

# Análise de Classificadores para Interfaces Cérebro-Computador

Palavras-Chave: Interfaces Cérebro-Computador, SSVEP Aprendizado de Máquina

Autores(as):

Vinícius Brito Santos Oliveira Carneiro, RA:244354- UNICAMP

Prof. Dr. Romis Ribeiro de Faissol Attux, DCA/FEEC - UNICAMP

## INTRODUÇÃO

Interfaces cérebro-computador (BCIs, do inglês *brain-computer interfaces*) são dispositivos capazes de traduzir comandos dados por um indivíduo a uma máquina usando vias distintas daquelas tradicionalmente associadas ao sistema nervoso [Wolpaw, 2002]. Essas interfaces possuem, tipicamente, uma operação próxima à do esquema apresentado na Fig. 1.

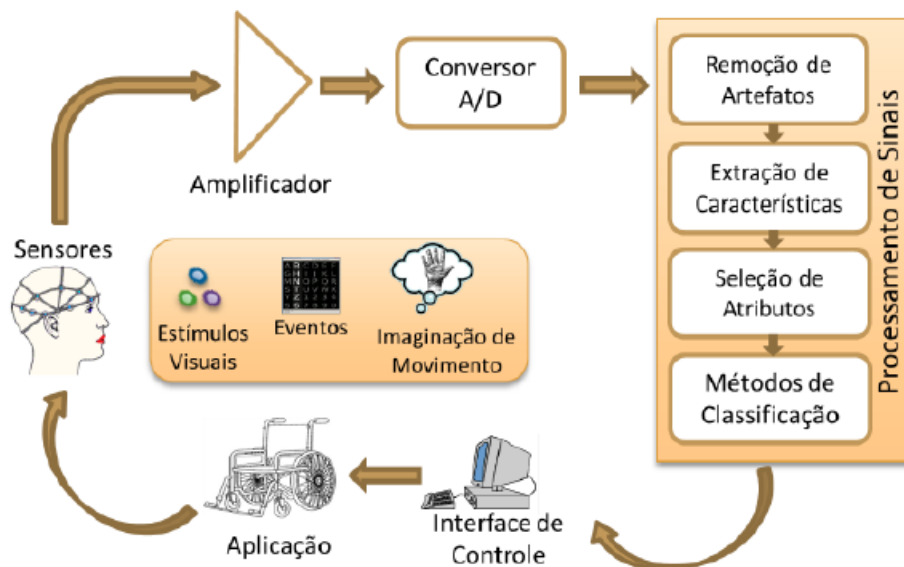


Figura 1 – Esquema de uma Interface Cérebro-Computador

Nesse esquema, o sinal de eletroencefalografia (EEG) é amostrado / digitalizado e, em seguida, passa por etapas de pré-processamento, extração / seleção de características e classificação. Com a resposta do classificador, pode-se acionar uma interface de controle ligada a uma aplicação genérica (no caso da ilustração, uma cadeira de rodas elétrica). Nota-se que o processo envolve algum tipo de protocolo e um paradigma, que, tipicamente, é baseado em potenciais visualmente evocados (SSVEPs, do inglês *steady state visually evoked potentials*), potenciais relacionados a eventos ou imagética motora [Wolpaw, 2002].

Pretende-se, aqui, focar a etapa de classificação através da análise do desempenho de dois classificadores não-lineares paradigmáticos – redes neurais e máquinas de vetores suporte – no problema de classificação de padrões em interfaces cérebro-computador baseadas em SSVEP. Este documento é um resumo do desenvolvimento do projeto até o momento.

## **METODOLOGIA**

No plano de trabalho, o cronograma proposto foi dividido em 5 etapas, sendo que, no primeiro semestre, os objetivos propostos foram os seguintes:

- Realização de um estudo geral sobre BCIs e sobre o paradigma de SSVEP em particular;
- Realização de um estudo sobre classificação não-linear com ênfase em redes neurais e SVMs.

Uma alteração feita no escopo inicial do projeto foi o fato que o estudo se baseou na linguagem de programação Python e não em MATLAB, uma vez que essa linguagem é amplamente utilizada tanto na indústria quanto na academia em aplicações ligadas ao aprendizado de máquina. Antes do início do projeto, o aluno cursou a disciplina MC102 (Algoritmos e Programação de Computadores) com base em Python, além de ter realizado estudos prévios por conta própria nesta mesma linguagem de programação [Géron 2022].

Do ponto de vista mais técnico, a etapa formativa em aprendizado de máquina foi realizada com o auxílio das Notas de Aula do Curso de Pós-Graduação IA048 (Aprendizado de Máquina) [Boccatto e Attux 2020]. Nessa etapa, foram estudados os seguintes temas: probabilidade, teoria da informação, teoria da estimação, regressão, classificação, redes neurais e máquinas de vetores-suporte. Sempre que necessário, foram realizadas simulações de apoio.

Neste momento, o projeto está em seu segundo semestre, no qual serão iniciadas as implementações práticas no âmbito de BCI juntamente com estudos complementares que sejam eventualmente necessários para finalizar o projeto. O cronograma para o semestre final do projeto é apresentado na Fig. 2. Atualmente, o projeto está no segundo semestre e terminando os seguintes objetivos:

- O terceiro objetivo é a construção, em Python, de uma interface completa que permita o uso de um classificador linear e dos dois tipos de classificadores não-lineares selecionados.
- O quarto objetivo é a análise comparativa dos desempenhos dos diferentes classificadores para um conjunto de dados representativo.

### **Classificação Linear:**

Neste estudo, utilizamos dados de um sinal EEG que continha 64 eletrodos. O eletrodo de número 28 foi escolhido para a análise por ser considerado o eletrodo de referência. Os dados do eletrodo 28 foram extraídos utilizando MATLAB e, em seguida, processados em Python com o objetivo de realizar uma regressão logística para distinguir entre frequências de 10 Hz e 12 Hz.

### **Extração de Frequências de 10 Hz e 12 Hz do Eletrodo 28:**

Para extrair as frequências de 10 Hz e 12 Hz do eletrodo 28, realizamos os seguintes passos:

### 1. Carregamento dos Dados:

- Os dados do eletrodo 28 foram carregados a partir de um arquivo CSV.

### 2. Filtragem dos Dados:

- Aplicamos filtros passa-banda nos dados do eletrodo 28 para isolar as frequências de 9-11 Hz (em torno de 10 Hz) e 11-13 Hz (em torno de 12 Hz). Isso foi feito utilizando um filtro Butterworth.

### 3. Segmentação da Sinal:

- Os dados foram segmentados em janelas de 2 segundos com um passo de 1 segundo. Cada segmento foi processado separadamente.

### 4. Transformada de Fourier (FFT):

- Para cada segmento, aplicamos a Transformada Rápida de Fourier (FFT) para converter os dados do domínio do tempo para o domínio da frequência.

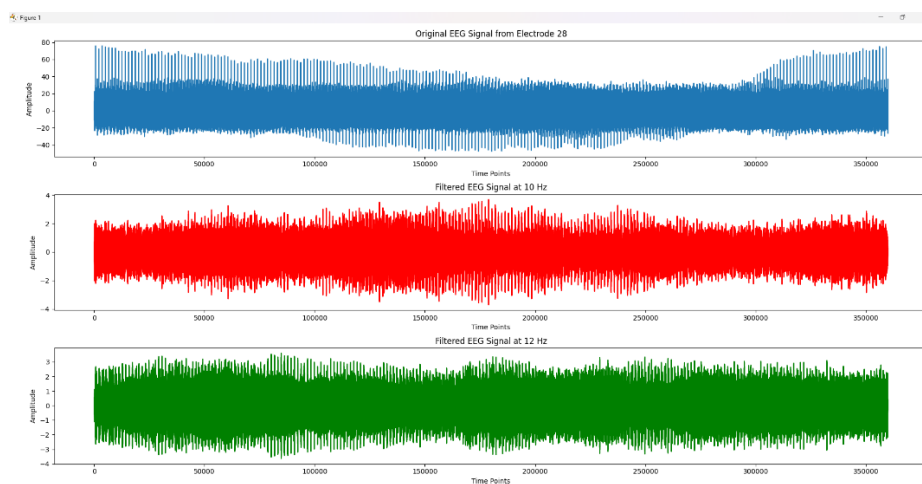


Figura 2 – Sinal original do eletrodo e os sinais em 10Hz e 12hz

## Características Extraídas e Seus Significados

A partir do espectro de potência obtido com a FFT, extraímos as seguintes características para cada janela de 2 segundos:

### 1. Potência Média (Mean Power):

- A Potência Média mede a energia média do sinal dentro da banda de frequência especificada (9-11 Hz para 10 Hz e 11-13 Hz para 12 Hz).

### 2. Potência de Pico (Peak Power):

- A Potência de Pico identifica o valor máximo da potência dentro da banda de frequência. Isso destaca o componente de frequência mais forte dentro da banda.

### 3. Amplitude Média (Mean Amplitude):

- A Amplitude Média mede a força média do sinal dentro da banda de frequência. Como a potência é o quadrado da amplitude, a raiz quadrada da potência dá a amplitude.

### 4. Desvio Padrão da Amplitude (Standard Deviation of Amplitude):

- O Desvio Padrão da Amplitude quantifica a variabilidade das amplitudes dentro da banda de frequência. Mede quão dispersos estão os valores de amplitude em relação à média.

### 5. Entropia Espectral (Spectral Entropy):

- A Entropia Espectral mede a complexidade ou aleatoriedade da distribuição de potência dentro da banda de frequência. Um valor mais alto indica uma distribuição mais uniforme de potência entre as frequências, sugerindo um sinal mais complexo.

## Regressão Logística e Análise dos Resultados

O modelo de regressão logística foi treinado e avaliado usando a matriz de características extraídas. Para isso, a matriz de características foi dividida em conjunto de treinamento (80%) e conjunto de teste (20%). O modelo foi então treinado utilizando o conjunto de treinamento e as previsões foram feitas no conjunto de teste.

A matriz de confusão gerada a partir do conjunto de teste revelou que:

- 71 janelas de 10 Hz foram corretamente classificadas como 10 Hz (Verdadeiros Positivos).
- 77 janelas de 12 Hz foram corretamente classificadas como 12 Hz (Verdadeiros Negativos).
- 51 janelas de 10 Hz foram incorretamente classificadas como 12 Hz (Falsos Negativos).
- 41 janelas de 12 Hz foram incorretamente classificadas como 10 Hz (Falsos Positivos).

A acurácia do modelo foi calculada como 61.7%, o que indica que o modelo de regressão logística conseguiu classificar corretamente cerca de 61.7% das janelas de 2 segundos dos dados EEG. A análise dos erros na matriz de confusão sugere que o modelo pode ter dificuldade em distinguir claramente entre as frequências de 10 Hz e 12 Hz, possivelmente devido à sobreposição das características extraídas ou à variabilidade dos sinais EEG.

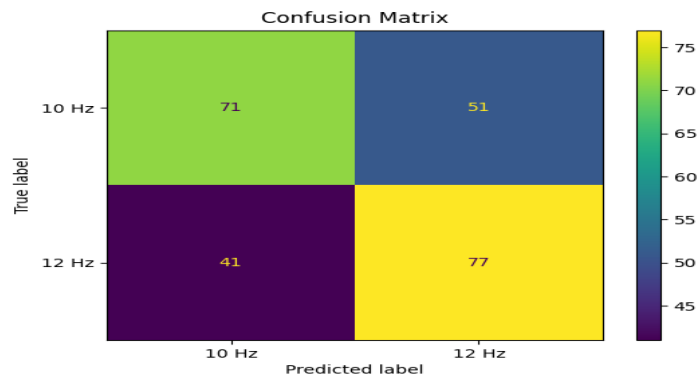


Figura 3 – Matriz de confusão

## Conclusão

Os resultados indicaram uma acurácia de 61,7% na classificação das frequências de 10 Hz e 12 Hz utilizando regressão logística, destacando a necessidade de refinamentos nos modelos de classificação e nas técnicas de extração de características. A análise sugere que a variabilidade dos sinais EEG e a sobreposição de características são fatores críticos que afetam o desempenho dos classificadores. Este estudo contribuiu significativamente para a compreensão dos desafios e técnicas no desenvolvimento de BCIs, estabelecendo uma base sólida para futuras melhorias na precisão e aplicabilidade desses sistemas em contextos reais.

## BIBLIOGRAFIA

- Boccatto e Attux, 2020] L. Boccatto, R. R. F. Attux, *Notas de Aula do Curso IA048 – Aprendizado de Máquina*, FEEC/UNICAMP, 2020.
- [Géron, 2022] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, O'Reilly, 2022.
- [Wolpaw, 2002] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, T. M. Vaughan, "Brain-Computer Interfaces for Communication and Control", *Clinical Neurophysiology*, Vol. 113, No. 6, pp. 767 – 791, 2002.