
- CARVALHO, S. N.; COSTA, T. B.; URIBE, L. F.; SORIANO, D. C.; YARED, G. F.; CORADINE, L. C.; ATTUX, R. Comparative analysis of strategies for feature extraction and classification in ssvep bcis. *Biomedical Signal Processing and Control*, v. 21, p. 34–42, 2015. ISSN 1746-8094. Citado na página 33.
- CORTES, C.; VAPNIK, V. Support-vector networks. *Machine learning*, Springer, v. 20, n. 3, p. 273–297, 1995. Citado na página 31.
- COVER, T. M.; THOMAS, J. A. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience, 2006. ISBN 0471241954. Citado na página 30.
- COYLE, S.; WARD, T.; MARKHAM, C.; G, M. On the suitability of near-infrared (nir) systems for next-generation brain-computer interfaces. *Neurocomputing*, p. 815–22, 08 2004. Citado na página 23.
- CURRAN, E.; SYKACEK, P.; STOKES, M.; ROBERTS, S.; PENNY, W.; JOHNSRUDE, I.; AM., O. Cognitive tasks for driving a brain-computer interfacing system: a pilot study. *IEEE transactions on neural systems and rehabilitation engineering*, IEEE, v. 12, p. 48–54, 2004. Citado na página 24.
- DOERSCH, C. *Tutorial on Variational Autoencoders*. 2016. Citado 2 vezes nas páginas 38 e 39.
- DONCHIN, E.; SMITH, D. The contingent negative variation and the late positive wave of the average evoked potential. *Electroencephalography and clinical Neurophysiology*, Elsevier, v. 29, p. 201–3, 1970. Citado na página 25.
- DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern Classification*. 2. ed. New York: Wiley, 2001. ISBN 978-0-471-05669-0. Citado 4 vezes nas páginas 18, 29, 30 e 51.
- GERON, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. [S.l.]: O’Reilly Media, 2022. ISBN 9781098125974. Citado 4 vezes nas páginas 31, 32, 36 e 50.
- GOODFELLOW, I.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. Generative adversarial networks. *Advances in Neural Information Processing Systems*, v. 3, 2014. Citado 4 vezes nas páginas 19, 38, 42 e 51.
- GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. *Journal of Machine Learning Research*, v. 3, p. 1157–1182, 2003. Citado 3 vezes nas páginas 29, 30 e 51.
- HO, J.; JAIN, A.; ABBEEL, P. Denoising diffusion probabilistic models. In: LAROCHELLE, H.; RANZATO, M.; HADSELL, R.; BALCAN, M.; LIN, H. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2020. v. 33, p. 6840–6851. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf>. Citado na página 51.

- IOFFE, S.; SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. Citado na página 43.
- J LI Y, S. W. Q. A semisupervised support vector machines algorithm for bci systems. *Computational intelligence and neuroscience*, 2007. Citado na página 32.
- JASPER, H. H. Ten-twenty electrode system of the international federation. *Electroencephalogr Clin Neurophysiol*, v. 10, p. 371–375, 1958. Citado na página 23.
- KAPER, M.; MEINICKE, P.; GROSSEKATHOEFER, U.; LINGNER, T.; RITTER, H. Bci competition 2003-data set iib: support vector machines for the p300 speller paradigm. *IEEE Transactions on biomedical Engineering*, IEEE, v. 51, n. 6, p. 1073–1076, 2004. Citado na página 33.
- KINGMA, D.; BA, J. Adam: A method for stochastic optimization. In: *International Conference on Learning Representations (ICLR)*. San Diego, CA, USA: [s.n.], 2015. Citado na página 76.
- KINGMA, D. P.; WELLING, M. Auto-encoding variational bayes. 2014. Citado 3 vezes nas páginas 18, 38 e 51.
- KLEM, G.; LÜDERS, H.; JASPER, H.; C, E. The ten-twenty electrode system of the international federation. *The International Federation of Clinical Neurophysiology*, 1999. Citado na página 23.
- KOSSALE, Y.; AIRAJ, M.; DAROUICHI, A. Mode collapse in generative adversarial networks: An overview. In: *2022 8th International Conference on Optimization and Applications (ICOA)*. [S.l.: s.n.], 2022. p. 1–6. Citado na página 47.
- KWON, J.; IM, C.-H. Novel signal-to-signal translation method based on stargan to generate artificial eeg for ssvep-based brain-computer interfaces. *Expert Systems with Applications*, v. 203, p. 117574, 2022. ISSN 0957-4174. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0957417422008880>>. Citado 7 vezes nas páginas 8, 14, 19, 51, 54, 64 e 65.
- LASHGARI, E.; LIANG, D.; MAOZ, U. Data augmentation for deep-learning-based electroencephalography. *Journal of Neuroscience Methods*, v. 346, 2020. Citado 5 vezes nas páginas 12, 19, 52, 53 e 54.
- LIMCHESING, T. J. C.; CHUA, A. E. C.; SHI, C. J. L.; BALDOVINO, R. G.; MUNSAYAC, F. E. T.; BUGTAL, N. T. A review on recent applications of eeg-based bci in wheelchairs and other assistive devices. In: *2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*. [S.l.: s.n.], 2021. p. 1–6. Citado na página 17.
- MACHADO, S.; ARAÚJO, F.; PAES, F.; VELASQUES, B.; CUNHA, M.; BUDDE, H.; BASILE, L.; ANGHINAH, R.; ARIAS-CARRIÓN, O.; CAGY, M.; PIEDADE, R.; GRAAF, T.; SACK, A.; RIBEIRO, P. Eeg-based brain-computer interfaces: An overview of basic concepts and clinical applications in neurorehabilitation. *Reviews in the neurosciences*, v. 21, p. 451–68, 01 2010. Citado na página 23.
- MAKHZANI, A.; SHLENS, J.; JAITLEY, N.; GOODFELLOW, I.; FREY, B. Adversarial autoencoders. 2016. Citado na página 46.

MARKAND, O. Artifacts. *Annual meeting of American Academy of Neurology.*, v. 1, p. 363–396, 1996. Citado na página 27.

NICOLAS-ALONSO, L.; GOMEZ-GIL, J. Brain computer interfaces, a review. *sensors (basel)*. p. 1211–79, 01 2012. Citado 2 vezes nas páginas 22 e 23.

ODENA, A.; OLAH, C.; SHLENS, J. Conditional image synthesis with auxiliary classifier gans. 2017. Citado 2 vezes nas páginas 42 e 45.

OPPENHEIM, A. V.; SCHAFER, R. W.; BUCK, J. R. *Discrete-Time Signal Processing*. Second. [S.l.]: Prentice-hall Englewood Cliffs, 1999. Citado 3 vezes nas páginas 18, 29 e 51.

PFURTSCHELLER, G.; NEUPER, C. Motor imagery activates primary sensorimotor area in humans. *Neuroscience letters*, Elsevier, v. 223, p. 65–68, 1997. Citado na página 24.

RADFORD, A.; METZ, L.; CHINTALA, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015. Citado 2 vezes nas páginas 42 e 43.

RAMADAN, R.; VASILAKOS, A. Brain computer interface: Control signals review. *Neurocomputing*, v. 223, 10 2016. Citado 2 vezes nas páginas 22 e 23.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. 2015. Citado na página 42.

TIELEMAN, T.; HINTON, G. *Lecture 6.5 - RMSProp, COURSERA: Neural Networks for Machine Learning*. [S.l.], 2012. Citado na página 80.

TODEREAN, R.; CHIUCHISAN, I. Application of support vector machine for the classification of sensorimotor rhythms in brain computer interface. p. 663–666, 2017. Citado na página 32.

VARGAS, V. G. *Filtragem espaço-temporal baseada no princípio MVDR aplicada a interfaces cérebro-computador sob o paradigma SSVEP*. 2021. Citado 3 vezes nas páginas 20, 25 e 27.

VăRBU, K.; MUHAMMAD, N.; MUHAMMAD, Y. Past, present, and future of eeg-based bci applications. *Sensors*, v. 22, n. 9, 2022. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/22/9/3331>>. Citado na página 23.

WOLPAW, J. R. e. a. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, v. 113, n. 6, p. 767–791, 2002. Citado 2 vezes nas páginas 17 e 20.

ZHANG, Y.; YANG, Q. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, v. 34, n. 12, p. 5586–5609, 2022. Citado na página 45.