

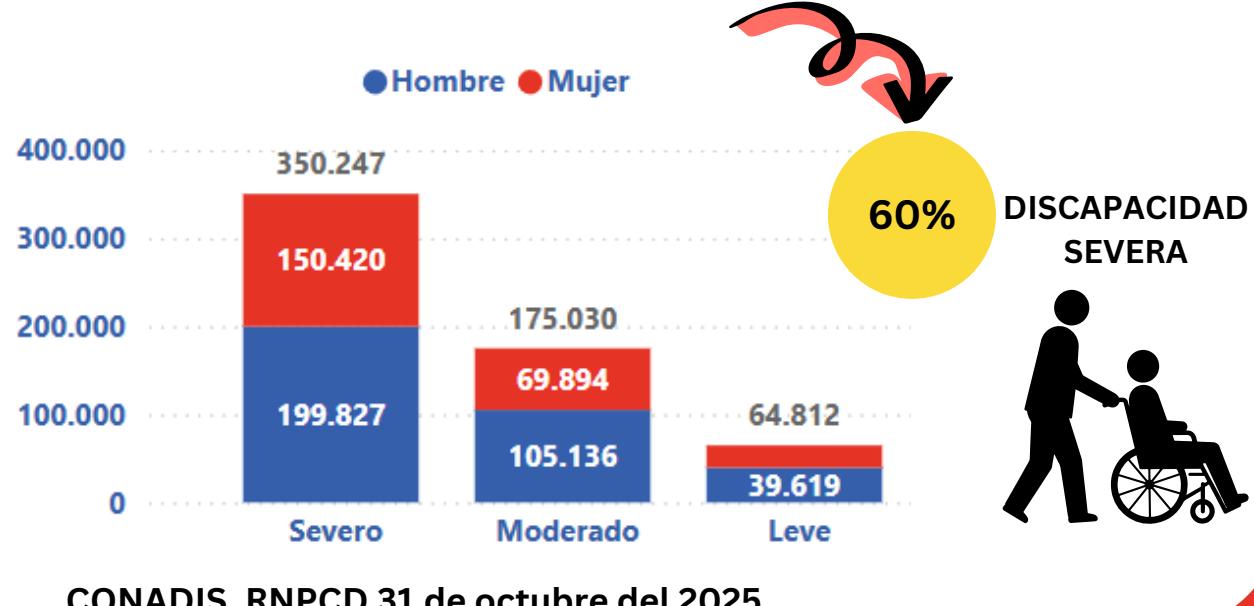
SISTEMA BCI CON ULTRACORTEX PARA OPERAR UN TECLADO INALÁMBRICO EN PERSONAS CON TETRALEJIA

Franco Peralta - Mariel Sánchez - Pamela Vilchez

1

INTRODUCCIÓN

Según el CONADIS hasta el 31 de octubre del 2025, a nivel nacional se ha registrado **591 434 personas** con discapacidad [1]



2

PROBLEMÁTICA

Tetraplejia

lesión en la médula espinal (C1-C8)

- Limitación física severa
- Pérdida del control motor voluntario
- Dificulta la interacción directa con el entorno
- Reduce significativamente la autonomía

3 PROPUESTA DE SOLUCIÓN

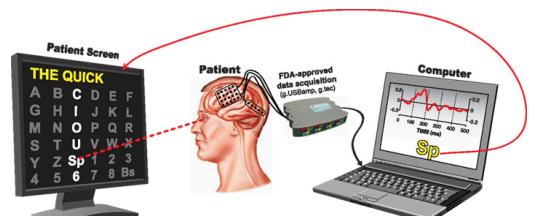
Sistema BCI no invasivo basado en el potencial P300, utilizando el Ultracortex Mark IV y OpenBCI para adquirir señales EEG que permiten a personas con tetraplejia operar un teclado inalámbrico sin movimiento corporal.

El sistema presenta una matriz de números que se iluminan de forma secuencial (paradigma oddball). Cuando el usuario fija su atención en un número, el estímulo objetivo genera un P300, el cual se detecta en regiones parietales y se traduce en un comando.

5

CONCLUSIONES

- El sistema BCI no invasivo basado en el potencial P300 mostró eficacia en la detección de respuestas cerebrales dentro del paradigma oddball, discriminando de manera confiable la intención del usuario.
- La combinación de Ultracortex Mark IV y OpenBCI permitió una adquisición EEG robusta, y el procesamiento en regiones parietales posibilitó la identificación clara del P300.
- La extrapolación mediante código tradujo la señal cerebral en la selección de letras, demostrando la viabilidad del reconocimiento de intención como paso hacia el control de un teclado inalámbrico en personas con tetraplejia.



6

TRABAJO A FUTURO

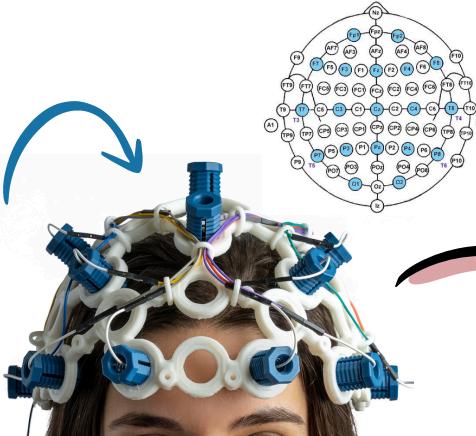
- Mejorar la calidad de la señal EEG mediante reducción de artefactos y filtros más robustos.
- Implementar modelos de deep learning para captar patrones no lineales del P300 y aumentar precisión e ITR.
- Integrar la decodificación del P300 con el control real de un teclado inalámbrico.
- Validar el desempeño en usuarios con limitaciones motoras dentro de entornos clínicos y de asistencia tecnológica.

4

METODOLOGÍA

Ultracortex
Mark IV

Registrar
señales EEG



Archivo EDF
Librería MNE en Python

1. Trigger detection

stim = raw.copy().pick(["StimulusCode"])[0][0][0]

Muestra Estímulo canal auxiliar

```
[1184 0 5]
[1216 0 6]
[1248 0 7]
[1280 0 4]
[1312 0 11]
[1344 0 1]
```

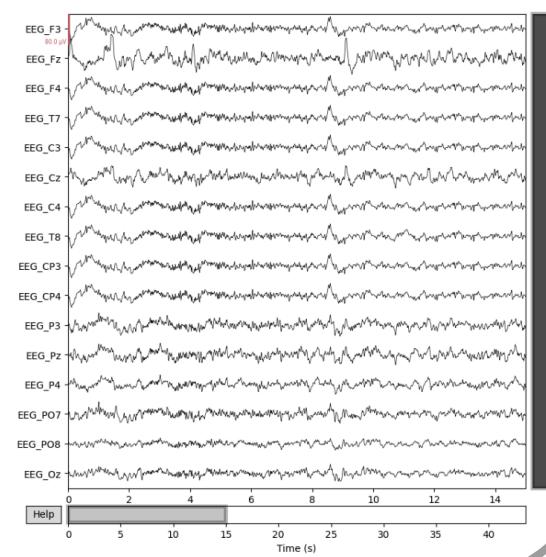
116 eventos

Matriz 6x6

- 6 filas (1-6)
- 6 columnas (7-12)

• Detecta transición 0 → código de estímulo

Datos crudos



2. Epoching

Segmentar datos EEG

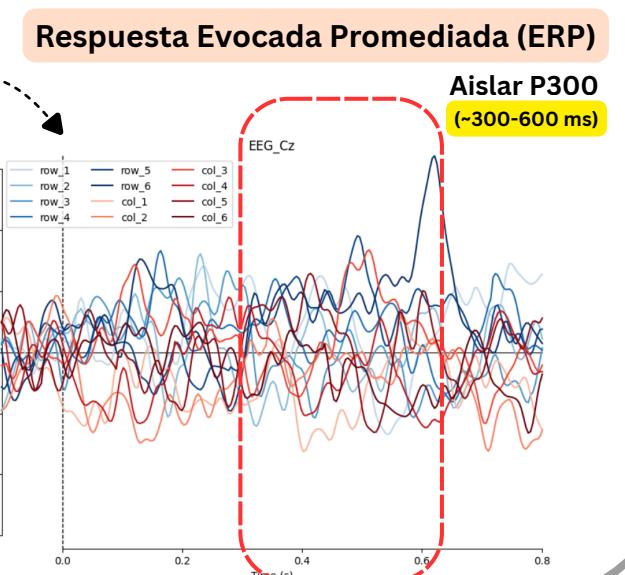
```
tmin = -0.1
tmax = 0.8

epochs = mne.Epochs(
    raw_eeg,
    events,
    event_id=event_id,
    tmin=tmin,
    tmax=tmax,
    baseline=(None, 0),
    preload=True
)

('row_1': 1.1587301587301569e-07,
 'row_2': -1.5168311688311689e-06,
 'row_3': 3.4936411736411776e-07,
 'row_4': -3.732442712842712e-06,
 'row_5': 5.561897546897547e-06,
 'row_6': 1.2603489177489174e-06,
 'col_1': -3.747166907166907e-07,
 'col_2': -1.0785761511216055e-07,
 'col_3': 3.1818874458874467e-07,
 'col_4': 1.5516459836459835e-06,
 'col_5': 5.70051428571427e-07,
 'col_6': -7.035760782427452e-07}
```

• Eliminar offset de la señal

116 muestras etiquetadas



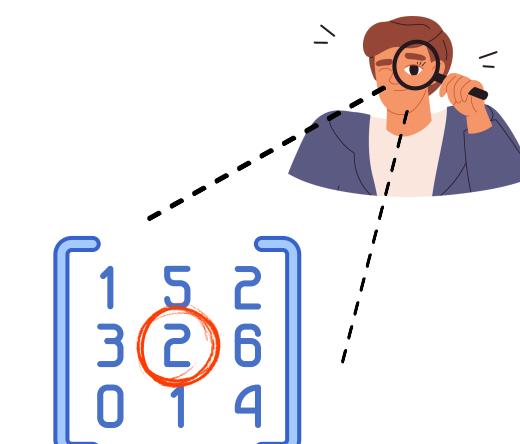
3. Cuantificación de la actividad cerebral

```
rows = {k:v for k,v in p300_amplitudes.items() if "row" in k} # Filas
cols = {k:v for k,v in p300_amplitudes.items() if "col" in k} # Columnas

row_selected = max(rows, key=rows.get)
col_selected = max(cols, key=cols.get)

row_selected, col_selected
('row_5', 'col_4')

matrix = [
    ["A", "B", "C", "D", "E", "F"],
    ["G", "H", "I", "J", "K", "L"],
    ["M", "N", "O", "P", "Q", "R"],
    ["S", "T", "U", "V", "W", "X"],
    ["Y", "Z", "1", "2", "3", "4"],
    ["5", "6", "7", "8", "9", "0"],
]
```



Repository
Github

