UNIVERSIDAD NACIONAL DEL LITORAL

DOCTORADO EN INGENIERÍA

Métodos de aprendizaje automático de mínima calibración para interfaces cerebro-computadora basadas en imaginería motora

Catalina María Galván

FICH
FACULTAD DE INGENIERÍA Y CIENCIAS HÍDRICAS

INTEC
INSTITUTO DE DESARROLLO TECNOLÓGICO PARA LA INDUSTRIA QUÍMICA

CIMEC
CENTRO DE INVESTIGACIÓN DE MÉTODOS COMPUTACIONALES

sinc(*i*)
INSTITUTO DE INVESTIGACIÓN EN SEÑALES, SISTEMAS E INTELIGENCIA COMPUTACIONAL

Tesis de Doctorado **2025**

UNIVERSIDAD NACIONAL DEL LITORAL
Facultad de Ingeniería y Ciencias Hídricas
Instituto de Desarrollo Tecnológico para la Industria Química
Centro de Investigación de Métodos Computacionales
Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional

Métodos de aprendizaje automático de mínima calibración para interfaces cerebro-computadora basadas en imaginería motora

Catalina María Galván

Tesis remitida al Comité Académico del Doctorado
como parte de los requisitos para la obtención
del grado de
DOCTOR EN INGENIERIA
Mención Inteligencia Computacional, Señales y Sistemas
de la
UNIVERSIDAD NACIONAL DEL LITORAL

2025

Secretaría de Posgrado, Facultad de Ingeniería y Ciencias Hídricas, Ciudad Universitaria, Paraje “El Pozo”, S3000, Santa Fe, Argentina

UNIVERSIDAD NACIONAL DEL LITORAL
Facultad de Ingeniería y Ciencias Hídricas
Instituto de Desarrollo Tecnológico para la Industria Química
Centro de Investigación de Métodos Computacionales
Instituto de Investigaciones en Señales, Sistemas e Inteligencia Computacional

Métodos de aprendizaje automático de mínima calibración para interfaces cerebro-computadora basadas en imaginería motora

Catalina María Galván

Lugar de Trabajo:

IMAL
Instituto de Matemática Aplicada del Litoral, IMAL, UNL,
CONICET, Santa Fe, Argentina
Facultad de Ingeniería y Ciencias Hídricas
Universidad Nacional del Litoral

Director:
Victoria Peterson IMAL-FIQ-UNL-CONICET

Co-director:
Diego Humberto Milone sinc(i)-FICH-UNL-CONICET

Jurado Evaluador:
Dr. Pablo Groisman IMAS, UBA-Conicet
Dr. Matías Di Martino Universidad Católica de Uruguay
Dr. Pablo Granitto CIFASIS, UNR-Conicet

2025

ACTA DE DEFENSA DE TESIS DE DOCTORADO

En la sede de la Facultad de Ingeniería y Ciencias Hídricas de la Universidad Nacional del Litoral, a los veintidós días del mes de diciembre del año dos mil veinticinco, se reúnen en forma online sincrónica los miembros del Jurado designado para la evaluación de la Tesis de Doctorado en Ingeniería, Mención Inteligencia Computacional, Señales y Sistemas, titulada ***"Métodos de aprendizaje automático de mínima calibración para interfaces cerebro-computadora basadas en imaginería motora."***, desarrollada por la Ing. Catalina María GALVÁN, DNI 39.032.008, bajo la dirección de la Dra. Victoria Peterson y la codirección del Dr. Diego Milone. Ellos son: el Dr. Pablo Groisman, el Dr. Matías Di Martino, y el Dr. Pablo Granitto.----------

La Presentación oral y defensa de la Tesis se efectúan bajo la modalidad online sincrónica según lo establecido por Resolución CS N° 382/21.----------

Luego de escuchar la Defensa Pública y de evaluar la Tesis, el Jurado considera:

Que la tesis representa una contribución sobresaliente y de alto nivel al campo de las interfaces cerebro-computadora para rehabilitación motora. Combina innovación metodológica, rigor experimental y una clara orientación hacia la aplicabilidad clínica. Los métodos desarrollados tienen el potencial de acelerar la adopción de las MI-BCIs en entornos reales, beneficiando a pacientes con discapacidad motora. La defensa de la tesis fue clara y precisa y la candidata demostró mucha solidez y conocimiento del área en la exposición y ante las preguntas del Jurado.
Por lo tanto, el Jurado aprueba la Tesis con calificación 10 (Diez) Sobresaliente.

Sin más, se da por finalizado el Acto Académico con la firma de los miembros del Jurado al pie de la presente. ----------

----------	----------	----------
Dr. Pablo Groisman	Dr. Matías Di Martino	Dr. Pablo Granitto

FACULTAD DE INGENIERÍA Y CIENCIAS HÍDRICAS * SANTA FE *
DR. JOSE LUIS MACOR
SECRETARIO DE POSGRADO
Facultad de Ingeniería y Cs. Hídricas

Dependencia
Dirección postal
Código postal, Ciudad, Argentina
Teléfono
Correo electrónico

DECLARACIÓN DEL AUTOR

Esta tesis ha sido remitida como parte de los requisitos para la obtención del grado académico de Doctor en Ingeniería - Mención Inteligencia Computacional, Señales y Sistemas ante la Universidad Nacional del Litoral y ha sido depositada en la Biblioteca de la Facultad de Ingeniería y Ciencias Hídricas para que esté a disposición de sus lectores bajo las condiciones estipuladas por el Reglamento de la mencionada Biblioteca.

Citaciones breves de esta tesis son permitidas sin la necesidad de un permiso especial, en la suposición de que la fuente sea correctamente citada. Solicitudes de permiso para la citación extendida o para la reproducción parcial o total de este manuscrito serán concebidos por el portador legal del derecho de propiedad intelectual de la obra.

TESIS POR COMPILACIÓN

La presente tesis se encuentra organizada bajo el formato de Tesis por Compilación, aprobado en la resolución Nº 255/17 (Expte. Nº 888317-17) por el Comité Académico de la Carrera Doctorado en Ingeniería, Facultad de Ingeniería y Ciencias Hídricas, Universidad Nacional del Litoral (UNL). De dicha resolución:

"En el caso de optar por la Tesis por Compilación, ésta consistirá en una descripción técnica de al menos 30 páginas, redactada en español e incluyendo todas las investigaciones abordadas en la tesis. Se deberán incluir las secciones habituales indicadas a continuación en la Sección Contenidos de la Tesis. Los artículos científicos publicados por el autor, en el idioma original de las publicaciones, deberán incluirse en un Anexo con el formato unificado al estilo general de la Tesis indicado en la Sección Formato. El Anexo deberá estar encabezado por una sección donde el tesista detalle para cada una de las publicaciones cuál ha sido su contribución. Esta sección deberá estar avalada por su director de Tesis. El documento central de la Tesis debe incluir referencias explícitas a todas las publicaciones anexadas y presentar una conclusión que muestre la coherencia de dichos trabajos con el hilo conceptual y metodológico de la tesis. Los artículos presentados en los anexos podrán ser artículos publicados, aceptados para publicación (en prensa) o en revisión."

Agradecimientos

Dicen que ninguna meta alcanzada es tan significativa como aquello en lo que nos convertimos buscando llegar a ella. A esta frase, que resume en pocas palabras cómo me siento, agregaría que esta transformación es el reflejo de conversaciones, desafíos compartidos y manos tendidas en el momento justo. Por eso, esta es también una oportunidad para reconocer a personas que fueron parte esencial de este recorrido y decirles "Gracias".

En primer lugar, quiero agradecer profundamente a mis directores, Victoria, Diego y Ruben, por su guía constante y su inmensa generosidad. Por saber acompañar en libertad y con paciencia, estando presentes en los grandes momentos y en los detalles. Gracias, además, por introducirme a la vocación científica con excelencia y compromiso, mostrando con el ejemplo que hacer ciencia es, ante todo, una forma de pensar y de estar en el mundo.

Gracias a mi familia y seres queridos. A Juan, mi compañero, por ser refugio y hogar, por regalarme paz. A mis padres y hermanas, por enseñarme a soñar en grande y sabiendo estar para mí incondicionalmente. Gracias a mi familia extendida y amigos, que celebran conmigo este logro, me alientan y confían en mí, incluso cuando a mí me cuesta hacerlo.

Finalmente, quiero agradecer a las instituciones que hicieron posible esta tesis.

- Al IMAL, que fue mi lugar de trabajo y también un espacio de crecimiento y de compartir diario. En particular, al NiCALab, al que tuve la suerte de ver crecer y que me regaló mucho más que compañeros de trabajo. Al sinc, instituto donde encontré también colegas y vínculos que enriquecieron profundamente este recorrido.

- Al CONICET, por haberme dado la beca que financió este doctorado e hizo posible que me dedicara plenamente a este camino.

- A la UNL, casa académica de este doctorado, por la formación brindada y por darme el espacio para desempeñarme en la docencia.

Acrónimos

En el presente documento, para muchas de las abreviaturas se utilizan los acrónimos correspondientes a las expresiones en idioma inglés, por ser ésta la forma más utilizada en la bibliografía. A continuación se listan los acrónimos utilizados a lo largo del mismo.

ACV	Accidente cerebro-vascular
ADT	Entrenamiento con datos de adaptación
BCI	Interfaz cerebro-computadora
BF	Filtrado rechaza banda
BOTDA	Transporte óptimo hacia atras para adaptación de dominio
CD	Dropout de canales
CFT	Finetuning del clasificador
CSP	Patrones espaciales comunes
CTI	Inclusión del ensayo actual
DL	Aprendizaje profundo
EA	Alineación euclídea
EEG	Electroencefalografía
ERD	Desincronización relacionada a eventos
ERS	Sincronización relacionada a eventos
FFT	Finetuning de modelo completo
GNA	Agregado de ruido gaussiano
GR-CSP	Patrones espaciales comunes con recentrado genérico
LDA	Análisis discriminante lineal
MI	Imaginería motora
MRI	Imagen por resonancia magnética
PSD	Densidad espectral de potencia
SDT	Entrenamiento con datos fuente

SR	Segmentación y recombinación
TM	Enmascarado temporal
WGAN	Redes generativas adversarias Wasserstein
XS-BOT	Transporte óptimo hacia atrás entre sujetos

Índice general

Índice de figuras

Índice de tablas

Resumen

El deterioro de la función motora de miembro superior tras accidentes cerebrovasculares u otras condiciones neurológicas afecta significativamente la calidad de vida. En este contexto, las interfaces cerebro-computadora basadas en imaginería motora (MI-BCIs) son prometedoras para la asistencia y la rehabilitación, al transformar la actividad cerebral en comandos de control sin usar de las vías neuromusculares. Sin embargo, su aplicación se ve limitada por la compleja decodificación de las señales de electroencefalografía (EEG). Una gran limitación es la variabilidad inter e intra-sujeto del EEG, que demanda extensas sesiones de calibración. En esta tesis se desarrollaron métodos de aprendizaje automático para reducir dichos tiempos de calibración y mejorar la robustez en la decodificación.

En primer lugar se desarrolló de PySimMIBCI, un método para generar datos artificiales de EEG que incorporan información sujeto-específica y características fisiológicamente plausibles para tareas de MI de mano. Este método permite introducir en la simulación factores de variabilidad como la fatiga o diferentes capacidades de modulación, que son aspectos interesantes a tener en cuenta a la hora de desarrollar y validar modelos de decodificación. Mediante PySimMIBCI se generaron datos sintéticos con características específicas de sujetos reales y se las utilizó para la aumentación de datos, obteniendo mejoras significativas en el desempeño de la decodificación.

El segundo eje abordó la transferencia de aprendizaje entre sujetos y bases de datos para reducir los tiempos de calibración para cada usuario. Con tal fin, se propuso XS-BOT, un enfoque de adaptación de dominio basado en transporte óptimo hacia atrás. A partir de un modelo entrenado en datos provenientes de múltiples usuarios, XS-BOT alinea las características del nuevo sujeto con la distribución de entrenamiento. La validación en diferentes bases de datos y modelos de decodificación demostró que el método supera marcadamente a otros enfoques del estado del arte, alcanzando tasas de aciertos robustas con un mínimo número de datos de calibración del nuevo usuario y de electrodos.

En conjunto, los aportes metodológicos y experimentales de esta tesis permiten avanzar hacia MI-BCIs de mínima calibración, robustas y centradas en la persona usuaria, características alineadas con las necesidades en un contexto de rehabilitación motora.

Abstract

The impairment of upper limb motor function following stroke or other neurological conditions significantly impacts quality of life of the affected person. In this context, motor imagery-based brain–computer interfaces (MI-BCIs) are promising tools for assistance and rehabilitation, as they transform brain activity into control commands without relying on neuromuscular pathways. However, their application is limited by the complex decoding of electroencephalography (EEG) signals. A major challenge is the inter- and intra-subject variability of EEG, which leads to the need for lengthy calibration sessions. This thesis addressed the development of machine learning methods to reduce calibration times and enhance the robustness of decoding.

First, PySimMIBCI was developed, a method for generating artificial EEG data that incorporates subject-specific information and physiologically plausible features for hand MI tasks. This method allows simulating variability factors such as fatigue or different modulation capacities, which are important aspects to consider when developing and validating decoding models. Using PySimMIBCI, synthetic data with subject-specific characteristics were generated and employed for data augmentation, leading to significant improvements in decoding performance.

The second focus addressed transfer learning across subjects and datasets to reduce calibration times for each user. To this end, XS-BOT was proposed, a domain adaptation approach based on backward optimal transport. Starting from a model trained on data from multiple users, XS-BOT aligns the features of the new subject with the training distribution. Validation across different datasets and decoding models demonstrated that the method markedly outperforms other state-of-the-art approaches, achieving robust accuracy rates with a minimal number of calibration data from the new user and electrodes.

Taken together, the methodological and experimental contributions of this thesis advance toward minimally calibrated, robust, and user-centered MI-BCIs, features that align with the needs in motor rehabilitation contexts.

Capítulo 1

Introducción

1.1. Motivación

La pérdida parcial o total de la función motora del miembro superior constituye una de las limitaciones funcionales más frecuentes asociadas a diversas condiciones neurológicas, como los accidentes cerebrovasculares (ACV), las lesiones medulares y las enfermedades neurodegenerativas. Estas alteraciones limitan la posibilidad de realizar actividades diarias de manera autónoma, disminuyendo significativamente la calidad de vida de la persona afectada [1]. En este contexto, las interfaces cerebro-computadora (BCIs, por sus siglas en inglés) surgen como una tecnología con potencial para asistencia y rehabilitación, ya que establecen una vía de control alternativa que utiliza únicamente señales provenientes de la actividad cerebral. De esta forma, no requieren que las vías neuromusculares de la persona se encuentren conservadas para establecer la interacción con el entorno [2].

Una forma no invasiva y relativamente económica de medir la actividad cerebral es mediante electroencefalografía (EEG) de superficie. Existen diferentes paradigmas para transformar la información contenida en el EEG en comandos de control o comunicación. En particular, las BCIs basadas en imaginería motora (MI-BCIs, por sus siglas en inglés) se fundamentan en la representación mental de determinados movimientos. En contextos de asistencia o rehabilitación, una MI-BCI puede traducir la intención de movimiento en señales de control, permitiendo ejecutar dicho movimiento mediante un exoesqueleto o brazo robótico [3]. El uso de estos sistemas ha demostrado ser efectivo para la recuperación motora de pacientes crónicos de ACV con paresia grave [4].

Desde el punto de vista computacional, las BCIs están basadas en modelos de aprendizaje automático que decodifican la actividad cerebral, lo que plantea un desafiante problema de reconocimiento de patrones. En los sistemas de MI basados en EEG, el algoritmo

debe identificaren en tiempo real la clase (por ejemplo, la imaginación del movimiento de una mano) a la que un segmento o ensayo de EEG pertenece, y convertirlo en una salida de control. Por lo tanto, es esencial que el modelo de aprendizaje automático sea capaz de detectar en la señal de EEG características específicas de cada tarea en cuestión, las cuales deben ser suficientemente distintas a aquellas relacionadas con cualquier otra tarea [5]. El reconocimiento de patrones de MI a partir de la señal de EEG enfrenta varios desafíos, muchos de los cuales se relacionan con características intrínsecas de esta señal [6, 7].

Una de las principales limitaciones de los sistemas basados en EEG es su baja relación señal a ruido y la dificultad de conocer con certeza las fuentes neuronales que dieron origen a la señal registrada. Incluso en contextos controlados, donde se instruye a la persona a realizar una tarea de MI específica, no es posible verificar si efectivamente llevó a cabo la tarea mental indicada. Esta incertidumbre complica el desarrollo, ajuste y evaluación de modelos de decodificación. En este contexto, la generación de señales de MI artificiales desempeña un papel crucial al proporcionar datos bien definidos para la validación e interpretación de modelos basados en EEG [8]. Con este fin, se han dedicado muchos trabajos a la generación de señales mediante el desarrollo de modelos neurofisiológicamente plausibles. Estos modelos buscan conservar los mecanismos y dinámicas subyacentes de la actividad cerebral, permitiendo generar datos sintéticos que replican de forma realista los patrones observados en registros reales. En esta línea, en [9] se generan datos de EEG simulados combinando actividad biológicamente plausible con modelos de cabeza específicos de cada usuario basados en imágenes por resonancia magnética (MRI). De forma similar, SimBCI [8] se destacó por ofrecer una herramienta para simular datos EEG siguiendo protocolos experimentales específicos de BCI. Estos avances en la generación de datos artificiales han contribuido a la exploración y validación de nuevos algoritmos de decodificación en el campo de las EEG-BCIs. Sin embargo, a pesar de su relevancia, aún no se ha incluido en dichos modelos información relacionada al nivel de fatiga y la capacidad de modulación cerebral del usuario durante la ejecución de tareas de BCI.

Por otro lado, la gran variabilidad asociada al EEG se traduce en cambios significativos en las propiedades estadísticas de la señal, no sólo entre diferentes usuarios sino también a lo largo del tiempo para una misma persona [7], lo que dificulta que modelos previamente entrenados alcancen un buen desempeño [10]. En consecuencia, en el área de MI-BCIs para rehabilitación tradicionalmente se ha requerido que el usuario realice largas y tediosas sesiones de calibración para recopilar suficientes datos de entrenamiento, lo que limita la practicidad y usabilidad de estos sistemas. La cantidad de datos de entrena-

miento necesarios se ha vuelto aún más crítica con la aparición de modelos de aprendizaje profundo (DL, por sus siglas en inglés), cuyo rendimiento depende fuertemente del volumen de datos disponibles para el ajuste [11]. Para que las MI-BCIs sean viables para aplicaciones reales, el tiempo destinado a la adquisición de datos de calibración y al ajuste del modelo debería ser mínimo en comparación con el tiempo de uso del sistema. Además, en el caso particular de la rehabilitación motora [4, 12, 13], una calibración breve permite que el paciente reciba retroalimentación que refleje su intención casi desde el inicio de la sesión, lo que incrementa la efectividad de la terapia [14]. Para mitigar esta problemática, se han propuesto diferentes estrategias. Por un lado, se han desarrollado enfoques para el escenario intra sujeto, en los que el modelo se entrena y evalúa utilizando datos del mismo usuario [15, 16]. Por otro lado, se han explorado estrategias de transferencia de aprendizaje entre sujetos, que buscan aprovechar datos disponibles de otros usuarios para entrenar el modelo y así reducir el esfuerzo de calibración en nuevos individuos [17].

Uno de los enfoques más utilizados en el contexto intra sujeto son las estrategias de aumentación de datos, las cuales buscan incrementar la cantidad y diversidad de las observaciones disponibles para el entrenamiento sin necesidad de adquirir nuevos datos de EEG. Estas incluyen la generación de nuevos ensayos de EEG mediante la recombinación aleatoria de segmentos de ensayos existentes [16, 18], la combinación a nivel de canales [19], la adición controlada de ruido gaussiano [20, 21], y otras transformaciones que preservan la información significativa de la señal [22]. Más recientemente, se han destacado métodos basados en modelos generativos, como las redes adversarias generativas (GANs), que permiten crear muestras artificiales con características temporales, frecuenciales y espaciales similares a las señales reales [23-27]. Sin embargo, el uso de señales EEG simuladas basadas en modelos neurofisiológicos no ha sido explorado como estrategia para la aumentación de datos reales, a pesar de su potencial para enriquecer los conjuntos de entrenamiento con ejemplos controlables y fisiológicamente plausibles.

En el contexto de la decodificación de MI entre sujetos, donde se busca que el sistema pueda utilizarse en datos de nuevos usuarios sin necesidad de entrenarse desde cero para cada uno, se han explorado diversas estrategias para afrontar la alta variabilidad entre sujetos. El método de Patrones Espaciales Comunes (CSP, por sus siglas en inglés), el enfoque tradicional de extracción de características en el contexto de MI-BCIs basadas en EEG, ha sido ampliamente estudiado en distintas variantes que buscan mejorar su capacidad de generalización [17, 28-35]. En esta línea, el método recientemente introducido CSP con recentrado genérico (GR-CSP, por sus siglas en inglés) [17] aplica transformaciones afines a las matrices de covarianza específicas de cada ensayo utilizadas por el algoritmo

CSP. Por otro lado, en el caso de las arquitecturas de DL, las estrategias de ajuste fino (*fine-tuning*) han sido el enfoque predominante para adaptar modelos preentrenados a nuevos sujetos [36-38]. De manera complementaria a la adaptación del modelo, las técnicas de alineamiento de datos ofrecen un enfoque general aplicable a diferentes esquemas de decodificación. Entre ellas, la alineación euclídea (EA, por sus siglas en inglés) [39] ha destacado por su simplicidad y efectividad [40]. Si bien estas técnicas representan avances relevantes en el abordaje de la variabilidad entre sujetos y en la mejora de la aplicabilidad de los sistemas MI-BCI, los resultados obtenidos hasta el momento aún distan de los alcanzados en escenarios intra sujeto. Además, la gran mayoría de los estudios han validado sus enfoques utilizando señales EEG registradas con más de 20 electrodos [41-43], lo cual limita su aplicabilidad en contextos donde se dispone de configuraciones más reducidas. En este sentido, es también deseable que los modelos desarrollados para la decodificación de MI obtengan un rendimiento competitivo con configuraciones reducidas, facilitando así la construcción de BCIs más accesibles y amigables para el usuario.

En este proyecto se desarrollaron e implementaron métodos de aprendizaje automático que buscan acercar las MI-BCIs a su uso para rehabilitación y asistencia del movimiento de miembro superior. En particular, se abordaron los siguientes desafíos: i) la falta de datos de EEG simulados bajo el paradigma de MI neurofisiológicamente plausibles, ii) el pequeño volumen de datos sujeto-específicos para el entrenamiento de modelos de DL para la decodificación de MI, iii) la marcada variabilidad entre sujetos y bases de datos de EEG correspondientes a tareas de MI. Estos desafíos fueron abordados mediante dos bloques de trabajo. El primer bloque se centró en la simulación de señales de EEG correspondientes a tareas de MI que sean adecuadas tanto para el desarrollo de modelos de aprendizaje automático como para estrategias de aumentación de datos. En segundo lugar, se propuso un enfoque de transferencia de aprendizaje entre sujetos que permitiese reducir los tiempos de calibración de las MI-BCIs utilizando un número reducido de canales de EEG.

1.2. Objetivos

El objetivo general de esta tesis es desarrollar métodos de aprendizaje automático confiables y que permitan reducir los tiempos de calibración de sistemas MI-BCIs basados en EEG, para impulsar su uso como tecnologías de asistencia y rehabilitación motora de miembro superior.

Objetivos específicos:

1. Desarrollar un método de simulación de datos de EEG correspondientes a tareas de MI, fisiológicamente plausibles y con inclusión de información sujeto-específica.
2. Proponer estrategias de aumentación de datos para modelos de aprendizaje profundo empleando datos de EEG correspondientes a tareas de MI simulados.
3. Implementar estrategias de transferencia de aprendizaje para minizar los tiempos de calibración en MI-BCIs.
4. Estudiar la capacidad de transferencia de aprendizaje entre sujetos de diferentes bases de datos.
5. Validar las técnicas propuestas empleando bases de datos públicas de libre acceso.

1.3. Organización del documento

Esta tesis se estructura en 5 capítulos y un Anexo donde se recopilan las publicaciones científicas relacionadas con la misma.

- Capítulo 1: En este primer capítulo se describió la problemática que motivó el desarrollo de esta tesis y se presentó el estado del arte. Además, se estableció el objetivo general y los objetivos específicos planteados.
- Capítulo 2: Aquí se presentan los materiales y la metodología utilizada a lo largo de la tesis para la decodificación de MI. Esto incluye las bases de datos, los modelos de decodificación, las métricas y las pruebas estadísticas utilizadas.
- Capítulo 3: En este capítulo se presenta un método de simulación de datos neurofisiológicamente plausibles y su uso para aumentación de datos durante el entrenamiento de modelos de aprendizaje profundo para decodificación de MI. Los resultados aquí presentados se relacionan con los objetivos específicos 1, 2 y 5.
- Capítulo 4: Presenta un enfoque para la adaptación de dominio entre sujetos, que permite la reducción de los tiempos de calibración al mínimo. Estos resultados se relacionan con los objetivos específicos 3, 4 y 5.
- Capítulo 5: Aquí se presentan las conclusiones de la tesis.

Capítulo 2

Materiales y metodología para la decodificación de MI-EEG

En este capítulo se presentan los materiales y la metodología utilizados a lo largo de la tesis para abordar la decodificación de MI a partir de señales de EEG. En particular, se describen las bases de datos empleadas, los modelos de aprendizaje automático utilizados y los criterios adoptados para su evaluación. Las siguientes secciones desarrollan en detalle cada uno de estos aspectos.

2.1. Bases de datos

Se eligieron bases de datos de EEG de acceso público correspondientes a tareas de MI de mano. En esta sección, se describirán las bases de datos utilizadas a lo largo de la tesis, tanto para el diseño como para la evaluación de los distintos métodos propuestos.

2.1.1. Base de datos 1: BCI Competition IV-1

La base de datos 1 corresponde a una parte del *Dataset 1* de la *BCI Competition IV* [44]. El conjunto de datos original contiene registros de sujetos reales y también datos artificiales generados mediante simulaciones. En este trabajo se utilizaron únicamente los datos reales correspondientes a los sujetos b y g, quienes realizaron tareas de MI de mano izquierda y derecha. Para cada sujeto, se registró una única sesión, dividida en una fase de entrenamiento y otra de evaluación. La fase de entrenamiento comprende las dos primeras rondas, con un total de 100 ensayos por clase, mientras que en la fase de evaluación la cantidad de ensayos disponibles varía entre sujetos.

Los datos fueron registrados a 1000 Hz utilizando 59 electrodos distribuidos densamente sobre las áreas sensoriomotoras, mediante un amplificador BrainAmp MR Plus (Brain Products GmbH, Munich, Germany). Los datos están disponibles públicamente en: http://www.bbci.de/competition/iv/.

2.1.2. Base de datos 2: Lee2019_MI

La base de datos Lee2019_MI [45] comprende registros de MI de mano izquierda y mano derecha de 54 participantes. Cada participante completó dos sesiones en días diferentes. A lo largo de esta tesis se utilizaron los datos correspondientes a las rondas de entrenamiento de ambas sesiones, para los cuales se tiene la etiqueta correspondiente a la tarea de MI indicada para cada ensayo. Cada sesión contiene 100 ensayos en total, balanceados por clase (50 ensayos por clase).

La señal de EEG fue registrada a 1000 Hz utilizando 62 electrodos posicionados según el sistema internacional 10-20. Para la adquisición se utilizó un amplificador BrainAmp (Brain Products; Munich, Germany). Los datos están disponibles públicamente en: https://gigadb.org/dataset/100542.

2.1.3. Base de datos 3: Cho2017

La base de datos Cho2017 [46] incluye datos de EEG correspondientes a tareas de MI de mano derecha y mano izquierda. Comprende datos de una única sesión de 52 participantes. Cada participante realizó entre 200 y 240 ensayos (5 o 6 rondas) balanceados por clases.

Las señales de EEG fueron adquiridas a 512 Hz utilizando el sistema Biosemi ActiveTwo. Se registraron 64 canales del sistema internacional 10-10. Los datos están disponibles públicamente en: http://gigadb.org/dataset/100295.

2.1.4. Bases de datos 4: Dreyer2023

La base de datos Dreyer2023 [47] está compuesta por tres subconjuntos (A, B y C), que siguen un mismo protocolo de MI de mano izquierda y mano derecha, aunque cada uno fue diseñado para abordar distintas preguntas de investigación. Cada participante completó una sesión de adquisición con un total de 240 ensayos balanceados por clase.

Las señales de EEG fueron registradas a 512 Hz utilizando 27 electrodos distribuidos según el sistema internacional 10-20 mediante un sistema g.USBAmp (g.tec, Austria). Los

subconjuntos A, B y C incluyen 60, 21 y 6 participantes, respectivamente. Los datos están disponibles públicamente en: https://doi.org/10.5281/zenodo.8089820.

2.2. Modelos de decodificación

Las técnicas para la decodificación de MI a partir del EEG han evolucionado impulsadas por los cambios en el área aprendizaje automático. Los modelos tradicionales están fuertemente basados en modelos clásicos de procesamiento de señales y aprendizaje estadístico, con enfoques que combinan filtrado espacial y clasificadores lineales. Más recientemente, con los avances en modelos de aprendizaje profundo, este tipo de arquitecturas han surgido como alternativas con potencial en el contexto de las EEG-BCIs que permiten aprender representaciones directamente a partir de los datos crudos.

A lo largo de esta tesis se emplearon distintos modelos de decodificación de MI, abarcando tanto estrategias tradicionales como modelos basados en redes neuronales. Específicamente, se incluyó un enfoque clásico ampliamente adoptado en la literatura y distintas arquitecturas convolucionales de aprendizaje profundo que representan el estado del arte en el área. En las siguientes secciones se describen brevemente los modelos utilizados.

2.2.1. Enfoque tradicional: CSP + LDA

Uno de los enfoques más establecidos para la decodificación de MI a partir de EEG se basa en la combinación del método de Patrones Espaciales Comunes (CSP) con un clasificador lineal, típicamente Análisis Discriminante Lineal (LDA, por sus siglas en inglés).

CSP es un método supervisado de filtrado espacial diseñado para transformar señales de EEG multicanal registradas en dos condiciones distintas, de manera que la nueva representación de los datos permita distinguir entre clases a partir de su varianza. En particular, busca proyectar las señales a un nuevo espacio donde las componentes obtenidas presenten alta varianza para una clase y baja varianza para la otra. Como la varianza de una señal de EEG filtrada en cierta banda equivale a su potencia en esa banda, este principio es especialmente útil en tareas de MI, donde las diferencias entre condiciones se manifiestan como variaciones de potencia en ciertas bandas de frecuencia, producto de fenómenos de desincronización y sincronización relacionados con eventos (ERD/ERS, por sus siglas en inglés) motores [48].

Formalmente, CSP descompone la señal mediante una matriz $\mathbf{W} \in \mathbb{R}^{C \times C}$, siendo C

el número de canales de EEG, que proyecta la señal $\mathbf{x}[n] \in \mathbb{R}^C$ a $\mathbf{x}_{CSP}[n] \in \mathbb{R}^C$, según

$$\mathbf{x}_{CSP}[n] = \mathbf{W}^T \mathbf{x}[n].$$

Cada vector columna $\mathbf{w}_i \in \mathbb{R}^C$ (con $i = 1, \ldots, C$) de $\mathbf{W}$ representa un filtro espacial, mientras que las columnas $\mathbf{a}_l \in \mathbb{R}^C (l = 1, \ldots, C)$ de la matriz $\mathbf{A} = (\mathbf{W}^{-1})^T$ son los patrones espaciales asociados. Dichos patrones pueden ser visualizados mediante mapas topográficos que permiten interpretar neurofisiológicamente las soluciones obtenidas [43].

Para aprender $\mathbf{W}$ se parte de un conjunto de E ensayos de EEG $\{\mathbf{X}_j\}_{j=1}^{E}$. Cada ensayo $\mathbf{X}_j \in \mathbb{R}^{C \times N}$, siendo N el número de muestras, pertenece a una y sólo una de dos clases posibles $k \in \{1, 2\}$. La matriz de covarianza espacial promedio correspondiente a la k-ésima clase, $\mathbf{\Sigma}_k$, se calcula como [49]

$$\mathbf{\Sigma}_k = \frac{1}{E_k} \sum_{j=1}^{E_k} \frac{1}{t-1} \mathbf{X}_j \mathbf{X}_j^T, \tag{2.1}$$

donde E_k es el número de ensayos que pertenecen a la clase k y $\mathbf{X}_j$ se supone centrada.

Optimizar los filtros espaciales de CSP consiste en maximizar el funcional

$$J_{\text{CSP}}(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{\Sigma}_1 \mathbf{w}}{\mathbf{w}^T \mathbf{\Sigma}_2 \mathbf{w}}, \tag{2.2}$$

el cual se resuelve mediante una descomposición generalizada en autovalores [32]. Una vez que los filtros fueron aprendidos, las características de CSP más utilizadas para la clasificación son el logarítmo de la varianza de las señales proyectadas:

$$f_i^j = \log\left(\mathbf{w}_i^T \left(\tfrac{1}{N-1} \mathbf{X}_j \mathbf{X}_j^T\right) \mathbf{w}_i\right). \tag{2.3}$$

Generalmente se seleccionan un número reducido M de características, correspondientes a filtros espaciales $\mathbf{w}_i$ seleccionados. En general, se emplean entre 2 y 3 pares de filtros $\mathbf{w}_i$, correspondientes a los primeros y últimos de la descomposición, ya que son los que concentran mayor discriminabilidad [48]. Además, en el enfoque más tradicional de CSP, la señal se filtra en una única banda frecuencial de 8 a 30 Hz, y se usan segmentos temporales de 0,5 a 2,5 s luego de presentada la indicación de la tarea para la decodificación [32].

En cuanto a la clasificación, LDA ha demostrado ser uno de los enfoques más populares en el contexto de las BCIs debido a su simplicidad y robustez. Además, sus bajos requerimientos computacionales lo hacen adecuados para sistemas BCI en línea [50]. Este

método se basa en encontrar una dirección que maximice la separación entre clases, maximizando la varianza entre clases y minimizando la varianza intra-clase. Sean $\boldsymbol{U}_b$ y $\boldsymbol{U}_w$ las matrices de covarianza inter-clase e intra-clase, respectivamente [51]. El método LDA busca encontrar la matriz de transformación lineal $\mathbf{D} \in \mathbb{R}^{M\times P}$, siendo M la dimensión del vector de características y $P < M$, mediante la maximización del funcional

$$J_{\text{LDA}}(\mathbf{D}) = \frac{\mathbf{D}^T \boldsymbol{U}_b \mathbf{D}}{\mathbf{D}^T \boldsymbol{U}_w \mathbf{D}}. \tag{2.4}$$

Para el caso de clasificación binaria, la solución explicita de este problema está dada por $\mathbf{D} = \boldsymbol{U}_w^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$, donde $\boldsymbol{\mu}_k$ denota la media de los vectores de características pertenecientes a la k-ésima clase [51].

CSP+LDA ha demostrado un desempeño robusto en escenarios intra sujeto y es ampliamente aceptado como el método estándar para la decodificación en MI-BCIs. No obstante, su capacidad de generalización entre sujetos es limitada, principalmente debido a la alta variabilidad en las señales de EEG [6, 35].

2.2.2. Modelos basados en aprendizaje profundo

EEGNet

EEGNet [42] es una red neuronal convolucional compacta específicamente diseñada para la decodificación en EEG-BCIs. Este modelo incorpora conceptos clásicos de extracción de características en EEG, como la combinación de información de diferentes canales y el filtrado en diferentes bandas de frecuencia, mediante el uso de convoluciones por canal (*depthwise*) y separables, y emplea técnicas como normalización por lotes y *dropout* a lo largo de la red. La arquitectura está compuesta por tres bloques, descriptos a continuación. Todo el diseño está pensado para señales de EEG con una frecuencia de muestreo original de 128 Hz.

- **Bloque 1:** Aplica una convolución temporal, obteniendo como salida F_1 mapas temporales, cada uno de los cuales es equivalente a la señal filtrada en una banda específica. Luego, un bloque de convolución por canal que utiliza un núcleo de tamaño $(C, 1)$ (siendo C el número de canales) actúa de forma independiente sobre cada uno de los F_1 mapas temporales, aprendiendo cómo combinar los canales para esa banda en particular. Para reducir la dimensión temporal de la señal, se incorpora una capa de *average pooling*. Para mejorar la generalización del modelo, se aplica

dropout durante el entrenamiento y se impone una restricción de norma máxima sobre los pesos w de los filtros espaciales aprendidos, asegurando que $\|w\|_2 < 1$.

- **Bloque 2:** Implementa una convolución separable, que consta de dos etapas. Primero, se realiza una nueva convolución por canal que actúa independientemente sobre cada uno de los mapas resultantes del bloque anterior. Esta operación permite resumir temporalmente la información contenida en una ventana. Luego, se aplica una convolución puntual (con un núcleo de tamaño $(1, 1)$), que aprende cómo combinar los distintos mapas de características generados previamente. Finalmente, se usa una capa de *average pooling* que continúa reduciendo la dimensión temporal.

- **Bloque 3:** Es el bloque de clasificación, en el cual las características extraídas ingresan directamente a una capa de clasificación softmax con K salidas, siendo K el número de clases.

El diseño de EEGNet busca no solo un buen desempeño en clasificación, sino también la posibilidad de interpretar los filtros aprendidos desde un punto de vista neurofisiológico. Se ha demostrado que los filtros temporales y espaciales pueden asociarse con patrones EEG bien conocidos, como desincronizaciones en la banda α en el caso de MI. Además, gracias a su tamaño reducido, EEGNet puede entrenarse con pocos datos, lo que permite utilizarla incluso en escenarios intra sujeto. Estas características la han convertido en uno de los modelos más utilizados para la decodificación de MI a partir de señales de EEG.

Deep ConvNet y Shallow ConvNet

Ambas redes fueron propuestas en el mismo trabajo [41] y consisten en arquitecturas convolucionales de diferente profundidad diseñadas para la decodificación de EEG.

Deep ConvNet es una red inspirada en el éxito de las redes convolucionales en visión computacional. Su rendimiento se ve notablemente mejorado por el uso de normalización por lotes y unidades lineales exponenciales (ELUs) como funciones de activación, y en el trabajo original demostró ser tan eficaz, o incluso superior, al método de *Filter Bank CSP* (FBCSP), utilizado como referencia en el área de MI-BCIs. En cuanto a su arquitectura, Deep ConvNet se compone de cuatro bloques de convolución-max-pooling. El primer bloque convolucional está dividido en dos capas: una para la convolución temporal y otra para el filtrado espacial. Esta separación no sólo permite manejar de forma más eficiente la gran cantidad de canales de entrada del EEG, sino que también impone una estructura más sencilla a la operación global, reduciendo la cantidad de parámetros a aprender y

ayudando a evitar el sobreajuste. A este le siguen tres bloques estándar de convolución-max-pooling. Finalmente, la red culmina con una capa de clasificación densa cuya salida se normaliza mediante una función softmax para obtener las probabilidades de cada clase.

Shallow ConvNet comprende una arquitectura más compacta. Esta red fue desarrollada específicamente para la decodificación de características de bandas de potencia y se inspira en el procesamiento de FBCSP. En cuanto a su arquitectura, las dos primeras capas de la shallow ConvNet realizan una convolución temporal y un filtrado espacial, al igual que en la deep ConvNet. Estos pasos son análogos al filtrado pasabanda y al filtrado espacial mediante CSP en FBCSP. A diferencia de la deep ConvNet, la convolución temporal de la shallow ConvNet utiliza un núcleo de mayor tamaño (25 frente a 10). Luego, se aplican una no linealidad cuadrática, una capa de *average pooling* y una función de activación logarítmica; en conjunto, estos pasos son análogos al cálculo del logaritmo de la varianza en las características de FBCSP. Es importante notar que estos pasos no se usan en la deep ConvNet. A diferencia de FBCSP, shallow ConvNet integra todos los pasos computacionales en una única red, permitiendo optimizar todos los pasos de forma conjunta. Además, debido a que incluye varias regiones de pooling dentro de un mismo ensayo, esta red puede aprender la estructura temporal de los cambios en la potencia de banda a lo largo del ensayo.

FBCNet

Filter Bank Convolutional Network (FBCNet) [52] es un modelo híbrido diseñado para la clasificación de MI-BCI a partir de señales de EEG en escenarios intra-sujeto. Este diseño tiene como objetivo extraer información espectro-espacial discriminativa para tareas de MI manteniendo una arquitectura compacta. FBCNet consta de cuatro etapas:

- **Representación en múltiples bandas frecuencias:** Esta primera etapa consiste en un filtrado del EEG en múltiples bandas de frecuencia estrechas (de 4 Hz cada una, entre 4–40 Hz). Esta estrategia permite capturar diferentes subbandas dentro de las bandas α y β de la señal de EEG, en las cuales se encuentra la información relevante para tareas de MI.

- **Filtrado espacial:** En este bloque se extraen patrones espaciales discriminativos para cada banda de frecuencia mediante convoluciones por canal con núcleos de tamaño $(C, 1)$, donde C es el número de canales.

- **Extracción de características temporales con una capa de varianza:** En

lugar de aplicar operaciones clásicas como promedio o máximo, FBCNet introduce una capa de varianza que produce una representación compacta de la información temporal de cada señal filtrada. Esta capa explota la relación entre varianza y potencia espectral (relacionada con patrones ERD/ERS), facilitando la extracción de información característica de la tarea de MI y reduciendo la dimensionalidad.

- **Clasificación :** Las características obtenidas se normalizan, se aplican funciones de activación y regularización, y finalmente se pasan a una capa densa con activación lineal y softmax para predecir las clases.

La arquitectura resultante es compacta, interpretable y en el trabajo original [52] obtuvo buen desempeño tanto en bases de datos públicas de sujetos sin patologías como en una base de datos de pacientes post ACV, mostrando mejoras significativas respecto a modelos clásicos (FBCSP) y redes convolucionales estándar (EEGNet, Deep ConvNet).

2.3. Métricas y pruebas estadísticas

2.3.1. Evaluación del desempeño de clasificación

Como forma estándar para medir el desempeño de clasificación se utilizó la tasa de aciertos, definida como la proporción de etiquetas correctamente predichas sobre el total de ensayos de evaluación.

Además, para evaluar el impacto de la estrategia de aumentación de datos propuesta, se utilizó el cambio relativo porcentual (RC) en la tasa de aciertos con respecto a su valor de referencia sin aumentación, definido como

$$RC = \frac{a_a - a_r}{a_r} 100\,\%, \tag{2.5}$$

donde a_a representa la tasa de aciertos en el conjunto de evaluación con aumentación de datos y a_r corresponde a la tasa de aciertos de referencia en el conjunto de evaluación sin aplicar la aumentación de datos. De esta forma, si $RC > 0$, la aumentación de datos mejora el rendimiento del modelo, mientras que si $RC < 0$, lo deteriora.

2.3.2. Distintividad entre clases

Para evaluar la separabilidad entre los ensayos pertenecientes a distintas clases de MI, se usó la distintividad entre clases (D) [53]. Esta métrica basada en geometría Rieman-

niana se calcula directamente a partir de los datos de EEG y se define como:

$$D = \frac{\delta_R\left(\overline{C}_1, \overline{C}_2\right)}{\frac{1}{2}\left(\sigma_{C_1} + \sigma_{C_2}\right)}, \tag{2.6}$$

donde $\overline{C}_k$ y σ_{C_k} corresponden, respectivamente, al promedio y a la desviación media absoluta de las matrices de covarianza para la clase k. A diferencia de la tasa de aciertos, D no depende de la salida de ningún clasificador, por lo que ofrece una medida intrínseca de cuán discriminables son los datos en términos de sus propiedades estadísticas.

2.3.3. Análisis estadístico

Con el fin de evaluar la significancia estadística de los resultados obtenidos, se emplearon pruebas no paramétricas adecuadas para comparar varios métodos de clasificación sobre los mismos conjuntos de datos [54].

En particular, se utilizó el test de Friedman [55], considerado el equivalente no paramétrico de ANOVA de medidas repetidas. Este test ordena los métodos evaluados para cada conjunto de datos: el método con mejor desempeño recibe el rango 1, el siguiente mejor el rango 2, y así sucesivamente. Posteriormente, se comparan los rangos promedio obtenidos por cada método a lo largo de todos los conjuntos de datos analizados. Bajo la hipótesis nula del test de Friedman, se asume que todos los métodos evaluados presentan un rendimiento equivalente, por lo que sus rangos promedio no difieren de manera significativa. Si los resultados permiten rechazar esta hipótesis, se concluye que al menos uno de los métodos presenta un desempeño diferente respecto de los demás.

Cuando se detectaron diferencias globales significativas, se procedío con comparaciones post hoc mediante el test de Nemenyi [56], análogo al test de Tukey en el contexto del ANOVA. Esta prueba permite determinar específicamente qué pares de métodos difieren significativamente entre sí, considerando un umbral crítico calculado en función del número de métodos y de conjuntos de datos evaluados.

Capítulo 3

Simulación de datos de MI-EEG fisiológicamente plausibles y su aplicación en aumentación de datos

3.1. Introducción

En este capítulo, se proponen aportes en dos áreas específicas: la generación de señales de EEG basada en modelos neurofisiológicos y la aumentación de datos durante el entrenamiento de modelos de DL aplicados a la decodificación en MI-BCIs. En particular, se presenta un enfoque de simulación de datos de MI-EEG realistas, construido mediante la incorporación de señales fisiológicamente significativas y específicas de la persona en modelos directos (*forward*) de EEG. Este desarrollo se asocia al primer, segundo y quinto objetivo específico de la tesis y dio origen a la publicación incluida en el Anexo A.

El enfoque propuesto, denominado PySimMIBCI, se puede utilizar para generar datos artificiales que son electrofisiológicamente similares a los datos reales de una determinada persona usuaria de MI-BCI. PySimMIBCI permite la simulación de distintas capacidades de modulación de MI y la introducción de diversas fuentes de variabilidad y artefactos en EEG, como fatiga y movimientos oculares. Cuando se incluye información específica de la persona usuaria, se convierte en una herramienta eficaz para la aumentación de datos en el contexto de modelos de DL para MI-BCIs. La implementación en Python de PySimMIBCI es de código abierto y está disponible de forma gratuita en github.com/catalinamagalvan/PySimMIBCI.

3.2. Metodología

3.2.1. Modelo biofísico para simulación de EEG

PySimMIBCI se basa en el enfoque de modelado biofísico de fuentes [57], el cual propone que el EEG se obtiene como combinación lineal de fuentes fisiológicas no observables [58]. Formalmente, sea $\boldsymbol{X} \in \mathbb{R}^{C \times N}$ un segmento de EEG, donde C representa la cantidad de canales y N es el número de muestras. Además, sea $\boldsymbol{G} \in \mathbb{R}^{C \times S}$, con S el número de fuentes, la matriz de conducción, que modela los efectos de la conducción tisular desde las fuentes neuronales hasta el cuero cabelludo y sea $\boldsymbol{Z} \in \mathbb{R}^{S \times N}$ la matriz que contiene las actividades de las fuentes corticales [59] no observables, incluyendo tanto señal de interés como componentes de ruido. Finalmente, sea $\boldsymbol{N} \in \mathbb{R}^{C \times N}$ una matriz de ruido superficial agregado a nivel de electrodos. El modelo de superposición lineal está dado por

$$\boldsymbol{X} = \boldsymbol{G}\boldsymbol{Z} + \boldsymbol{N}. \tag{3.1}$$

Este modelo directo vincula linealmente, a través de $\boldsymbol{G}$, los dipolos de corriente ubicados en los vértices de mallas corticales con los electrodos de EEG colocados en el cuero cabelludo. La construcción de dicha matriz de conducción puede variar en complejidad, desde un modelo de esfera simple hasta un modelo realista y específico de la persona [8, 57]. En este trabajo, para obtener $\boldsymbol{G}$, se emplea un modelo de cabeza basado en imágenes de resonancia magnética que incluye información morfológica y tiene en cuenta las propiedades de conductividad de los diferentes tejidos [60].

El enfoque descripto anteriormente muestra cómo los datos de EEG pueden modelarse desde una perspectiva biofísica. Sin embargo, para simular datos de EEG en el contexto de una BCI, es esencial incorporar información temporal y espacial específica asociada al paradigma de interés. Por lo tanto, para simular datos informativos desde la perspectiva del decodificador de MI, PySimMIBCI integra esta información en las fuentes $\boldsymbol{Z}$ utilizadas en el modelo generativo biofísico. Se consideran componentes independientes, cada uno de los cuales se caracteriza por tres atributos: eventos, forma de onda e información espacial, equivalentes a las propiedades *cuándo*, *qué* y *dónde*, propuestas en [8]. La interpretación de estas características es muy intuitiva: la información de eventos (*cuándo*) define los inicios y duraciones específicas de la actividad en una línea de tiempo; la información de forma de onda (*qué*) especifica el contenido de la señal introducido en cada evento (por ejemplo, sincronizaciones/desincronizaciones en bandas específicas del EEG); y la información espacial (*dónde*) indica la ubicación cortical de la actividad de la fuente.

3.2.2. Simulación de datos de MI de mano derecha e izquierda

A continuación se describe cómo puede construirse $\boldsymbol{Z}$ para un escenario de MI de mano izquierda vs. derecha. Estas fuentes artificiales se obtienen como una superposición de actividad relacionada a la tarea y actividad no relacionada con la tarea.

Actividad relacionada con la tarea. Es bien sabido que las tareas de MI de mano generan distintos patrones de sincronización (ERS) y desincronización (ERD) en las bandas α (8–13 Hz) y β (14–30 Hz) del EEG sobre la corteza sensorimotora [61, 62]. Una ERD corresponde a una disminución de la potencia, mientras que una ERS representa un aumento de la potencia en relación con un período de referencia. PySimMIBCI considera para la simulación la ERD típica en la banda α en el área motora contralateral de la mano [63, 64]. Los datos se generan como señales con distribución normal estándar [65], filtradas para restringir el contenido a una sub-banda α estrecha, con una frecuencia central y un ancho de banda específicos. Esta actividad periódica está presente en ambas áreas motoras de la mano durante todo el experimento, con una reducción de la amplitud (ERD) en el lado contralateral durante cada tarea de MI. Para simular el protocolo de adquisición, se genera una línea de tiempo artificial con ensayos de cada clase ordenados aleatoriamente. En cuanto a la información espacial, los generadores son dipolos ubicados en parches esféricos de 30 mm de radio en los centroides de las parcelas del área prefrontal derecha e izquierda. Para la localización de dichas estructuras surco-girales en la corteza se emplea la parcelación `aparc.a2009s` [66], disponible en *FreeSurfer*[1].

Actividad no relacionada con la tarea. PySimMIBCI incluye la simulación de ruido de fondo y de dos tipos de artefactos oculares, como en [44]. El ruido de fondo comprende actividad aperiódica y consiste en un componente espectralmente $1/f^{\lambda}$ relacionado con procesos neuronales basales [9]. Este componente está presente de forma constante en toda la superficie cortical para todas las frecuencias f. Para generarlo, se filtra temporalmente ruido blanco gaussiano no correlacionado espacial ni temporalmente de manera que tenga una densidad espectral de potencia (PSD) específica según el exponente de ruido, λ. Luego, mediante una transformación de coloreado [67], estos componentes se fuerzan a tener una matriz de covarianza determinada por la proximidad de las fuentes (a mayor cercanía de las fuentes, mayor correlación).

Por otra parte, se agregaron dos tipos de artefactos oculares, específicamente de parpadeo y movimiento ocular. Para el artefacto de parpadeo, los inicios de activación se

[1] `http://surfer.nmr.mgh.harvard.edu/`

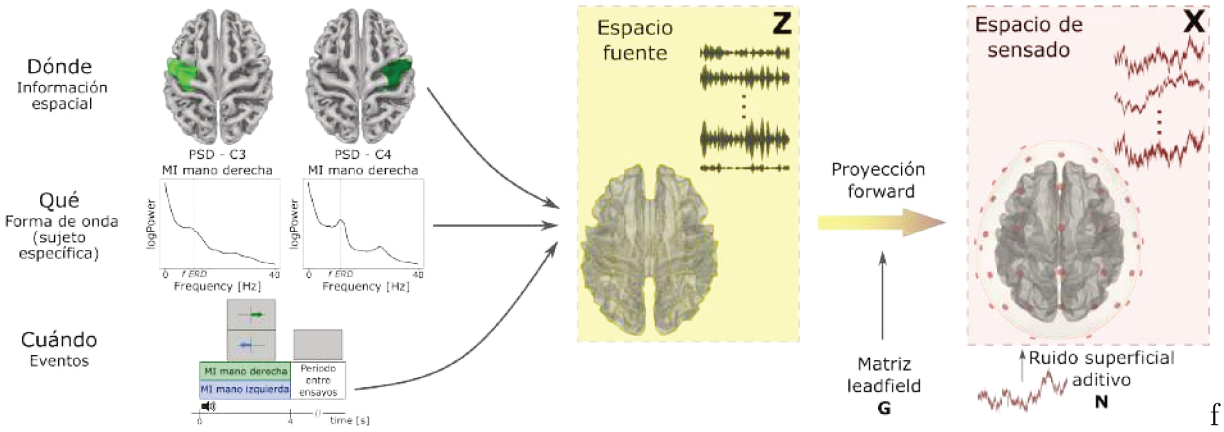

Figura 3.1: Ejemplo de simulación de datos de EEG mediante PySimMIBCI para la tarea de MI mano derecha. Los datos en el espacio fuente se generan mediante la combinación de información espacial, forma de onda y eventos relacionados con la tarea de la BCI. Luego, se utiliza una proyección hacia adelante para obtener los datos en el espacio sensor y, en este nivel, se puede añadir ruido superficial.

muestrean a partir de un proceso de Poisson no homogéneo con una tasa que oscila entre 4.5 y 17 parpadeos/minuto [68]. El núcleo de activación consiste en una ventana de Hanning de 250 ms. Los dos dipolos generadores se ubican en el plano $z = 0$ a $\pm 30°$ del plano medio (nasion). Los artefactos por movimiento ocular también se modelan como un proceso de Poisson. En cada activación, el movimiento ocular se genera como una señal aleatoria con una duración aleatoria de máximo 500 ms y los dipolos generadores se ubican de la misma manera que en la implementación del artefacto de parpadeo.

La Figura 3.1 resume PySimMIBCI para la tarea de MI de mano derecha. La actividad en el espacio de las fuentes Z se genera combinando información específica tanto en las características espaciales, como en la forma de onda y de eventos. En este ejemplo, la información espacial incluye las áreas motoras de la mano derecha e izquierda, y la forma de onda relacionada a la tarea es la ERD característica en la banda α. La información de eventos incluye una línea de tiempo típica en protocolos de MI: 4 s de MI seguidos por una pausa de duración variable entre ensayos. A través de una proyección mediante una matriz $\boldsymbol{G}$ esta información puede accederse a nivel de los sensores, donde se puede agregar ruido superficial mediante una matriz $\boldsymbol{N}$, resultando en la señal simulada $\boldsymbol{X}$.

3.2.3. Simulación de diferentes capacidades de modulación

Durante el desarrollo de métodos de decodificación de EEG, suele resultar útil contar con datos que correspondan a diversos niveles de habilidad en el control de una MI-BCI

para probar hipótesis y realizar ajustes. Con este fin, en este trabajo se proponen dos estrategias para generar personas artificiales con diferentes características de modulación de MI. La primera estrategia consiste en modificar el porcentaje de α ERD en el área motora de la mano contralateral descrita en la Sección 3.2.2. Los porcentajes más altos de ERD indican una modulación más fuerte y, por lo tanto, corresponderían a personas más competentes, mientras que ERDs menos pronunciados se asociarían con personas con una menor capacidad de modulación. La segunda estrategia simula escenarios en los que la persona no realiza la tarea de MI requerida. Esto se realiza incluyendo una cierta proporción de intentos fallidos de MI, es decir, intentos sin ERD en el área correspondiente. Pueden simularse personas artificiales con diferentes niveles de competencia para controlar una MI-BCI usando estas ideas por separado o conjuntamente.

3.2.4. Inclusión de efectos de fatiga

Otro aspecto importante a la hora de simular datos de MI-EEG es que la participación prolongada en tareas cognitivamente exigentes, como la MI, puede conducir a la fatiga mental [69], la cual se acompaña de cambios en la actividad cerebral que dan lugar a modificaciones en el EEG [70]. Con el fin de generar señales MI-EEG realistas que reflejen los efectos relacionados con la fatiga, aquí se propone una forma de incluirlos en los datos simulados. Varios estudios han encontrado que tanto la potencia de la banda θ (4-8 Hz) como de la banda α (8-13 Hz) en el EEG aumentan con la fatiga mental [71-73]. Más específicamente, en [74, 75] la fatiga mental se ha asociado con incrementos progresivos tanto en la potencia θ frontal como en la potencia α parietal a medida que avanza la tarea. Por lo tanto, estos marcadores sirven como una manera de simular la fatiga mental. En PySimMIBCI, se asume que los generadores cerebrales de la actividad θ y α son dipolos en las áreas frontal y parietal, respectivamente. Al igual que para la actividad α relacionada a la tarea, estos componentes se generan como señales con distribución normal estándar filtradas para obtener las actividades en las bandas de interés. Este efecto de fatiga puede simularse a partir de cualquier instante específico y su nivel se incrementa linealmente con el tiempo.

3.2.5. Simulación de datos de MI-EEG personalizados para aumentación de datos

Esta sección describe el enfoque propuesto para la inclusión de características específicas de la persona en la simulación de datos de MI-EEG, con el fin de utilizarlos en la

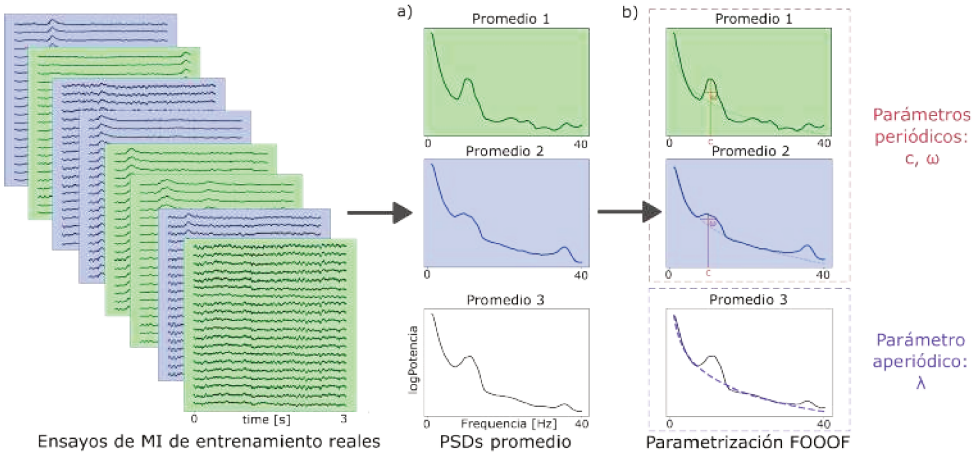

Figura 3.2: Extracción de parámetros específicos de la persona usuaria. a) Primero, se calculan las PSD promedio a partir de los ensayos de entrenamiento. b) Luego, se realiza la parametrización periódica y aperiódica sobre cada PSD promedio usando FOOOF [76].

aumentación de datos. En primer lugar se calcula la PSD de los ensayos reales de entrenamiento de MI-EEG para cada canal [77]. Como se muestra en la Fig. 3.2 a), nos interesan tres PSDs promedio: Promedio 1, el de los ensayos de MI de la mano derecha para el canal C4; Promedio 2, el de los ensayos de MI de la mano izquierda para el canal C3; y Promedio 3, el de todos los ensayos y de todos los canales. Luego, se utiliza el método *fitting oscillations and one-over f* (FOOOF) [76] para obtener los parámetros periódicos y aperiódicos de los distintos PSDs promedios, como se ve en la Fig. 3.2 b).

Es razonable pensar que la actividad aperiódica se distribuye uniformemente en la corteza y no depende de tareas específicas de MI. Por lo tanto, el componente aperiódico se parametriza a partir de la PSD promedio sobre todos los ensayos y canales (Promedio 3). En cambio, considerando que los dos canales de EEG principales en el área motora de la mano son C3 y C4 [64], sólo estos se utilizan para ajustar los componentes periódicos. Es decir, las PSDs utilizadas para extraer los parámetros periódicos se calculan sobre los ensayos de entrenamiento de cada clase para ambos canales (Promedio 1 y 2).

Luego de extraer estos parámetros de datos de MI-EEG reales, se simulan datos personalizados como se describe a continuación: El exponente aperiódico λ se utiliza para generar la actividad de fondo y la ERD en la banda α se genera a partir de los parámetros de frecuencia central c y ancho de banda ω del pico más alto en el rango 7–14 Hz. Si la persona no presenta un pico detectable en la banda α, se considera un pico genérico con $c = 11{,}5$ Hz y $\omega = 3$ Hz, de acuerdo con el pico α más frecuente entre todas las personas reales (ver Fig. A1 en el Material Suplementario del Anexo A).

	ERD [%]	Ensayos fallidos [%]
ideal	50	0
S40	40	0
S30	30	0
S20	20	0
S10	10	0
SF10	50	10
SF20	50	20
SF30	50	30
SF40	50	40

Tabla 3.1: Parámetros de simulación para personas artificiales con diferentes capacidades de modulación.

3.3. Experimentos y resultados

Con el objetivo de evaluar la calidad de las señales generadas por PySimMIBCI y su utilidad como estrategia de aumentación de datos para modelos de decodificación de MI-EEG, se realizaron distintos experimentos utilizando las Bases de datos reales 1 y 2 presentadas en el Capítulo 2. Para estas pruebas se utilizó el modelo de decodificación FBCNet. En todos los experimentos, la arquitectura se entrenó utilizando el optimizador Adam con la configuración predeterminada y la función de pérdida entropía cruzada, siguiendo el trabajo original [52]. El modelo se entrenó utilizando early stopping con una paciencia de 200 épocas. Se seleccionó el modelo con la mayor tasa de aciertos en validación. El número máximo de épocas de entrenamiento se limitó a 1500.

3.3.1. Evaluación de la calidad de las señales simuladas

Tal como se describe en la Sección 3.2.3, se simularon personas usuarias con diferentes características de modulación y posteriormente se compararon con datos reales de la Base de datos 1. El sujeto g de esta base de datos se usó como referencia, por ser el que había obtenido el mejor desempeño de clasificación en el estudio original [44]. Se extrajeron sus parámetros neurofisiológicos siguiendo los pasos descritos en la Sección 3.2.5 y para cada persona simulado se generaron 200 ensayos (100 por clase). Primero, el porcentaje de ERD se varió desde un 50 % (persona usuaria ideal [44]) hasta un 10 % (S10) en decrementos de 10 %. Luego, se simularon casos en los que no realizan la tarea requerida incluyendo una proporción creciente de ensayos sin ERD, desde un 10 % (sujeto SF10) hasta un 40 % (sujeto SF40). En estos últimos casos, el porcentaje de ERD se fijó en 50 %. En la Tabla 3.1 se presenta un resumen de los parámetros de simulación para cada sujeto.

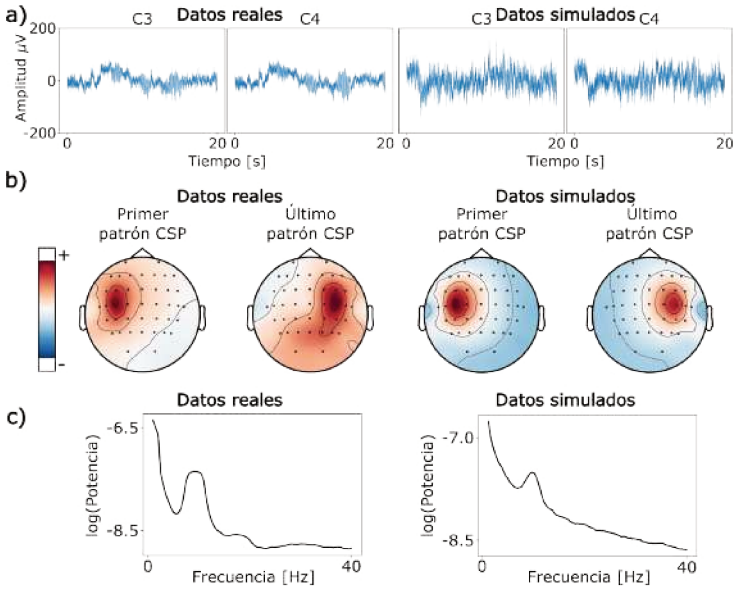

Figura 3.3: Gráficas temporales, espaciales y espectrales para datos de EEG reales y simulados. a) 20 s de ambas señales en los canales C3 y C4. b) Primer y último patrón de CSP. c) PSDs promedio.

Para evaluar cualitativamente la similitud entre las señales generadas artificialmente y los datos reales de MI-EEG, se realizó una inspección visual basada en información temporal, espectral y espacial (patrones de CSP). Se compararon los 200 ensayos correspondientes al sujeto simulado ideal con los 200 ensayos de la sesión 1 del sujeto g de la Base de datos 1. La Fig. 3.3 muestra dichas comparaciones. El panel (a) muestra 20 s de señales simuladas y reales en los canales C3 y C4. El panel (b) muestra las topografías del primer y último patrón de CSP en cada caso. Finalmente, en el panel (c) se muestra la PSD promedio para los 200 ensayos para los canales C3 y C4. Como puede observarse gráficamente, las características temporales, espaciales y espectrales de las señales MI-EEG simuladas resultan muy similares a las de sus contrapartes reales.

Por otra parte se evaluó el desempeño de clasificación de FBCNet usando los datos MI-EEG simulados y se lo comparó con el obtenido con datos reales en un escenario de validación cruzada (CV) de 10 particiones (8 particiones para entrenamiento, 1 para validación y la restante para evaluación). Sólo la sesión 1 (200 ensayos) de datos reales fue parte de este análisis. En cuanto a los datos simulados, se consideraron los 200 ensayos de cada uno de los nueve sujetos simulados de la Tabla 3.1. La Fig. 3.4 (a) muestra los resultados de este análisis de CV. Como puede verse, los datos simulados pueden alcanzar un desempeño de clasificación similar al de los datos reales de MI-EEG. Además, como era esperado, porcentajes más bajos de ERD y mayores proporciones de ensayos fallidos

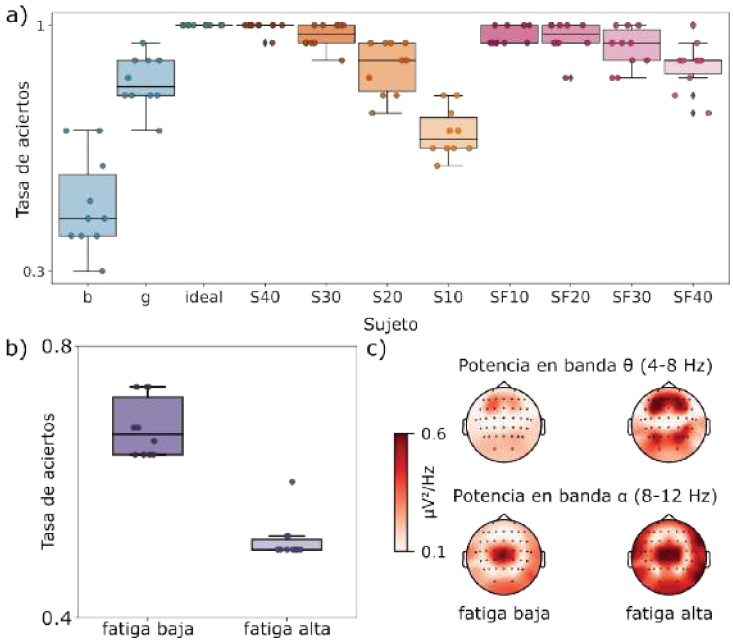

Figura 3.4: Resultados en datos simulados con diferentes capacidades de modulación y efectos de fatiga. a) Tasa de aciertos en una partición de evaluación en un escenario de validación cruzada. Los puntos celestes representan los resultados obtenidos para las dos personas reales, y los puntos verdes, el sujeto simulado ideal. Los sujetos simulados con porcentajes decrecientes de ERD se muestran en una paleta naranja, mientras que se utilizó una paleta rosa para los sujetos simulados con porcentajes crecientes de intentos fallidos. b) Tasa de aciertos con baja y alta fatiga, usando un modelo entrenado sin fatiga. c) Mapas topográficos en las bandas θ y α para ambas condiciones de fatiga.

obtienen un menor desempeño de clasificación.

Para mostrar cómo los efectos de la fatiga pueden influir en el rendimiento de la decodificación, se llevó a cabo el siguiente experimento: la primera mitad de la sesión descripta en la Sección 3.2.4, donde no había efectos de fatiga, se utilizó para entrenar FBCNet. Los datos de la segunda mitad de la sesión se dividieron en dos particiones de evaluación: una con efectos bajos y otra con efectos altos de fatiga. La Fig. 3.4 (b) muestra la tasa de aciertos obtenida para cada partición. Se observa una disminución notable en la tasa de aciertos cuando se simulan efectos de fatiga mental. Además, la Fig. 3.4 (c) ilustra los mapas topográficos en las bandas θ y α. Como era esperado, estos gráficos muestran que, a medida que el nivel de fatiga mental simulada aumenta a lo largo de la sesión, también aumentan las potencias en las bandas θ y α en las áreas frontal y parietal.

3.3.2. PySimMIBCI como estrategia de aumentación de datos

Para la aumentación de datos mediante PySimMIBCI, se generaron 100 ensayos por clase para cada persona real utilizando la configuración ideal (50 % de ERD y 0 % de ensayos fallidos) y parámetros neurofisiológicos personalizados, como se describe en la Sección 3.2.5. El impacto de esta estrategia se evaluó en un escenario intra sujeto, entre sesiones (entrenamiento en sesión 1, evaluación en sesión 2). Del conjunto de entrenamiento, se seleccionaron 25 ensayos por clase para validación. Para tener en cuenta la variabilidad asociada a la inicialización del modelo, el proceso se repitió 10 veces con diferentes semillas aleatorias. Se comparó el desempeño del modelo con y sin aumentación, usando el cambio relativo porcentual (RC) respecto a la tasa de aciertos sin aumentación. PySimMIBCI se comparó con otras estrategias de aumentación del estado del arte, para cada una de las cuales también se generaron 200 ensayos. A continuación se describen las mismas brevemente.

- Segmentación y recombinación (SR) [78]: los ensayos de entrenamiento (2 s) se dividen en 8 segmentos y se crean nuevos ensayos concatenando segmentos análogos aleatorios de ensayos de la misma clase.
- Agregado de ruido gaussiano (GNA) [21]: se generan copias de cada ensayo de entrenamiento agregando ruido gaussiano (media cero y desvío estándar 0,2).
- Enmascarado temporal (TM) [20]: los ensayos aumentados se obtienen haciendo cero todos los canales de segmentos temporales aleatorios de 100 muestras de datos de MI-EEG reales.
- Dropout de canales (CD) [22]: se obtienen nuevos ensayos de entrenamiento haciendo cero canales elegidos aleatoriamente de los ensayos originales. La probabilidad de eliminar cada canal es 0,2.
- Filtrado rechaza banda (BF) [20]: se generan nuevos ensayos aplicando un filtro rechaza banda de 5 Hz de ancho, con frecuencia central aleatoria entre 0 y 40 Hz .
- Wasserstein GAN (WGAN): el modelo de [79] se adaptó para la generación de datos MI-EEG. El entrenamiento se realizó de manera específica para cada persona.

Para todas las estrategias de aumentación de datos evaluadas, los lotes de entrenamiento se construyen con un número fijo de ensayos reales y simulados, garantizando siempre el balance entre clases. Más precisamente, cada lote contiene N_b ensayos y tiene

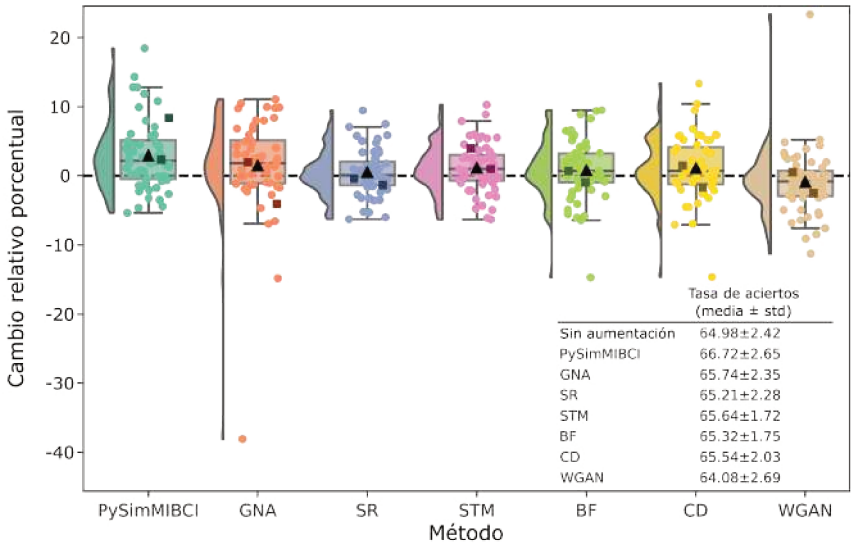

Figura 3.5: Resultados de aumentación de datos en dos bases de datos reales de MI-EEG. Cada punto corresponde al valor medio de RC a lo largo de 10 semillas aleatorias para una persona real (cuadrados oscuros: Base de datos 1, círculos claros: Base de datos 2). El valor promedio de RC para todas las personas se representa con un triángulo negro.

la siguiente estructura: N_a ensayos aumentados por clase, seleccionados aleatoriamente del total de ensayos generados, y $N_r = (N_b - 2N_a)/2$ ensayos reales seleccionados aleatoriamente de cada clase. Este enfoque de muestreo introduce una variabilidad valiosa, que ejerce un efecto de regularización en el entrenamiento del modelo. Al mismo tiempo, conserva el balance de clases dentro de cada lote y mantiene constante la proporción de ensayos reales/simulados. Con estas consideraciones, aseguramos que el modelo se beneficie de ejemplos de entrenamiento diversos mientras preserva una composición de datos consistente entre diferentes lotes (véase la Fig. A2 en el Material Suplementario del Anexo A para observar el efecto de distintos métodos de muestreo en la construcción de lotes). Se usó $N_a = 1$, ya que no se encontraron diferencias al agregar más ensayos simulados en cada lote de entrenamiento (véase la Fig. A3 en el Material Suplementario del Anexo A).

Los resultados correspondientes se muestran en la Fig. 3.5, en la cual el RC promedio a lo largo de las 10 semillas aleatorias se grafica para un total de 56 personas. PySimMIBCI alcanza el valor promedio de tasa de aciertos absoluta más alto (66,72 %±2,65 %). Además, es notable que PySimMIBCI es el único enfoque que no produce un RC negativo mayor al 5 % para ningún caso. El análisis estadístico (test de Friedman y post-hoc de Nemenyi) confirma que PySimMIBCI supera significativamente a las demás estrategias (excepto GNA, con la que resulta comparable). En la Sección 6 del Material Suplementario del Anexo A se presentan los resultados para cuatro modelos de DL adicionales.

Capítulo 4

Transferencia de aprendizaje entre sujetos

4.1. Introducción

Este capítulo aborda uno de los desafíos más importantes que limitan la aplicabilidad de las MI-BCIs: la marcada variabilidad intersujeto presente en las señales de EEG, la cual dificulta la generalización de los modelos de decodificación entrenados sobre un grupo de usuarios a sujetos no vistos. En este contexto, el presente trabajo propone XS-BOT, un método de adaptación de dominio basado en transporte óptimo hacia atrás, diseñado para mejorar la usabilidad de BCIs en entornos de rehabilitación, reduciendo al mínimo la cantidad de datos que deben recopilarse del nuevo usuario antes de permitirle el control en tiempo real del sistema. En estos escenarios, donde la tarea de MI a realizar es indicada por el sistema pero la tarea mental que realmente realiza el usuario es incierta y debe decodificarse, XS-BOT aprovecha la disponibilidad esta etiqueta como una valiosa señal de supervisión para guiar la adaptación. Partiendo de un modelo de decodificación de MI previamente entrenado en un grupo de sujetos, XS-BOT se aplica en el espacio de características, acercando la distribución de las características del sujeto objetivo a la distribución de características del dominio fuente o de entrenamiento, evitando la necesidad de recalibrar el modelo de decodificación. Nuestro enfoque utiliza sólo un pequeño número de ensayos etiquetados del sujeto objetivo para estimar el plan de transporte óptimo. Los resultados en diferentes bases de datos, que comprenden en total 139 sujetos de evaluación, muestran que XS-BOT puede mejorar sustancialmente la decodificación de MI, superando en aproximadamente un 20 % a los métodos de adaptación competidores,

tanto para modelos de decodificación basados en ML estadístico como en DL, y utilizando datos provenientes de sólo 3 canales de EEG. Esto vuelve a XS-BOT un método adecuado para una calibración rápida en el contexto de MI-BCIs para rehabilitación, impulsando el uso práctico de estos sistemas. Este enfoque contribuye directamente a los objetivos específicos 3, 4 y 5 de la tesis y dio origen a una publicación actualmente en revisión, cuyo manuscrito completo se incluye en el Anexo B.

4.2. XS-BOT: adaptación de dominio entre sujetos y bases de datos basada en transporte óptimo

La variabilidad del EEG en las MI-BCIs las convierte naturalmente en un problema de adaptación de dominio, en el cual el objetivo es abordar la caída en el desempeño que ocurre cuando un modelo entrenado con datos de un dominio fuente se aplica a datos de un dominio objetivo con una distribución diferente, aunque parecida [80]. Entre las técnicas de adaptación de dominio, BOTDA [10] es un método que utiliza principios del transporte óptimo para alinear distribuciones de datos aplicando un mapeo inverso desde el espacio de características objetivo al espacio de características fuente. Dicho método fue originalmente propuesto para la adaptación entre sesiones dentro de un mismo usuario de MI-BCIs [10].

XS-BOT extiende BOTDA a un escenario entre sujetos y entre bases de datos, caracterizado por marcados cambios de distribución debidos a la variabilidad entre individuos y las configuraciones de adquisición heterogéneas. En este escenario, el dominio fuente Ω_s se refiere a una base datos grande y diversa que contiene datos etiquetados de múltiples sujetos, utilizada para entrenar el modelo de decodificación, mientras que el dominio objetivo Ω_t corresponde a un nuevo sujeto, proveniente de una base de datos no vista durante el entrenamiento. Ambos dominios tienen un conjunto de datos asociado

$$\mathcal{S} \doteq \left\{\left(\mathbf{x}_j^s, y_j^s\right)\right\}_{j=1}^{N_s} \subset \Omega_s \times \mathcal{K}, \quad \mathcal{T} \doteq \left\{\left(\mathbf{x}_i^t, y_i^t\right)\right\}_{i=1}^{N_t} \subset \Omega_t \times \mathcal{K},$$

donde N_s y N_t son el número de ensayos en los conjuntos de datos fuente y objetivo, respectivamente. Los vectores $\mathbf{x}_j^s \in \Omega_s$, $\mathbf{x}_i^t \in \Omega_t$ son características extraídas de las señales de EEG, donde $\Omega_s, \Omega_t \subset \mathbb{R}^d$, y $y_j^s, y_i^t \in \mathcal{K}$ representan las tareas mentales indicadas, pertenecientes a K posibles clases de MI, con $\mathcal{K} \doteq \{k_l\}_{l=1}^K$. Es importante notar que, debido a la naturaleza de las MI-BCIs, estas etiquetas no necesariamente constituyen verdadera la actividad mental realizada, sino la tarea indicada por el sistema [81].

La adaptación de dominio entre Ω_t y Ω_s se aborda estimando un plan de transporte óptimo γ^* que permite mapear las características del sujeto objetivo a la distribución de características del dominio fuente. Esto se realiza considerando la formulación discreta Sinkhorn LASSO por grupos del problema de trasporte óptimo

$$\gamma^* \doteq \underset{\gamma}{\operatorname{argmin}} \langle \gamma, M \rangle_F + \lambda W_e(\gamma) + \nu W_c(\gamma), \tag{4.1}$$

donde $\langle \cdot, \cdot \rangle_F$ denota el producto interno de Frobenius y $M = (m_{i,j})$, $m_{i,j} \geq 0$, es una matriz tal que $m_{i,j} = m\left(\mathbf{x}_i^t, \mathbf{x}_j^s\right)$ representa el costo de transportar una unidad de masa de probabilidad desde $\mathbf{x}_i^t$ hasta $\mathbf{x}_j^s$, dada una función de costo m. Una elección común para m es el cuadrado de la distancia euclídea [82]. En (4.1), $W_e(\gamma)$ y $W_c(\gamma)$ actúan como regularizadores y λ y ν son sus respectivos hiperparámetros. Específicamente, $W_e(\gamma) \doteq \sum_{ij} \gamma_{i,j} \log(\gamma_{i,j})$ corresponde a la negentropía, que favorece planes de transporte más suaves y menos ralos, mientras que $W_c(\gamma) \doteq \sum_j \sum_k ||\gamma_{\mathcal{I}_k,j}||_2$, con $\mathcal{I}_k$ denotando el conjunto de índices en el dominio objetivo que pertenecen a la clase $k \in \mathcal{K}$, es el término LASSO por grupos, que fomenta que las muestras de la misma clase se transporten como grupo [83].

La estimación de γ^* requiere datos de ambos dominios. Supongamos que se cuenta con un pequeño número N_v de ensayos de adaptación del usuario objetivo etiquetados. Usando este conjunto de adaptación, se selecciona un subconjunto del dominio fuente del mismo tamaño, N_v, y los parámetros de regularización, λ y ν, que maximizan el desempeño del transporte en los datos de adaptación. Tanto el subconjunto del dominio fuente como los parámetros de regularización permanecen fijos durante la adaptación. Para cada ensayo ℓ, se realiza la adaptación construyendo un conjunto de transporte $\mathcal{V}_\ell$, que combina el conjunto de adaptación fijo con el ensayo actual

$$\mathcal{V}_\ell \doteq \left\{\left(\mathbf{x}_i^t, y_i^t\right)\right\}_{i=1}^{N_v} \cup \left\{\left(\mathbf{x}_\ell^t, y_\ell^t\right)\right\} \subset \mathcal{T}. \tag{4.2}$$

Una vez estimado γ^*, y dado que el ensayo actual ℓ fue parte del conjunto de transporte, su vector de características transformado $\hat{\mathbf{x}}_\ell^t$ puede obtenerse directamente mediante el cálculo de baricentro ponderado

$$\hat{\mathbf{x}}_\ell^t = \sum_j \gamma_{\ell,j}^* \mathbf{x}_j^s, \tag{4.3}$$

donde los pesos $\gamma_{\ell,j}^*$ determinan cuánta masa del ensayo del dominio objetivo $\mathbf{x}_\ell^t$ se transporta a cada ensayo del dominio fuente $\mathbf{x}_j^s$. Finalmente, $\hat{\mathbf{x}}_\ell^t$ puede usarse para alimentar al clasificador (ya entrenado) y predecir su etiqueta de clase. Una ventaja clave de este

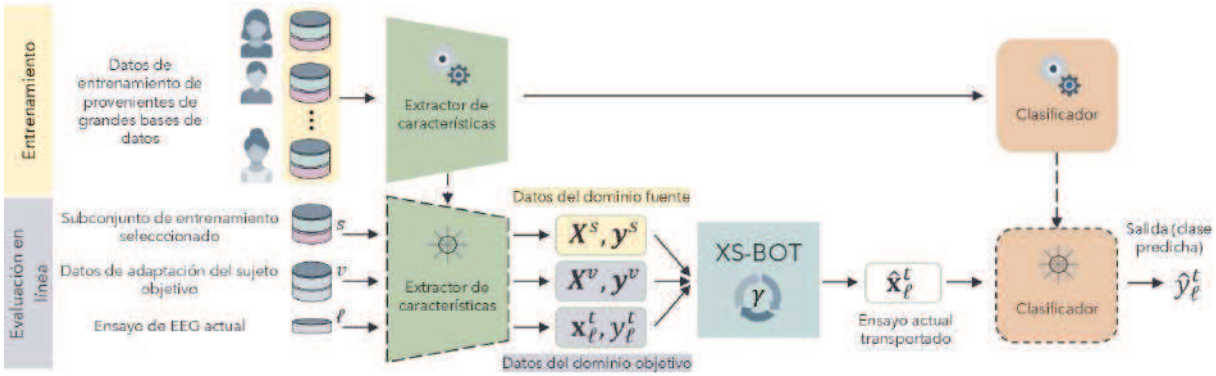

Figura 4.1: Fases de entrenamiento y evaluación en línea de XS-BOT. El modelo de decodificación se entrena con un gran conjunto de datos de múltiples sujetos. Durante la evaluación en línea para un ensayo ℓ, se aprende un plan de transporte óptimo γ^* utilizando ensayos de adaptación del sujeto objetivo, junto con el ensayo actual y su etiqueta indicada. Las características del ensayo actual se transforman usando γ^* y luego se ingresan al clasificador previamente entrenado.

enfoque es que evita el reentrenamiento del modelo, permitiendo una adaptación computacionalmente eficiente e implementable en tiempo real.

En este enfoque, la etiqueta indicada para el ensayo actual ℓ se aprovecha al aprender γ^*, específicamente sólo dentro del término de regularización LASSO por grupos. Sin embargo, aunque este término promueve que los ensayos del dominio objetivo en el mismo grupo se transporten juntos, dicho transporte puede fallar cuando hay una discrepancia entre la tarea indicada y la tarea mental realmente realizada [81].

En resumen, como se muestra en la Figura 4.1, el primer paso del enfoque propuesto, XS-BOT, consiste en entrenar un modelo de decodificación utilizando datos de múltiples sujetos, típicamente de bases de datos públicas. De esta forma se busca que el modelo aprenda características generalizables a través de una variedad de individuos [36, 84]. Luego, para cada sujeto objetivo, se utiliza un pequeño conjunto de adaptación (típicamente 10 ensayos por clase) para seleccionar un subconjunto del dominio fuente y los parámetros de regularización óptimos, siguiendo un enfoque basado en datos (ver Material Suplementario del Anexo B para más detalles). Durante la fase de evaluación, el subconjunto del dominio fuente seleccionado, el conjunto de adaptación y los parámetros de regularización permanecen fijos. Para cada ensayo, se realiza la adaptación y la predicción de su clase siguiendo el siguiente esquema aplicable en tiempo real. Se extrae el vector de características correspondiente, $\mathbf{x}_\ell^t$, se calcula el plan de transporte óptimo γ^* a partir del conjunto de transporte $\mathcal{V}_\ell$ definido en (4.2) y del subconjunto fuente seleccionado, se transforman las características del ensayo según γ^* de modo que coincidan con la distribución del dominio

fuente, y finalmente el clasificador previamente entrenado se usa para predecir la clase a partir de dicho vector transformado.

4.3. Experimentos y resultados

Los experimentos se llevaron a cabo en las Bases de datos 2, 3 y 4 presentadas en el capítulo 2. La Base de datos 2 [45] (54 sujetos) se utilizó para entrenamiento, mientras que la evaluación se realizó sobre las Bases de datos 2 [46] (52 sujetos), 3A (60 sujetos), 3B (21 sujetos) y 3C (6 sujetos) [47]. La elección de la Base de datos 2 para entrenamiento se basó en resultados previos que mostraron que este conjunto de datos era el mejor "donante" al ser evaluado en diferentes de bases de datos de MI-BCI [84].

A todas las bases de datos se les aplicó el mismo preprocesamiento. Las señales crudas fueron segmentadas en ensayos desde 0 a 3 segundos después de la señal de instrucción de MI. Para asegurar una configuración común, las señales se re-referenciaron al canal Fz. Cada ensayo fue filtrado entre 4 y 40 Hz, remuestreado a 128 Hz y recortado al intervalo entre 0.5 y 2.5 s, el cual sirvió como entrada para el modelo de decodificación. Para el análisis, sólo se seleccionaron los canales C3, Cz y C4, con el fin de utilizar una mínima cantidad de electrodos en áreas relevantes para MI de mano [84].

XS-BOT fue aplicado en dos enfoques de decodificación de MI base: uno tradicional basado en CSP seguido de LDA, y uno basado en DL mediante EEGNet. Para CSP+LDA, los ensayos de EEG fueron filtrados en banda entre 8–30 Hz [43]. Luego, se utilizaron dos componentes de CSP para extraer las características, calculadas como el logaritmo de la potencia promedio de las señales proyectadas. Las dos características resultantes se usaron para alimentar a un clasificador LDA. Para los modelos de aprendizaje profundo, cada ensayo fue estandarizado utilizando su propia media y desviación estándar antes de la entrada al modelo. Para EEGNet, utilizamos la arquitectura EEGNet-8,2 tal como se propone en el trabajo original [42]. Se extrajeron las características latentes antes de la clasificación. EEGNet fue entrenada con AdamW ($lr = 0{,}001$) y una función de pérdida de entropía cruzada durante 500 épocas. En el material suplementario del Anexo B se brindan más detalles sobre las arquitectura y los hiperparámetros utilizados.

XS-BOT fue evaluado en un escenario entre sujetos y entre bases de datos, con mínima cantidad de datos de adaptación del sujeto objetivo. La Base de datos 1 fue utilizada completa para el entrenamiento, sin reservar datos para validación. Para cada sujeto objetivo en las Bases de datos 2 y 3, se utilizó un pequeño subconjunto etiquetado de ensayos $N_v = 20$ (los primeros 10 ensayos de cada clase) como conjunto de adaptación. El

valor de N_v coincide con el utilizado en el trabajo original de BOTDA [10]. Los ensayos restantes se reservaron para evaluación, a fin de simular la implementación en línea.

El desempeño de XS-BOT fue comparado contra las siguientes estrategias base y métodos de adaptación, implementados bajo las mismas condiciones experimentales:

- Entrenamiento con datos fuente (SDT): el modelo se entrena con datos fuente de múltiples sujetos sin ninguna adaptación al sujeto objetivo.
- Entrenamiento con datos de adaptación (ADT): el modelo se entrena exclusivamente con los pocos ensayos de adaptación disponibles del sujeto objetivo.
- Alineación euclídea (EA) [39]: los datos se alinean en el espacio euclídeo antes del entrenamiento y la evaluación. Para cada sujeto del dominio fuente, la alineación se realiza independientemente mediante la matriz de covarianza media calculada sobre todos los ensayos de ese sujeto. Para el sujeto objetivo, cada ensayo se alinea usando la media aritmética de sus ensayos de adaptación.
- CSP con recentrado genérico (GR-CSP) [17]: alinea las matrices de covarianza de los ensayos con la media riemanniana del conjunto de entrenamiento. Durante la evaluación, las características se extraen de las matrices de covarianza alineadas utilizando una transformación afín con una matriz de centrado actualizada incrementalmente.
- Estrategias de ajuste fino para modelos de DL [37, 38]: los modelos preentrenados con datos fuente se ajustan con datos de adaptación del objetivo. El ajuste fino completo (FFT) actualiza todas las capas, mientras que el ajuste fino sólo del clasificador (CFT) actualiza únicamente las capas utilizadas para la clasificación.

Todos los métodos comparados fueron evaluados utilizando el mismo número de ensayos de adaptación del sujeto objetivo. La única excepción es SDT, que se incluyó como un método de referencia, que no realiza ningún tipo de adaptación.

La Figura 4.2 (a) muestra los resultados obtenidos utilizando CSP+LDA como modelo base. Puede observarse que XS-BOT supera a todos los competidores, alcanzando una tasa de aciertos promedio de sobre todos los sujetos de 0.820 ± 0.163 (media $\pm$ desviación estándar). El método base SDT obtiene una tasa de aciertos de 0.594 ± 0.105, mientras que los otros métodos adaptativos obtienen desempeños sustancialmente inferiores a XS-BOT: ADT (0.584 ± 0.107), CSP-GR (0.627 ± 0.118) y EA (0.617 ± 0.115). La prueba de Friedman [55] rechaza la hipótesis nula de tendencias centrales iguales entre los métodos ($p \ll 0{,}001$), y la posterior prueba post-hoc de Nemenyi [56] confirma que el rendimiento

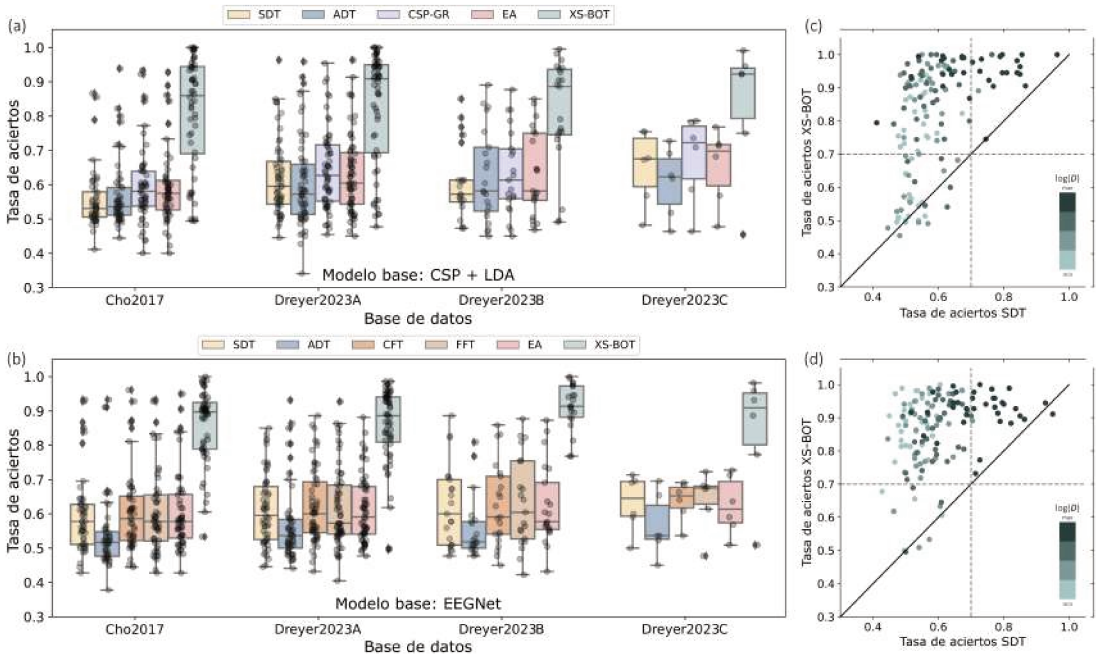

Figura 4.2: Resultados de diferentes métodos de adaptación en cuatro bases de datos de MI-EEG. Cada punto representa un sujeto de evaluación. Tasa de aciertos para SDT, ADT, CSP-GR, EA, FFT, CFT y XS-BOT en (a) CSP+LDA y (b) EEGNet; y para XS-BOT vs. SDT, en (c) CSP+LDA y (d) EEGNet. En (c) y (d), la línea diagonal indica igual rendimiento para ambos métodos y las líneas grises punteadas, el umbral de 0.7 de referencia. El color de cada punto codifica el logaritmo de la distintividad entre clases del sujeto.

de XS-BOT es significativamente mejor que SDT, ADT, EA y CSP-GDR, con $p \ll 0{,}001$ para todas las comparaciones de a pares correspondientes. Para evaluar si estos beneficios se extienden al usar modelos de DL, repetimos el análisis pero utilizando EEGNet (Figura 4.2 (b)). XS-BOT nuevamente muestra una mejora notable, con una tasa de aciertos promedio de 0.866 ± 0.109, que supera ampliamente al resultado sin adaptación (SDT, 0.610 ± 0.110) así como a todas las demás estrategias de adaptación: ADT (0.555 ± 0.105), AD-CFT (0.617 ± 0.112), AD-FFT (0.616 ± 0.119) y EA (0.612 ± 0.106). Estos resultados destacan a XS-BOT como un avance significativo en la decodificación de MI, con pruebas estadísticas que confirman la solidez de estos resultados ($p \ll 0{,}001$). Nótese que los métodos competidores, en ambos enfoques de ML, consistentemente se mantienen muy cercanos al desempeño del método sin adaptación, lo que refleja la elevada dificultad del escenario de transferencia entre sujetos y bases de datos. Este rendimiento resulta inferior al umbral de tasa de aciertos de 0.7 para un control efectivo de MI-BCI [45], lo que sugiere que darían lugar a sistemas sin utilidad práctica.

A pesar de la mejora consistente en la tasa de aciertos promedio a través de los sujetos, el rendimiento individual de XS-BOT varía sujeto a sujeto. Para comprender mejor estas diferencias, se investigó cómo las características del sujeto objetivo se relacionan con el rendimiento de la adaptación. En particular, se analizó si métricas como la distintividad de clase o la tasa de aciertos del modelo sin adaptación (SDT) tienen alguna relación con los resultados variables en la efectividad de XS-BOT. Las Figuras 4.2 (c) y (d) ilustran el impacto de la adaptación en el desempeño por sujeto. Cada punto representa un sujeto, con el eje x indicando la tasa de aciertos para SDT y el eje y la tasa de aciertos para XS-BOT. La línea diagonal sólida corresponde a un rendimiento igual entre ambos métodos, mientras que las líneas grises muestran el umbral de 0.7 de tasa de aciertos [45] utilizado como referencia. El color de cada punto refleja la distintividad de clase del sujeto, dividida en cuatro categorías basadas en los cuartiles de los valores log-transformados de distintividad de clase. Tonos más claros indican menor distintividad de clase (es decir, menor discriminabilidad entre clases de MI), mientras que tonos más oscuros representan mayor distintividad de clase. Esta codificación visual nos permite evaluar cómo la separabilidad de clases se relaciona con el rendimiento sin y con adaptación.

La Figura 4.2 (c) presenta los resultados para CSP+LDA. La mayoría de los sujetos están por encima de la diagonal, lo que indica ganancias consistentes gracias a la adaptación. Las pocas excepciones son sujetos con un rendimiento inicial SDT pobre y baja distintividad de clase, lo que sugiere que sus señales MI pueden carecer de suficiente información discriminativa entre clases para apoyar el aprendizaje de un plan de transporte

Entrenamiento en la Base de datos 3					
Evaluación	**SDT**	**ADT**	**CSP-GR**	**EA**	**XS-BOT**
Base de datos 2	0.627 ± 0.108	0.595 ± 0.113	0.646 ± 0.119	0.631 ± 0.114	**0.837** ± 0.140
Base de datos 4A	0.618 ± 0.114	0.594 ± 0.112	0.643 ± 0.117	0.632 ± 0.118	**0.830** ± 0.162
Base de datos 4B	0.595 ± 0.096	0.594 ± 0.122	0.634 ± 0.122	0.626 ± 0.126	**0.831** ± 0.151
Base de datos 4C	0.664 ± 0.107	0.602 ± 0.102	0.676 ± 0.122	0.646 ± 0.113	**0.834** ± 0.217
media ± std	0.620 ± 0.109	0.595 ± 0.112	0.644 ± 0.118	0.631 ± 0.116	**0.833** ± 0.153
Entrenamiento en la Base de datos 4A					
Base de datos 2	0.592 ± 0.095	0.595 ± 0.113	0.645 ± 0.116	0.627 ± 0.114	**0.862** ± 0.151
Base de datos 3	0.556 ± 0.076	0.567 ± 0.097	0.597 ± 0.113	0.593 ± 0.110	**0.774** ± 0.178
Base de datos 4B	0.568 ± 0.065	0.594 ± 0.122	0.631 ± 0.118	0.621 ± 0.125	**0.870** ± 0.144
Base de datos 4C	0.597 ± 0.075	0.602 ± 0.102	0.670 ± 0.120	0.660 ± 0.105	**0.830** ± 0.177
media ± std	0.574 ± 0.084	0.584 ± 0.108	0.625 ± 0.116	0.614 ± 0.114	**0.827** ± 0.166
Entrenamiento en la Base de datos 4B					
Evaluación	**SDT**	**ADT**	**CSP-GR**	**EA**	**XS-BOT**
Base de datos 2	0.554 ± 0.064	0.595 ± 0.113	0.649 ± 0.117	0.626 ± 0.118	**0.806** ± 0.159
Base de datos 3	0.500 ± 0.000	0.567 ± 0.097	0.606 ± 0.112	0.593 ± 0.104	**0.735** ± 0.179
Base de datos 4A	0.528 ± 0.049	0.594 ± 0.112	0.647 ± 0.117	0.632 ± 0.117	**0.791** ± 0.168
Base de datos 4C	0.527 ± 0.045	0.602 ± 0.102	0.677 ± 0.118	0.645 ± 0.109	**0.845** ± 0.052
media ± std	0.528 ± 0.051	0.587 ± 0.107	0.636 ± 0.116	0.619 ± 0.113	**0.781** ± 0.168
Entrenamiento en la Base de datos 4C					
Evaluación	**SDT**	**ADT**	**CSP-GR**	**EA**	**XS-BOT**
Base de datos 2	0.581 ± 0.065	0.595 ± 0.113	0.652 ± 0.116	0.624 ± 0.111	**0.807** ± 0.168
Base de datos 3	0.504 ± 0.013	0.567 ± 0.097	0.603 ± 0.113	0.599 ± 0.109	**0.771** ± 0.177
Base de datos 4A	0.570 ± 0.080	0.594 ± 0.112	0.650 ± 0.117	0.629 ± 0.116	**0.811** ± 0.161
Base de datos 4B	0.576 ± 0.084	0.594 ± 0.122	0.644 ± 0.121	0.627 ± 0.126	**0.837** ± 0.124
media ± std	0.555 ± 0.071	0.587 ± 0.109	0.637 ± 0.117	0.619 ± 0.113	**0.802** ± 0.164

Tabla 4.1: Resultados al entrenar en diferentes bases de datos (modelo base: CSP+LDA).

efectivo. Para SDT, solo los sujetos MI con una distintividad de clase muy alta (puntos más oscuros) tienden a lograr buen rendimiento, lo que sugiere que en BCIs entre sujetos sólo los usuarios de MI con una discriminabilidad muy alta pueden usar de manera confiable el sistema sin adaptación de datos. Es importante señalar que sujetos con menor distintividad se benefician más de la adaptación con XS-BOT, permitiendo que muchos de ellos superen el umbral de 0.7. La Figura 4.2 (d) presenta el análisis análogo para EEGNet. Si bien las tendencias generales se mantienen consistentes con las del panel (c), las mejoras introducidas por XS-BOT son aún más pronunciadas en este contexto. Aquí, no sólo un mayor número de sujetos supera el umbral de 0.7, sino que además las ganancias sobre SDT son mayores, creando una clara brecha entre XS-BOT y el resultado sin adaptación.

Como complemento al análisis principal, se realizó un experimento adicional en el que se empleó cada base de datos disponible como conjunto de entrenamiento, una por vez, utilizando para evaluación las demás. Los resultados de este análisis se presentan en la Tabla 4.1 para el caso de CSP+LDA y en la Tabla 4.2 para EEGNet, mostrando en ambos casos una tendencia coherente con la observada en el experimento principal y confirmando la robustez del método frente a diferentes datos de entrenamiento.

Entrenamiento en la Base de datos 3						
Evaluación	**SDT**	**ADT**	**CFT**	**FFT**	**EA**	**XS-BOT**
Base de datos 2	0.613 ± 0.117	0.554 ± 0.109	0.622 ± 0.119	0.632 ± 0.124	0.608 ± 0.122	**0.790** ± 0.157
Base de datos 4A	0.588 ± 0.089	0.571 ± 0.109	0.599 ± 0.099	0.607 ± 0.105	0.596 ± 0.092	**0.768** ± 0.153
Base de datos 4B	0.595 ± 0.106	0.561 ± 0.089	0.617 ± 0.125	0.621 ± 0.132	0.607 ± 0.108	**0.753** ± 0.186
Base de datos 4C	0.593 ± 0.084	0.558 ± 0.094	0.599 ± 0.080	0.613 ± 0.105	0.601 ± 0.074	**0.803** ± 0.131
media ± std	0.599 ± 0.103	0.562 ± 0.105	0.610 ± 0.110	0.619 ± 0.116	0.602 ± 0.105	**0.775** ± 0.158
Entrenamiento en la Base de datos 4A						
Evaluación	**SDT**	**ADT**	**CFT**	**FFT**	**EA**	**XS-BOT**
Base de datos 3	0.585 ± 0.102	0.540 ± 0.106	0.590 ± 0.111	0.599 ± 0.116	0.595 ± 0.106	**0.760** ± 0.176
Base de datos 4B	0.615 ± 0.111	0.561 ± 0.089	0.630 ± 0.123	0.632 ± 0.130	0.619 ± 0.107	**0.783** ± 0.180
Base de datos 4C	0.605 ± 0.090	0.558 ± 0.094	0.627 ± 0.095	0.646 ± 0.098	0.619 ± 0.096	**0.788** ± 0.161
Base de datos 2	0.624 ± 0.115	0.554 ± 0.109	0.621 ± 0.118	0.631 ± 0.121	0.625 ± 0.116	**0.786** ± 0.177
media ± std	0.606 ± 0.109	0.550 ± 0.104	0.611 ± 0.115	0.619 ± 0.119	0.612 ± 0.109	**0.775** ± 0.175
Entrenamiento en la Base de datos 4B						
Evaluación	**SDT**	**ADT**	**CFT**	**FFT**	**EA**	**XS-BOT**
Base de datos 2	0.602 ± 0.102	0.554 ± 0.109	0.605 ± 0.111	0.620 ± 0.114	0.617 ± 0.104	**0.779** ± 0.166
Base de datos 3	0.577 ± 0.095	0.540 ± 0.106	0.576 ± 0.108	0.584 ± 0.115	0.585 ± 0.096	**0.805** ± 0.143
Base de datos 4A	0.586 ± 0.088	0.571 ± 0.109	0.588 ± 0.092	0.601 ± 0.102	0.606 ± 0.099	**0.809** ± 0.152
Base de datos 4C	0.602 ± 0.096	0.558 ± 0.094	0.598 ± 0.093	0.587 ± 0.096	0.607 ± 0.098	**0.839** ± 0.152
media ± std	0.589 ± 0.095	0.556 ± 0.107	0.590 ± 0.103	0.601 ± 0.110	0.603 ± 0.100	**0.799** ± 0.153
Entrenamiento en la Base de datos 4C						
Evaluación	**SDT**	**ADT**	**CFT**	**FFT**	**EA**	**XS-BOT**
Base de datos 2	0.599 ± 0.092	0.554 ± 0.109	0.599 ± 0.099	0.609 ± 0.110	0.587 ± 0.091	**0.824** ± 0.122
Base de datos 3	0.559 ± 0.090	0.540 ± 0.106	0.557 ± 0.098	0.570 ± 0.110	0.559 ± 0.076	**0.778** ± 0.160
Base de datos 4A	0.598 ± 0.090	0.571 ± 0.109	0.601 ± 0.103	0.606 ± 0.107	0.589 ± 0.093	**0.845** ± 0.129
Base de datos 4B	0.584 ± 0.094	0.561 ± 0.089	0.586 ± 0.103	0.605 ± 0.122	0.585 ± 0.095	**0.831** ± 0.137
media ± std	0.586 ± 0.092	0.556 ± 0.106	0.586 ± 0.101	0.597 ± 0.111	0.580 ± 0.088	**0.819** ± 0.139

Tabla 4.2: Resultados al entrenar en diferentes bases de datos (modelo base: EEGNet).

Para evaluar si las mejoras de rendimiento de XS-BOT se extienden a diferentes arquitecturas de aprendizaje profundo, se realizaron experimentos adicionales empleando otros dos modelos DL ampliamente utilizados para la decodificación de MI: ShallowConvNet y DeepConvNet [41]. Ambos modelos fueron entrenados y evaluados siguiendo el mismo esquema experimental descrito para EEGNet.

Los resultados se presentan en la Figura 4.3. El panel (a) muestra el rendimiento de ShallowConvNet, mientras que el panel (b) presenta los resultados para DeepConvNet. En el caso de ShallowConvNet, XS-BOT alcanza una tasa de aciertos media de 0.830 ± 0.115, superando ampliamente el resultado sin adaptación (SDT) de 0.602 ± 0.103. XS-BOT también supera de forma significativa a todos los métodos de adaptación competidores: FFT (0.606 ± 0.110), CFT (0.605 ± 0.107), ADT (0.575 ± 0.107) y EA (0.563 ± 0.069). La prueba de Friedman rechaza la hipótesis nula de igualdad de rendimiento entre métodos ($p \ll 0{,}001$), y las pruebas post-hoc de Nemenyi confirman que XS-BOT presenta un rendimiento significativamente superior al de SDT y a todos los demás métodos adaptativos ($p \ll 0{,}001$ en todas las comparaciones por pares).

En el caso de DeepConvNet, XS-BOT alcanza una tasa de aciertos promedio de 0.854 ± 0.123, superando nuevamente de forma notable al método de referencia SDT, para el

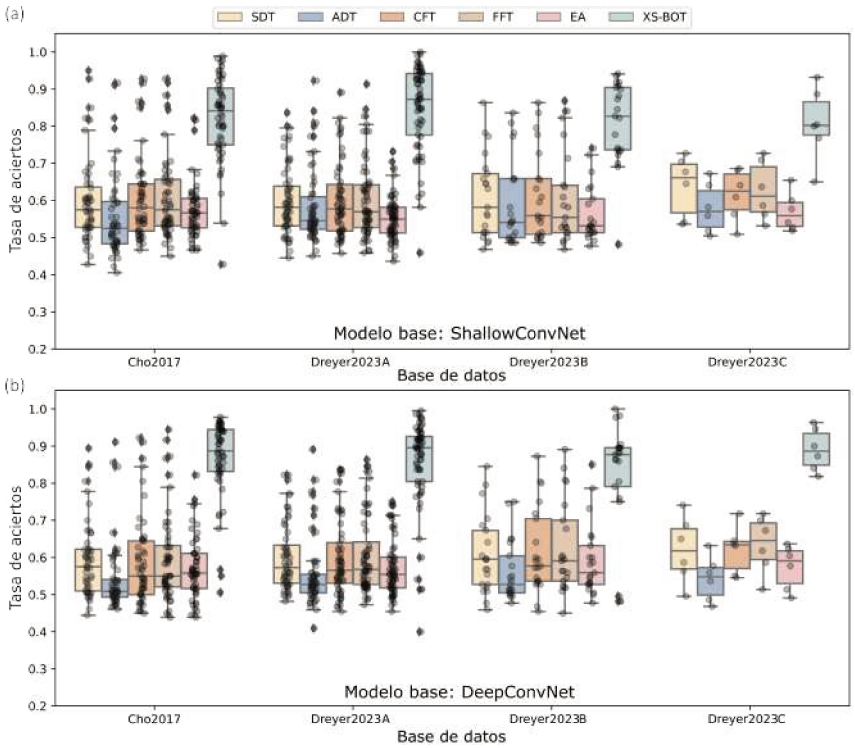

Figura 4.3: Comparación de XS-BOT frente a los métodos con CTI en diferentes bases de datos de MI-EEG. Cada punto representa un sujeto de prueba. (a) Resultados para Shallow ConvNet. (b) Resultados para Deep ConvNet.

cual la tasa de aciertos promedio resulta 0.600 ± 0.097. Asimismo, una vez más XS-BOT supera a todos los demás métodos de adaptación por una amplia diferencia: FFT (0.606 ± 0.112), CFT (0.597 ± 0.109), EA (0.576 ± 0.082) y ADT (0.549 ± 0.089). Las pruebas estadísticas correspondientes confirman la significancia de estas diferencias ($p \ll 0{,}001$).

Finalmente, para entender en mayor profundidad a qué se debe el rendimiento de XS-BOT y la influencia del uso de la etiqueta indicada, lo comparamos con los mismos métodos competidores que antes, pero ahora incluyendo la etiqueta indicada del ensayo actual durante el entrenamiento o ajuste fino del modelo, es decir, variantes con inclusión del ensayo actual (CTI). Aunque tales enfoques podrían no ser viables para escenarios en línea debido a su alto tiempo de cómputo, pueden proporcionar cotas superiores para el rendimiento útiles.

En particular, para el entrenamiento con datos de adaptación con inclusión del ensayo actual (ADT-CTI), se entrena un nuevo modelo de decodificación en cada ensayo de prueba combinando los datos de adaptación del sujeto con los datos del ensayo actual. En este escenario, tanto el extractor de características como el clasificador incorporan información de la tarea proveniente de la etiqueta actual durante la optimización del modelo. Para el modelo de DL, CTI también se aplicó en las estrategias de ajuste fino. Como antes, en FFT-CTI se ajusta completamente el modelo, mientras que CFT-CTI restringe el ajuste fino solo a las capas del clasificador. Es oportuno señalar que estos enfoques CTI tienen una ventaja inherente sobre XS-BOT: ajustan directamente los parámetros del clasificador utilizando la etiqueta indicada del ensayo actual. Sin embargo, XS-BOT mantiene el clasificador fijo y usa la etiqueta indicada solo para guiar la transformación de las características de forma que los grupos se muevan juntos.

También consideramos una variante (CSP-CTI), en la que el extractor de características (CSP) se entrena usando tanto los datos de adaptación como el ensayo actual, mientras que el clasificador (LDA) se entrena sólo con los datos de adaptación. En otras palabras, mientras CSP se ajusta para cada nuevo ensayo, la frontera de clasificación LDA se aprende sin acceso al ensayo actual. CSP-CTI permite una comparación más alineada con XS-BOT, donde el ensayo actual influye en cómo se transforman las características CSP pero no en el clasificador.

La Figura 4.4 (a) muestra los resultados para CSP+LDA. XS-BOT supera de manera consistente a CSP-CTI, lo que sugiere que usar la información del ensayo actual para guiar el proceso de adaptación, como lo hace XS-BOT, es más efectivo que incluir el ensayo actual en el entrenamiento del extractor de características. Además, aunque ADT-CTI reentrena el modelo completo para cada ensayo, XS-BOT logra un mejor rendimiento.

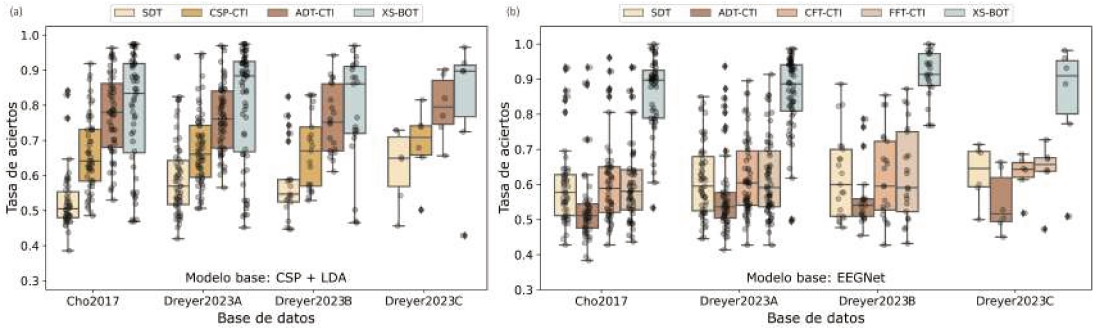

Figura 4.4: Comparación de XS-BOT contra los métodos de referencia CTI en diferentes bases de datos de MI-EEG. Cada punto representa un sujeto de prueba. (a) Resultados para CSP+LDA. (b) Resultados para EEGNet.

Figure 4.4 (b) muestra que para EEGNet, a diferencia de CSP+LDA, ADT-CTI obtiene un rendimiento peor que SDT. Esto podría explicarse por el hecho de que el tamaño del conjunto de adaptación combinado con el ensayo actual no es lo suficientemente grande como para entrenar un modelo de DL desde cero. Los enfoques de ajuste fino (CFT-CTI y FFT-CTI) tampoco logran alcanzar a SDT. Es notable que XS-BOT supera significativamente a todos estos enfoques CTI, destacando su capacidad para explotar eficazmente la información del ensayo actual sin necesidad de reentrenar el modelo. El enfoque propuesto no sólo tienen un mejor desempeño de clasificación, sino que también ofrece viabilidad práctica en un contexto de BCI en tiempo real. Para EEGNet, las estrategias CTI requieren entrenar o ajustar partes del modelo para cada ensayo, lo cual es inviable en la realidad debido al tiempo computacional asociado. En contraste, XS-BOT evita estas limitaciones de tiempo, lo que lo hace muy adecuado para MI-BCIs a lazo cerrado, donde se debe brindar retroalimentación inmediatamente después de que el usuario realizó la tarea de MI en cada ensayo. Cabe destacar que el tiempo promedio de adaptación por ensayo fue inferior a 40 ms tanto para los modelos CSP+LDA como EEGNet utilizando una computadora de escritorio estándar (Intel i7, 16 GB de RAM).

Capítulo 5

Conclusiones

Uno de los grandes desafíos en la aplicación práctica de las MI-BCIs en contextos de rehabilitación y asistencia motora es la necesidad de largas etapas de calibración personalizadas para cada usuario. A lo largo de esta tesis se desarrollaron y validaron estrategias para mitigar esta problemática, a través de métodos de aprendizaje automático y procesamiento de señales que permiten reducir significativamente los tiempos y requerimientos de calibración.

En primer lugar, se propuso PySimMIBCI, un método de simulación de señales de MI-EEG que permite generar datos artificiales realistas e interpretables, con información específica del usuario. Estos datos no sólo resultan útiles durante el desarrollo y la validación de modelos de decodificación, sino que también se pueden integrar como herramienta de aumentación en esquemas de aprendizaje profundo, enriqueciendo el entrenamiento sin necesidad de adquirir nuevos datos reales.

En segundo lugar, se diseñó XS-BOT, una estrategia de adaptación entre sujetos y entre bases de datos que, mediante transporte óptimo inverso, permite la decodificación de MI de una manera efectiva en usuarios no vistos durante el entrenamiento. XS-BOT evita el reentrenamiento del modelo, manteniendo un costo computacional mínimo y requiere únicamente una pequeña cantidad de datos de calibración del usuario objetivo y un número reducido de canales. Esta solución representa un avance concreto hacia MI-BCIs más prácticas y usables, especialmente en escenarios clínicos donde se desea reducir el tiempo de calibración para así maximizar el tiempo de control de la BCI, que es cuando efectivamente ocurre la rehabilitación.

Complementariamente, se colaboró en un proyecto orientado al diseño de MI-BCIs co-adaptativas centrados en el usuario, donde se empleó transporte óptimo inverso para la adaptación entre sesiones y la evaluación en tiempo real de la capacidad de modulación

de MI. Como resultado de este trabajo, se generó la publicación incluida en el Anexo C. Además, se adquirió experiencia en evaluaciones de MI-BCIs en tiempo real, tanto en el desarrollo de *software* específico para la adquisición y el procesamiento en línea de EEG como en la experimentación con sujetos humanos, habiento colaborado en la adquisición de datos en pruebas pilotos.

En conjunto, los desarrollos aquí presentados contribuyen al objetivo de reducir las barreras prácticas de las MI-BCIs, acercando su uso real en entornos de rehabilitación. Si bien aún quedan desafíos abiertos, como su validación con pacientes, los resultados obtenidos sientan una base sólida para el avance hacia tecnologías de asistencia más eficientes, simples de usar y centradas en la persona.

Bibliografía

[1] Leon Poltawski et al. "Assessing the impact of upper limb disability following stroke: a qualitative enquiry using internet-based personal accounts of stroke survivors". En: *Disability and rehabilitation* 38.10 (2016), págs. 945-951.

[2] Jonathan Wolpaw y Winter Wolpaw. "Brain-computer interfaces: something new under the sun". En: *Brain-computer interfaces: principles and practice* 14 (2012).

[3] CaMIlle Jeunet, EMIlie Jahanpour y Fabien Lotte. "Why standard brain-computer interface (BCI) training protocols should be changed: an experimental study". En: *Journal of neural engineering* 13.3 (2016), pág. 036024.

[4] Ander Ramos-Murguialday et al. "Brain-machine interface in chronic stroke: randomized trial long-term follow-up". En: *Neurorehabilitation and neural repair* 33.3 (2019), págs. 188-198.

[5] Fabien Lotte, Laurent Bougrain y Maureen Clerc. "Electroencephalography (EEG)-based brain-computer interfaces". En: *Wiley encyclopedia of electrical and electronics engineering* (2015), pág. 44.

[6] Fabien Lotte et al. "A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update". En: *Journal of neural engineering* 15.3 (2018), pág. 031005.

[7] Wei Wu, Srikantan Nagarajan y Zhe Chen. "Bayesian Machine Learning: EEG/MEG signal processing measurements". En: *IEEE Signal Processing Magazine* 33.1 (2015), págs. 14-36.

[8] Jussi T Lindgren et al. "simBCI—a framework for studying BCI methods by simulated EEG". En: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26.11 (2018), págs. 2096-2105.

[9] Elham Barzegaran et al. "EEGSourceSim: A framework for realistic simulation of EEG scalp data using MRI-based forward models and biologically plausible signals and noise". En: *Journal of Neuroscience Methods* 328 (2019), pág. 108377.

[10] Victoria Peterson et al. "Transfer learning based on optimal transport for motor imagery brain-computer interfaces". En: *IEEE Transactions on Biomedical Engineering* 69.2 (2021), págs. 807-817.

[11] Yannick Roy et al. "Deep learning-based electroencephalography analysis: a systematic review". En: *Journal of Neural Engineering* 16.5 (2019), pág. 051001.

[12] Andrea Biasiucci et al. "Brain-actuated functional electrical stimulation elicits lasting arm motor recovery after stroke". En: *Nature communications* 9.1 (2018), pág. 2421.

[13] Alexander A Frolov et al. "Post-stroke rehabilitation training with a motor-imagery-based brain-computer interface (BCI)-controlled hand exoskeleton: a randomized controlled multicenter trial". En: *Frontiers in neuroscience* 11 (2017), pág. 400.

[14] Chaofei Fan et al. "Plug-and-play stability for intracortical brain-computer interfaces: a one-year demonstration of seamless brain-to-text communication". En: *Advances in neural information processing systems* 36 (2023), págs. 42258-42270.

[15] Joshua Giles et al. "A transfer learning algorithm to reduce brain-computer interface calibration time for long-term users". En: *Frontiers in Neuroergonomics* 3 (2022), pág. 837307.

[16] Fabien Lotte. "Generating artificial EEG signals to reduce BCI calibration time". En: *5th International Brain-Computer Interface Workshop*. 2011, págs. 176-179.

[17] Satyam Kumar et al. "Transfer learning promotes acquisition of individual BCI skills". En: *PNAS nexus* 3.2 (2024), pgae076.

[18] Sung-Jin Kim, Dae-Hyeok Lee y Yeon-Woo Choi. "CropCat: Data Augmentation for Smoothing the Feature Distribution of EEG Signals". En: *2023 11th International Winter Conference on Brain-Computer Interface (BCI)*. IEEE. 2023, págs. 1-4.

[19] Yu Pei et al. "Data augmentation: using channel-level recombination to improve classification performance for motor imagery EEG". En: *Frontiers in Human Neuroscience* 15 (2021), pág. 645952.

[20] Mostafa Neo Mohsenvand, Mohammad Rasool Izadi y Pattie Maes. "Contrastive representation learning for electroencephalogram classification". En: *Machine Learning for Health*. PMLR. 2020, págs. 238-253.

[21] Fang Wang et al. "Data augmentation for EEG-based emotion recognition with deep convolutional neural networks". En: *International Conference on Multimedia Modeling*. Springer. 2018, págs. 82-93.

[22] Aaqib Saeed et al. "Learning from heterogeneous eeg signals with differentiable channel reordering". En: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, págs. 1255-1259.

[23] Yuanzhe Dong et al. "An Approach for EEG Data Augmentation Based on Deep Convolutional Generative Adversarial Network". En: *2022 IEEE International Conference on Cyborg and Bionic Systems (CBS)*. IEEE. 2023, págs. 347-351.

[24] Fatemeh Fahimi et al. "Generative adversarial networks-based data augmentation for brain-computer interface". En: *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[25] Sujit Roy et al. "MIEEG-GAN: generating artificial motor imagery electroencephalography signals". En: *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2020, págs. 1-8.

[26] Qiqi Zhang y Ying Liu. "Improving brain computer interface performance by data augmentation with conditional deep convolutional generative adversarial networks". En: *arXiv preprint arXiv:1806.07108* (2018).

[27] Yonghao Song et al. "Common Spatial Generative Adversarial Networks based EEG Data Augmentation for Cross-Subject Brain-Computer Interface". En: *arXiv preprint arXiv:2102.04456* (2021).

[28] Qingguo Wei y Xinjie Ding. "Intra-and inter-subject common spatial pattern for reducing calibration effort in MI-based BCI". En: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 31 (2023), págs. 904-916.

[29] Siamac Fazli et al. "Subject-independent mental state classification in single trials". En: *Neural networks* 22.9 (2009), págs. 1305-1312.

[30] Hyohyeong Kang, Yunjun Nam y Seungjin Choi. "Composite common spatial pattern for subject-to-subject transfer". En: *IEEE Signal Processing Letters* 16.8 (2009), págs. 683-686.

[31] Haiping Lu et al. "Regularized common spatial pattern with aggregation for EEG classification in small-sample setting". En: *IEEE transactions on Biomedical Engineering* 57.12 (2010), págs. 2936-2946.

[32] Fabien Lotte y Cuntai Guan. "Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms". En: *IEEE Transactions on biomedical Engineering* 58.2 (2010), págs. 355-362.

[33] Wojciech Samek et al. "Robust spatial filtering with beta divergence". En: *Advances in Neural Information Processing Systems* 26 (2013).

[34] Daniel Bartz y Klaus-Robert Müller. "Covariance shrinkage for autocorrelated data". En: *Advances in neural information processing systems* 27 (2014).

[35] Benjamin Blankertz et al. "Invariant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing". En: *Advances in neural information processing systems* 20 (2007).

[36] Aarthy Nagarajan et al. "Transferring a deep learning model from healthy subjects to stroke patients in a motor imagery brain–computer interface". En: *Journal of Neural Engineering* 21.1 (2024), pág. 016007.

[37] Kaishuo Zhang et al. "Adaptive transfer learning for EEG motor imagery classification with deep convolutional neural network". En: *Neural Networks* 136 (2021), págs. 1-10.

[38] Fangzhou Xu et al. "A transfer learning framework based on motor imagery rehabilitation for stroke". En: *Scientific Reports* 11.1 (2021), pág. 19783.

[39] He He y Dongrui Wu. "Transfer learning for brain–computer interfaces: A Euclidean space data alignment approach". En: *IEEE Transactions on Biomedical Engineering* 67.2 (2019), págs. 399-410.

[40] Bruna Junqueira et al. "A systematic evaluation of euclidean alignment with deep learning for EEG decoding". En: *Journal of Neural Engineering* 21.3 (2024), pág. 036038.

[41] Robin Tibor Schirrmeister et al. "Deep learning with convolutional neural networks for EEG decoding and visualization". En: *Human brain mapping* 38.11 (2017), págs. 5391-5420.

[42] Vernon Lawhern et al. "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces". En: *Journal of neural engineering* 15.5 (2018), pág. 056013.

[43] Benjamin Blankertz et al. "Optimizing spatial filters for robust EEG single-trial analysis". En: *IEEE Signal processing magazine* 25.1 (2007), págs. 41-56.

[44] Michael Tangermann et al. "Review of the BCI competition IV". En: *Frontiers in Neuroscience* 6 (2012), pág. 55.

[45] Min-Ho Lee et al. "EEG dataset and OpenBMI toolbox for three BCI paradigms: An investigation into BCI illiteracy". En: *GigaScience* 8.5 (2019), giz002.

[46] Hohyun Cho et al. "EEG datasets for motor imagery brain–computer interface". En: *GigaScience* 6.7 (2017), gix034.

[47] Pauline Dreyer et al. "A large EEG database with users' profile information for motor imagery brain-computer interface research". En: *Scientific Data* 10.1 (2023), pág. 580.

[48] Fabien Lotte et al. "A review of classification algorithms for EEG-based brain-computer interfaces". En: *Journal of Neural Engineering* 4.2 (2007), R1.

[49] Florian Yger, Maxime Berar y Fabien Lotte. "Riemannian approaches in brain-computer interfaces: a review". En: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25.10 (2016), págs. 1753-1762.

[50] KA Colwell et al. "Channel selection methods for the P300 Speller". En: *Journal of neuroscience methods* 232 (2014), págs. 6-15.

[51] Andrew R Webb. *Statistical pattern recognition*. John Wiley & Sons, 2003.

[52] Ravikiran Mane et al. "FBCNet: A multi-view convolutional neural network for brain-computer interface". En: *arXiv preprint arXiv:2104.01233* (2021).

[53] Fabien Lotte y Camille Jeunet. "Defining and quantifying users' mental imagery-based BCI skills: a first step". En: *Journal of neural engineering* 15.4 (2018), pág. 046030.

[54] Janez Demšar. "Statistical comparisons of classifiers over multiple data sets". En: *The Journal of Machine learning research* 7 (2006), págs. 1-30.

[55] Milton Friedman. "The use of ranks to avoid the assumption of normality implicit in the analysis of variance". En: *Journal of the American Statistical Association* 32.200 (1937), págs. 675-701.

[56] Peter Bjorn Nemenyi. *Distribution-free multiple comparisons*. Princeton University, 1963.

[57] David Sabbagh et al. "Predictive regression modeling with MEG/EEG: from source power to signals and cognitive states". En: *NeuroImage* 222 (2020), pág. 116893.

[58] Lucas C Parra et al. "Recipes for the linear analysis of EEG". En: *Neuroimage* 28.2 (2005), págs. 326-341.

[59] Sylvain Baillet, John C Mosher y Richard M Leahy. "Electromagnetic brain mapping". En: *IEEE Signal processing magazine* 18.6 (2001), págs. 14-30.

[60] Christoph M Michel y Denis Brunet. "EEG source imaging: a practical review of the analysis steps". En: *Frontiers in Neurology* 10 (2019), pág. 325.

[61] Gert Pfurtscheller y FH Lopes Da Silva. "Event-related EEG/MEG synchronization and desynchronization: basic principles". En: *Clinical Neurophysiology* 110.11 (1999), págs. 1842-1857.

[62] Gert Pfurtscheller y Christa Neuper. "Motor imagery activates primary sensorimotor area in humans". En: *Neuroscience letters* 239.2-3 (1997), págs. 65-68.

[63] Yongwoong Jeon et al. "Event-related (De) synchronization (ERD/ERS) during motor imagery tasks: Implications for brain–computer interfaces". En: *International Journal of Industrial Ergonomics* 41.5 (2011), págs. 428-436.

[64] Gert Pfurtscheller et al. "Mu rhythm (de) synchronization and EEG single-trial classification of different motor imagery tasks". En: *NeuroImage* 31.1 (2006), págs. 153-159.

[65] Sophocles J Orfanidis. *Introduction to signal processing*. Prentice-Hall, Inc., 1995.

[66] Christophe Destrieux et al. "Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature". En: *Neuroimage* 53.1 (2010), págs. 1-15.

[67] Maliha Hossain. "Whitening and coloring transformations for multivariate Gaussian data". En: *A lecture partly based on the ECE662 Spring* (2014).

[68] Anna Rita Bentivoglio et al. "Analysis of blink rate patterns in normal subjects". En: *Movement disorders* 12.6 (1997), págs. 1028-1034.

[69] Yvonne Tran et al. "The influence of mental fatigue on brain activity: Evidence from a systematic review with meta-analyses". En: *Psychophysiology* 57.5 (2020), e13554.

[70] Teng Cao et al. "Objective evaluation of fatigue by EEG spectral analysis in steady-state visual evoked potential-based brain-computer interfaces". En: *Biomedical Engineering online* 13.1 (2014), págs. 1-13.

[71] Thomas Jacquet et al. "Mental fatigue induced by prolonged motor imagery increases perception of effort and the activity of motor areas". En: *Neuropsychologia* 150 (2021), pág. 107701.

[72] Saroj KL Lal y Ashley Craig. "Driver fatigue: electroencephalography and psychological assessment". En: *Psychophysiology* 39.3 (2002), págs. 313-321.

[73] JO Phipps-Nelson, Jennifer R Redman y Shantha MW Rajaratnam. "Temporal profile of prolonged, night-time driving performance: breaks from driving temporarily reduce time-on-task fatigue but not sleepiness". En: *Journal of Sleep Research* 20.3 (2011), págs. 404-415.

[74] Leonard J Trejo et al. "EEG-based estimation and classification of mental fatigue". En: *Psychology* 6.05 (2015), pág. 572.

[75] Upasana Talukdar y Shyamanta M Hazarika. "Estimation of mental fatigue during EEG based motor imagery". En: *Intelligent Human Computer Interaction: 8th International Conference, IHCI 2016, Pilani, India, December 12-13, 2016, Proceedings 8*. Springer. 2017, págs. 122-132.

[76] Thomas Donoghue et al. "Parameterizing neural power spectra into periodic and aperiodic components". En: *Nature Neuroscience* 23.12 (2020), págs. 1655-1665.

[77] Donald B Percival y Andrew T Walden. *Spectral analysis for physical applications*. cambridge University press, 1993.

[78] Fabien Lotte. "Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain–computer interfaces". En: *Proceedings of the IEEE* 103.6 (2015), págs. 871-890.

[79] Nik Khadijah Nik Aznan et al. "Simulating brain signals: Creating synthetic EEG data via neural-based generative models for improved SSVEP classification". En: *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2019, págs. 1-8.

[80] Sinno Jialin Pan y Qiang Yang. "A Survey on Transfer Learning". En: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), págs. 1345-1359. DOI: 10.1109/TKDE.2009.191.

[81] Victoria Peterson et al. "Towards subject-centered co-adaptive brain-computer interfaces based on backward optimal transport". En: *Journal of Neural Engineering* (2025).

[82] Leonid V Kantorovich. "On the translocation of masses". En: *Dokl. Akad. Nauk. USSR (NS)*. Vol. 37. 1942, págs. 199-201.

[83] Nicolas Courty et al. "Optimal transport for domain adaptation". En: *IEEE transactions on pattern analysis and machine intelligence* 39.9 (2016), págs. 1853-1865.

[84] Pierre Guetschel y Michael Tangermann. "Transfer Learning between Motor Imagery Datasets using Deep Learning–Validation of Framework and Comparison of Datasets". En: *arXiv preprint arXiv:2311.16109* (2023).

Anexos

Contribuciones

Galván, C. M.; Spies, R. D.; Milone, D. H.; & Peterson, V. (2024). Neurophysiologically meaningful motor imagery EEG simulation with applications to data augmentation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 32, 2346-2355.

En este trabajo se propone PySimMIBCI, un método de simulación de datos que permite generar señales artificiales de MI-EEG realistas y con características específicas del usuario, incorporando información neurofisiológica en modelos directos basados en MRI, obteniéndose datos bien definidos para el desarrollo de modelos de decodificación y para el uso en la aumentación de datos en modelos de deep learning aplicados a MI-BCIs.

Las contribuciones de la tesista en este trabajo fueron:

- Diseño e implementación del método de simulación PySimMIBCI y la estrategia de aumentación de datos basada en señales simuladas.
- Desarrollo de experimentos para evaluar la calidad de los datos simulados y la estrategia de aumentación propuesta.
- Visualización y discusión de resultados con coautores.
- Redacción y corrección del manuscrito.
- Creación y estructuración del repositorio público del proyecto.

Galván, C. M.; Milone, D. H.; Spies, R. D. & Peterson, V. (2025). No need to re-train for new users: cross-subject domain adaptation in rehabilitative BCIs via backward optimal transport. *En revisión.*

En este trabajo se propone XS-BOT, una estrategia de adaptación de dominio entre sujetos y entre bases de datos que permite una decodificación efectiva en MI-BCIs

con aplicación en rehabilitación. Este enfoque evita la necesidad de reentrenar modelos, requiriendo pocos datos del nuevo usuario para la adaptación y sólo tres canales de EEG.

Las contribuciones de la tesista en este trabajo fueron:

- Extensión y adaptación del método BOTDA para el escenario de adaptación entre sujetos y entre base de datos.
- Implementación y comparación con múltiples estrategias de adaptación.
- Visualización y discusión de resultados con coautores.
- Redacción y corrección del manuscrito.
- Desarrollo del código y organización del repositorio público.

Peterson, V.; Spagnolo, V.; Galván, C. M.; Nieto, N.; Spies, R., & Milone, D. H. (2025). Towards subject-centered co-adaptive brain-computer interfaces based on backward optimal transport. *Journal of Neural Engineering.*

En este trabajo se propone un método para BCIs co-adaptativas centradas en el usuario que, mediante transporte óptimo inverso, mitiga cambios de dominio entre sesiones y permite evaluar en tiempo real la capacidad de modulación de MI, resultando útil para brindar retroalimentación que ayude al control de la BCI.

Las contribuciones de la tesista en este trabajo fueron:

- Generación de los datos artificiales utilizados para las pruebas de hipótesis.
- Participación activa en la discusión de los experimentos y la interpretación de los resultados.
- Revisión crítica del manuscrito y aportes a su mejora.

Anexo A

Neurophysiologically meaningful motor imagery EEG simulation with applications to data augmentation

Neurophysiologically meaningful motor imagery EEG simulation with applications to data augmentation

C. M. Galván, R. D. Spies, D. H. Milone and V. Peterson

C. M. Galván, V. Peterson and R. D. Spies are at Instituto de Matemática Aplicada del Litoral, IMAL, UNL, CONICET, Santa Fe, Argentina
(e-mail: cgalvan@santafe-conicet.gov.ar, vpeterson@santafe-conicet.gov.ar, rspies@santafe-conicet.gov.ar)
D. H. Milone is at Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional, sinc(i), FICH-UNL, CONICET, Santa Fe, Argentina
(email: dmilone@sinc.unl.edu.ar)

Abstract

Motor imagery-based Brain-Computer Interfaces (MI-BCIs) have gained a lot of attention due to their potential usability in neurorehabilitation and neuroprosthetics. However, the accurate recognition of MI patterns in electroencephalography signals (EEG) is hindered by several data-related limitations, which restrict the practical utilization of these systems. Moreover, leveraging deep learning (DL) models for MI decoding is challenged by the difficulty of accessing user-specific MI-EEG data on large scales. Simulated MI-EEG signals can be useful to address these issues, providing well-defined data for the validation of decoding models and serving as a data augmentation approach to improve the training of DL models. While substantial efforts have been dedicated to implementing effective data augmentation strategies and model-based EEG signal generation, the simulation of neurophysiologically plausible EEG-like signals has not yet been exploited in the context of data augmentation. Furthermore, none of the existing approaches have integrated user-specific neurophysiological information during the data generation process. Here, we present PySimMIBCI, a framework for generating realistic MI-EEG signals by integrating neurophysiologically meaningful activity into biophysical forward models. By means of PySimMIBCI, different user capabilities to control an MI-BCI can be simulated and fatigue effects can be included in the generated EEG. Results show that our simulated data closely resemble real data. Moreover, a proposed data augmentation strategy based on our simulated user-specific data significantly outperforms other state-of-the-art augmentation approaches, enhancing DL models performance by up to 15%.

Keywords

Brain-Computer Interfaces, Surface EEG, Motor Imagery, Deep Learning, Data Augmentation, EEG simulation.

1 Introduction

Decoding methods for the development of Brain-Computer Interfaces (BCIs) are commonly based on non-invasive Electroencephalography (EEG) recordings. Despite the significant progress observed in the last decade, these systems still face several data-related challenges which restrict their use in real-world scenarios [1]. One of the main challenges arises from the inability to determine the actual mental activities of users throughout the experiment based on EEG signals. This is even more evident for endogenous BCIs, which rely on users' intentions regardless of the external stimuli presented. Consequently, the artificial generation of EEG-like signals plays a crucial role in providing well-defined data for the development, validation, and interpretation of decoding models.

Although real experiments and real data will ultimately always be needed for testing and validating experimental hypotheses, collecting large-scale data under specific experimental conditions is expensive, time-consuming, and quite often not possible. Despite the existence of publicly available EEG-BCI databases, they remain scarce, hindering the development of algorithmic solutions for these systems. Motor imagery (MI) BCIs for motor rehabilitation are particularly affected by these data limitations since the MI is an endogenous process hard to master by the BCI user [2]. In addition, the desirable MI-BCIs-based rehabilitation scenario should minimize the time spent in system calibration and maximize the feedback delivery to enhance the rehabilitative impact [3]. Data scarcity in the field of MI-BCIs has become even more critical with the advent of deep learning (DL) models, whose performance strongly depends on the volume of available data for training [4].

To address these challenges, there have been mainly two lines of work: model-based EEG signal generation and data augmentation techniques. Multiple alternatives have been proposed within each of these lines. Regarding data augmentation, one of the simplest approaches consists in generating new EEG trials by combining randomly selected segments of the few available trials [5, 6]. Similarly, channel-level recombination is presented as a data augmentation method in [7]. Another strategy consists of the generation of new training samples by the addition of Gaussian noise to each sample of the original training data, as suggested in [8, 9]. In the same line, in [8, 10] a set of augmentations based on transformations that do not affect the meaningful information of EEG data were used. In the last few years, methods using generative adversarial networks (GANs) have gained much attention among artificial MI-EEG data generation techniques [11–15]. For instance, in [11], a GAN model was trained to generate artificial EEG samples that exhibited notable similarity to real data in the time, frequency and time-frequency domains. A cross-subject EEG classification framework using a GAN-based model for augmentation was proposed in [15]. In particular, in order to preserve the spatial features of the EEG signals and enhance the discrimination between different classes, a specific module was introduced in the discriminator. Similarly, in [16], customized neural-based generative models were used to generate steady-state visual evoked potential EEG signals for data augmentation purposes. Nevertheless, existing data augmentation methods, whether employing data transformations or neural networks, can generate data that disregard the

key neurophysiological information from the EEG, creating signals that might not be observed in real-world scenarios [17].

From a different perspective, multiple works have been dedicated to model-based EEG signal generation through the development of neurophysiologically plausible models. These models aim to capture the underlying mechanisms and dynamics of brain activity, allowing for the creation of synthetic data that closely mimic the patterns observed in real-world recordings. In this line, in [18] the authors generated simulated EEG data by combining biologically plausible activity with user-specific head models based on magnetic resonance imaging (MRI). Similarly, the SimBCI approach [19] stood out by providing a pipeline to simulate EEG data following particular experimental BCI protocols. These advancements in generating meaningful EEG-like data have significantly contributed to the exploration and validation of novel decoding algorithms in the field of EEG analysis and BCIs. However, though relevant in the field, the level of attention, fatigue and user MI brain modulation ability have not yet been included as possible inputs in the state-of-the-art models.

Despite the great efforts invested in implementing effective data augmentation strategies and model-based MI-EEG signal generation, it is noteworthy that neurophysiologically plausible EEG-like signals have not yet been exploited in the context of data augmentation for DL. Additionally, existing neurophysiologically plausible models lack the capability to incorporate user-specific information, which could be valuable when considering data simulation as a strategy for data augmentation.

In this work, improvements are simultaneously proposed in these two specific areas: model-based EEG signal generation and data augmentation for MI-BCI decoding. Specifically, a model-based simulation framework for the generation of realistic MI-EEG-like data built by embedding user-specific neurophysiologically meaningful activity into MRI-based forward models is presented. The main objectives are two, namely: i) account for reliable and well-defined data for building EEG decoding models, and ii) augment training data for DL in a task- and user-specific manner. Our approach, which we named PySimMIBCI, extends the method presented in [19] and can be used to create artificial brain recordings that are electrophysiologically similar to real data for a given BCI user. The proposed framework allows the simulation of different subject modulation capabilities and the introduction of different sources of EEG variability and artifacts, such as fatigue and eye-movement. When user-specific information is included, it becomes an effective tool for data augmentation in the context of DL models for MI-BCIs. For this purpose, a way to construct training batches that keeps a fixed number of real and simulated trials while guaranteeing class balance is proposed. The presented framework is open source and freely available (https://github.com/catalinamagalvan/PySimMIBCI), extending previously existing implementations in the widely accepted MNE-Python Library [20].

2 A framework for realistic user-specific MI-EEG data generation

2.1 Generative forward model for BCI-EEG simulations

There are two main typical approaches in the field of EEG modeling: biophysical source modeling and statistical modeling [21]. The former proposes that the EEG signals are

obtained by linear combinations of unobservable physiological sources, such as a parcel of cerebral cortex with synchronously firing neurons [22]. In contrast, statistical modeling views sources as statistical entities, in principle unconnected to anatomical concepts [23]. PySimMIBCI is based on the biophysical source modeling approach [21], motivated by the goal to generate artificial signals that are neurologically plausible. Formally, let $X \in R^{c \times t}$ be the measured EEG observations over time, where c denotes the number of EEG channels and t is the number of samples; let also $G \in R^{c \times s}$ be the leadfield or gain matrix modeling the effects of tissue conduction from neuronal sources to the scalp, with s being the number of sources, and $Z \in R^{s \times t}$ comprise the matrix of source activities in the cortex [24], including both signal and noise components. Finally, let $N \in R^{c \times t}$ be a matrix of surface noise added at the electrode level. The linear superposition model leads to

$$X = GZ + N. \tag{1}$$

This biophysical forward model linearly links the current dipoles located on the vertices of the cortical meshes to the EEG electrodes located on the scalp. In a simulation framework based on a biophysical source model, the generated EEG X is obtained by linear mixing unobserved brain sources Z through the leadfield G. Such a leadfield may vary in complexity from a single-sphere model to a physiologically realistic, user-specific head model [19, 21]. To get G, we employ an MRI-based model that includes physiological information about head and skull morphology, taking into account the conductivity properties of the tissues [25]. Section 2.2 describes how Z can be constructed for a left vs. right hand MI scenario. For simplicity, N is not considered in this work.

The framework described above shows how EEG data can be modeled from a biophysical perspective. However, when considering BCIs, it is essential to incorporate specific temporal and spatial information associated with the particular BCI protocol. Therefore, to simulate informative EEG data from the decoding perspective of BCIs, our framework, PySimMIBCI, integrates this information into the biophysical generative model. Each signal component is characterized by three attributes: events, waveform and spatial information, which are equivalent to the *when*, *what* and *where* properties, proposed in [19]. The interpretation of these characteristics is very intuitive: the events information (*when*) defines the specific onsets and durations on a timeline; the waveform information (*what*) specifies the signal content introduced in each event (e.g. synchronizations/desynchronizations in specific EEG bands or particularly evoked responses); and the spatial information (*where*) indicates the cortical location of the source activity. Note that this approach assumes independence among signal components, allowing to analyze the effects of varying these parameters during the generation process.

2.2 Simulating left vs. right hand MI-EEG data

Different neural signals can be used to design and build a BCI. In particular, BCIs based on the MI paradigm have a clear importance in the context of hand motor rehabilitation since it has been shown that they can provide additional benefits to conventional physiotherapy [26, 27]. As in [19], here we simulate right hand MI vs. left hand MI data. Our artificial MI-EEG-like signals are obtained as a superposition of task-dependent rhythmic activity generated on the motor cortex and task-unrelated activity, which includes background noise and eye-related artifacts. The temporal and spatial properties of each component in the artificial signals are described below.

Task-related activity: It is well-known that hand MI causes different event-related synchronization (ERS) and desynchronization (ERD) patterns in multiple bands of the EEG in the sensorimotor cortex [28]. Briefly, an ERD corresponds to a decrease in power, whereas an ERS represents an increase in power relative to a baseline. The most relevant frequency bands involved in MI-EEG modulation have been found to be the α (8–14 Hz) and β (14–30 Hz) bands [29]. The proposed simulation considers the typical ERD in the α band in the contralateral hand motor area [30, 31]. Thus, the generators of this rhythm are assumed to be dipoles in the right and left hand motor areas. For the precise localization of sulco-gyral structures in the cortex the *aparc.a2009s* parcellation [32] available in the FreeSurfer[1] package is employed. For the spatial information, circular patches of 30 mm radius, centered on the centroids of the right and left prefrontal area parcels are configured. Data is generated as standard Gaussian-Distributed Signals [33] filtered in order to restrict the frequency content to a narrow subband, with a specific central frequency and bandwidth within the α band. Rhythmic activity in this subband is present in both the left and right hand motor areas throughout the experiment, with a reduction in amplitude (ERD) on the contralateral side during the hand MI tasks. To simulate the acquisition protocol, an artificial timeline with randomly ordered trials of each class is generated.

Task-unrelated activity: One type of noise and two types of artifacts are considered in our simulation framework: background noise, blink artifact and eye-movement artifact, as in [34]. Background noise comprises aperiodic neural activity, that is, spectrally pink components related to neural background processes [18]. It is modeled by $1/f^{\lambda}$. This component is consistently present throughout the cortex surface for all frequencies f. In order to generate it, temporally and spatially uncorrelated white Gaussian noise is temporally filtered to have a specific power spectral density (PSD) according to the noise exponent λ. Subsequently, by means of a coloring transformation [35], the resulting components are forced to have a predetermined covariance matrix. Through this spatial filtering, a covariance structure determined by the proximity of sources, that is with closer sources exhibiting a stronger correlation, is obtained. Furthermore, we simulate the effects of eye-related artifacts, specifically blink and eye-movement artifacts. For the blink artifact, the *mne.simulation.add_eog* function of MNE-Python [20] is used. In this implementation, random activation onsets are drawn from an inhomogeneous Poisson process with blink rate oscillating between 4.5 and 17 blinks/minute according to the low and high blink rates [36]. The activation kernel consists of a 250 ms Hanning window. Two activated dipoles are located in the $z = 0$ plane at $\pm 30°$ away from the midline (nasion). Eye-movement artifacts are modeled as a Poisson process. In each activation, the eye-movement is generated as random signal with maximum random duration of 500 ms. The two generator dipoles are located as in the blink artifact implementation [20].

Fig. 1 summarizes PySimMIBCI for the task of right hand MI. Firstly, as shown in the left side of the figure, source activity is simulated by combining specific spatial, waveform and events' information. In this example, spatial information comprises right and left hand motor areas and task-related waveform is the characteristic ERD in the α band in the contralateral hand motor area. Events' information involves a typical protocol stimulation timeline for MI: 4 s of MI followed by an inter-trial break of variable duration. All this information is mixed at the source space and gives origin to the source

[1] http://surfer.nmr.mgh.harvard.edu/

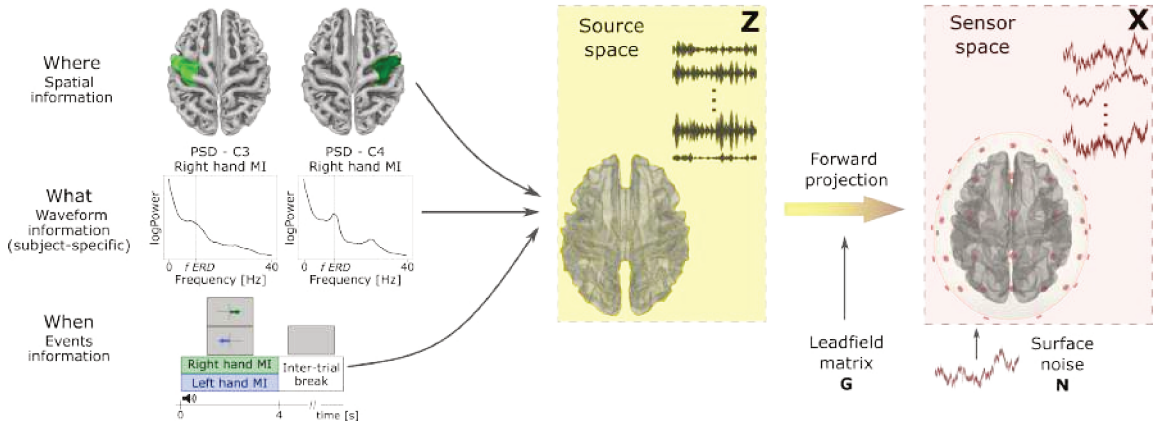

Figure 1: PySimMIBCI pipeline exemplified by the simulation of right hand MI-EEG data. Source space data is generated by the combination of specific spatial, waveform, and events' information related to the BCI task. A forward operator is then used to get sensor space data and at this level different types of surface noise and artifacts are added.

activity Z. Through a forward projection, characterized by a leadfield matrix G, this information can be accessed at the sensor level, where different types of noises can be added via matrix N resulting in the simulated EEG signal X.

2.3 Simulation of users with different brain modulation capabilities and inclusion of fatigue effects

When developing EEG decoding methods, it is often useful to generate data corresponding to different groups of users with diverse levels of proficiency in controlling an MI-BCI. This enables researchers to test hypotheses and gain insights from the obtained results. To this end, two strategies for generating artificial users with different MI modulation characteristics are proposed here. The first strategy involves manipulating the percentage of α ERD in the contralateral hand motor area described in Section 2.2. Higher percentages of ERD indicate stronger modulation and thus would correspond to more proficient users, while less pronounced ERDs would be associated with users with reduced modulation capability. The second strategy simulates scenarios where the user does not perform the required task (also known as null signals in the Evoked Potentials literature [37]). This is done by including a certain proportion of failed MI trials, that is, trials without ERD in the corresponding area. Artificial users with different levels of competence to control an MI-BCI can be then simulated by separately or jointly combining these ideas, as further shown in Section 3.

Another important issue is the fact that prolonged engagement in a cognitively demanding task, such as MI, can lead to mental fatigue, degrading mental activities performance [38]. In other words, attentional focus becomes less efficient and more error-prone as the time spent on the task increases. These consequences of mental fatigue are accompanied by changes in the brain activity, particularly in low-frequency bands, resulting in altered EEG signals [39]. In order to generate realistic MI-EEG signals that reflect fatigue-related effects, we propose a way to include them in the simulated data. Several studies of EEG oscillations have found that both θ (4-8 Hz) and α (8-13 Hz) band powers increase with mental fatigue [40–42]. More specifically, in [43, 44] mental fatigue has been associated with progressive increases in both frontal θ and parietal α powers with task

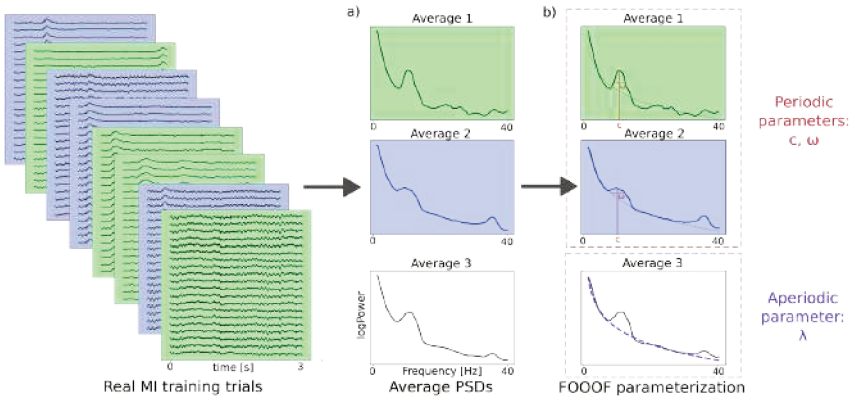

Figure 2: Extraction of user-specific periodic and aperiodic information. a) First, PSD is calculated for each channel of each training trial. b) After that, periodic and aperiodic parameterization is done over different sets of average PSDs using FOOOF [45].

duration. Thus, these neural markers serve as a way of simulating mental fatigue. In our simulation framework, θ and α brain generators are assumed to be dipoles in the frontal and parietal areas, respectively. As before, we generate this component by a standard Gaussian-Distributed Signal filtered to get the θ and α band activities. Fatigue can be added from any specific onset time and its level is linearly increased with time, as shown in Section 3.

2.4 User-specific MI-EEG generation for neurophysiologically meaningful data augmentation

Data augmentation techniques aim to enlarge training datasets by artificially increasing its size and diversity. A traditional approach involves creating additional training samples through label-preserving transformations, such as the geometric transformations used in computer vision [46]. In the context of MI-EEG decoding, data consists of a collection of noisy, non-stationary time series from different electrodes and, thus, straightforward geometric transformations are not suitable since they may distort temporal and spatial-domain features. Additionally, while GAN models have been employed to generate MI-EEG data [13] and can generate data through user-specific training, the personalized neuromodulation information associated with MI can potentially be degraded or even lost during the data generation process. This becomes particularly relevant in MI decoding since the models are typically user-specific, i.e. they are trained and tested on data from the same person. To address these limitations, our proposed data simulation framework is conceived for generating user-specific MI-EEG signals for data augmentation.

The first step to generate user-specific data consists in calculating the PSD of real MI-EEG training trials for each channel by the multitaper method [47]. As shown in Fig. 2 a), three different average PSDs of interest are obtained: Average 1), the average over all right hand MI trials for channel C4; Average 2), the average over all left hand MI trials for channel C3, and Average 3), the average over all training trials and over all channels. Next, the fitting oscillations and one-over f (FOOOF) [45] method is employed to get specific parameters from the different average PSDs, as seen in Fig. 2 b). The FOOOF method is a physiologically-informed tool to parameterize neural PSDs in two constituents: aperiodic and periodic components. The aperiodic activity is exponentially

decreasing with frequency and it can be described by

$$L(f) = b - \log(f^{\lambda}), \tag{2}$$

where λ is the exponent which characterizes the pattern of aperiodic power across frequencies, b is the offset that specifies the uniform shift in power and f is the frequency [45]. In contrast, periodic components of neural data consist of overlying putative periodic oscillatory peaks, which in FOOOF are individually modeled by a Gaussian-like function

$$P(f) = a \exp\left(\frac{-(f-c)^2}{2\omega^2}\right), \tag{3}$$

where a is the power, c is the central frequency and ω is the width of the peak.

It is reasonable to expect that aperiodic activity will be similarly distributed throughout the cortex and not associated with any specific MI tasks. Hence, for the aperiodic component parameterization, we employed the average PSD computed over all training trials and across all EEG channels (Average 3). On the contrary, taking into account that the two main EEG channels on the hand motor area are C3 and C4 [31], only the information in these channels is used for fitting the periodic components. That is, the average PSDs to extract periodic components are calculated over all training trials belonging to each class for both channels (Average 1 and Average 2).

After the extraction of these parameters from real MI-EEG, user-specific data is simulated as described below. The aperiodic exponent λ is employed to generate the user-specific background activity. Further, the α ERD is generated based on the parameters of central frequency and bandwidth of the highest peak in the 7-14 Hz band. If the user has no peak detected in the α band, a generic peak with central frequency $c = 11.5$ Hz and bandwidth $\omega = 3$ Hz is considered, according to the most likely α peak across all real users (Fig. A1 in Supplementary Material).

3 Data and experimental setup

In order to evaluate the quality of our simulations, an analysis comparing the artificial MI-EEG data with their real-world counterparts was conducted. Further, to show the impact of our data augmentation approach in DL, we employed real MI-EEG data to train the model and compare its performance before and after the inclusion of simulated data. In both cases, two publicly available datasets were used in this work:

- Dataset-1 [34] comprises two sessions of left vs. right hand MI-EEG data from two users (b and g). For each user, the first session contains 100 trials per class, while the number of trials for the second session varies among users. Data was recorded at 1000 Hz using 59 electrodes densely distributed over the sensorimotor areas. Since the simulation process requires knowing the exact electrodes' position, only the 41 channels from to the 10-5 international electrode system were used. To train and test DL models data were downsampled to 250 Hz following [48].

- Dataset-2 [49] is a 54-users two-sessions left vs. right hand MI-EEG dataset. Each session comprises 100 trials per class. EEG was recorded at 1000 Hz using 62 electrodes positioned according to the international 10-20 system. For running DL models, 20 channels in the motor region were chosen, as in [49], and, here again following [48], data were downsampled to 250Hz.

Table 1: Simulation parameters for artificial users with different brain modulation capabilities.

User name	ERD [%]	Failed trials [%]
ideal	50	0
S40	40	0
S30	30	0
S20	20	0
S10	10	0
SF10	50	10
SF20	50	20
SF30	50	30
SF40	50	40

As DL model for MI decoding we used the Filter Bank Convolutional Network (FBCNet) proposed in [48], a hybrid model designed for user-specific MI-BCI classification. The main reason for selecting this state-of-the-art model is because it has demonstrated superior performance in several datasets, and its implementation in PyTorch [50] is open and readily accessible. FBCNet comprises four stages: i) spectral filtering of the EEG in multiple narrow bands; ii) extraction of spatial discriminative patterns for each view by a depthwise convolutional layer; iii) compact representation of the temporal information by a variance layer; and iv) classification by a fully connected layer. In all the experiments, the architecture was trained using Adam optimizer with default settings and the log-cross-entropy loss, following the original work [48]. The model was trained using early stopping with a patience of 200 epochs. Once the stopping criterion was reached, the network parameters with the best validation accuracy were restored. The maximum number of training epochs was limited to 1500. The architecture and hyperparameters settings are described in detail in Section 7 of the Supplementary Material.

3.1 PySimMIBCI for realistic MI-EEG data simulation

As described in Section 2.3, users with different modulation characteristics were simulated, and subsequently compared with real MI-EEG data from Dataset-1. User g from this dataset was employed as the guide BCI-user since it had obtained the best classification performance in the original study [34]. Neurophysiological information was extracted from this user following the steps described in Section 2.4. For each simulated user, 200 artificial trials were generated (100 per class). First, ERD percentage was varied from 50% (ideal user [34]) to 10% (S10) in steps of 10%. Then, we simulated users that do not perform the required task by including an increasing proportion of trials without ERD, from 10% (user SF10) to 40% (user SF40). In the latter cases, the ERD percentage was always set to 50%. A summary of the simulation parameters can be found in Table 1.

Next, to illustrate the effects of mental fatigue, we introduced the associated parietal α band and frontal θ band activities, as detailed in Section 2.3. As before, user g from Dataset-1 was the guide BCI-user and 200 trials were generated. Considering that mental fatigue is most likely to manifest towards the end of a long session, the first half of the trials excluded fatigue effects, while the second half included a progressively increasing level of fatigue, simulated by a linearly increasing amplitude with time in both bands.

3.2 PySimMIBCI as a plausible data augmentation strategy

To leverage PySimMIBCI for data augmentation purposes, 100 trials per class were generated for each real user with the ideal parameters (50% of ERD and 0% bad trials) and the corresponding periodic and aperiodic extracted information, as described in Section 2.4. To see the corresponding ablation study, refer to Section 5 of the Supplementary Material.

The proposed data augmentation strategy was compared with state-of-the-art approaches, which were selected to cover the main strategies presented in the literature at which method implementation was clear or publicly available. We briefly describe them below:

- Segmentation and recombination (SR): training trials (2 s) were splitted into 8 segments and new trials were created by concatenating random analogous segments from trials of the same class, as in [51].
- Gaussian noise addition (GNA): copies of each training trial were generated by adding Gaussian noise with zero mean and standard deviation equal to 0.2 [9].
- Time Masking (TM): augmented trials were derived from the original training trials by zeroing out all channels of randomly chosen 100-sample time segments [8].
- Channels Dropout (CD): new training trials were obtained by zeroing out randomly chosen channels from the original training trials. The probability of dropping each channel was 0.2, as in [10].
- Bandstop Filtering (BF): new trials were generated by applying a 5 Hz bandwidth bandstop filter, with random central frequency, between 0 and 40 Hz [8].
- Wasserstein GAN (WGAN): The model in [16] was adapted for generating MI-EEG data. Training was conducted on a user-specific basis.

In all the cases, the size of the obtained augmented dataset was twice the size of the original real MI-EEG dataset.

Baseline and augmented models share the same hyperparameter setting but differ in the training data composition. For all the tested data augmentation strategies, training batches are constructed with a fixed number of real and simulated trials, always guaranteeing class balance. More precisely, each batch comprises N_b trials and has the following structure: N_a augmented trials for each class, randomly sampled from the total of simulated trials, and $N_r = (N_b - 2N_a)/2$ randomly sampled real trials of each class. This sampling approach introduces valuable variability, which has a regularizing effect on model calibration. Simultaneously, it conserves class balance within each batch and fixes real/simulated trials ratio across batches. With these considerations, we ensure that the model benefits from diverse training examples while preserving a consistent data composition across different batches (refer to Fig. A2 in Supplementary Material to see the effect of different sampling methods for batch construction). In this study we used $N_a = 1$ since no differences were found when adding more simulated trials into each training batch (see Fig. A3 in Supplementary Material).

To evaluate the impact of the proposed data augmentation strategy we used the relative change (RC) in accuracy over a baseline performance, defined as follows:

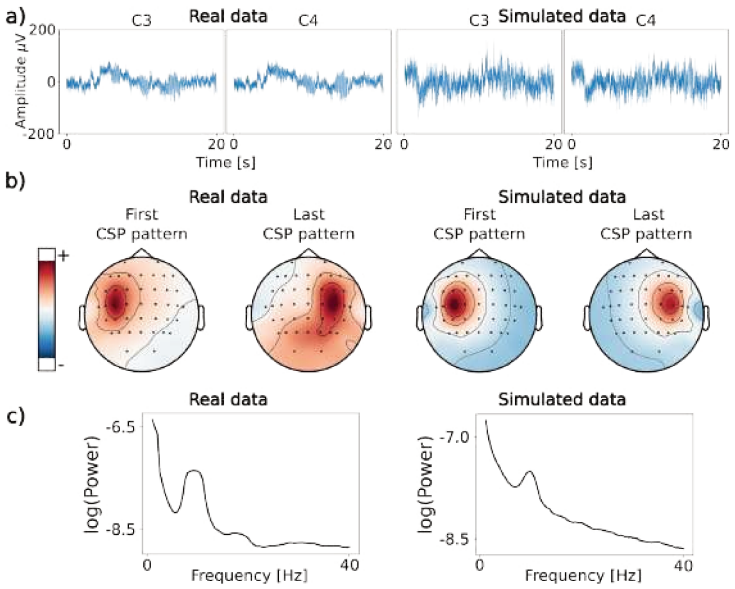

Figure 3: Temporal, spatial and spectral plots for real and simulated EEG data. a) 20 s of both signals in channels C3 and C4. b) First and last CSP patterns. c) Average PSDs.

$$RC = \frac{a_a - a_r}{a_r}, \tag{4}$$

where a_a refers to test accuracy with data augmentation, that is, ratio of number of correct predictions over the total number of predictions, and a_r is the reference test accuracy (without data augmentation). If $RC > 0$, data augmentation improves model performance while if $RC < 0$, it deteriorates it.

4 Results

4.1 PySimMIBCI for realistic MI-EEG data simulation

To assess the similarity between artificially generated signals and real MI-EEG data, a qualitative comparison based on temporal, spectral and spatial information was first conducted. The simulated data corresponded to the 200 trials from the ideal user (Table 1), while the real data comprised 200 trials from session 1 of user g from Dataset-1.

The visual comparison of real and artificial data in the time domain was made at channels C3 and C4. Fig. 3 (a) shows 20 seconds of both simulated and real signals. The spatial assessment was made by examining the scalp topographies of the first and last CSP patterns (Fig. 3 b)), which represent time-invariant EEG spatial source distributions. These patterns provide insights into the underlying cortical activity of each MI task [52, 53]. Finally, to analyze the spectral contents of both signals, the average PSD across the 200 trials was plotted for C3 and C4 channels (Fig. 3 c)). As it can be observed, spatial, temporal and spectral traces of the simulated MI-EEG very well resemble their real counterparts.

To quantify the similarity between simulated and real data, we computed the cross-correlation between the corresponding channels of different pairs of signals: 1) simulated vs. real signal (simulated-real), 2) two real signals, i.e. user b vs. user g from Dataset-1 (real-real), and 3) randomly generated vs. real signal (random-real). It turns out that

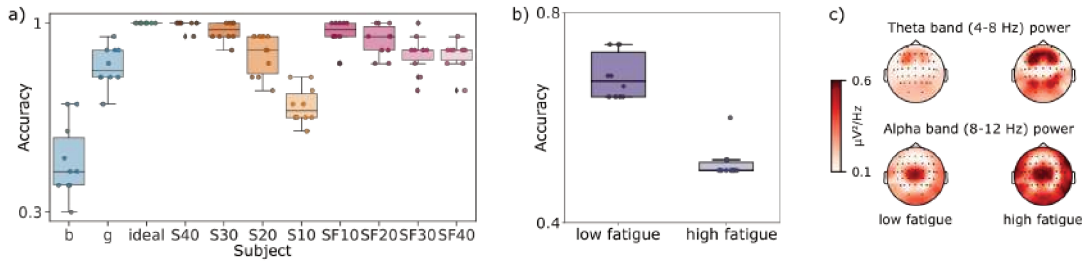

Figure 4: Results for simulated data corresponding to different brain modulation capabilities and fatigue effects simulation. a) 10-fold CV results for real and simulated data. Each point corresponds to test accuracy for one fold. Light-blue points represent the results obtained for two real MI-EEG users, and green points the results for the ideal simulated user (50% of ERD and 0% of failed trials). Simulated users with decreasing percentages of ERD (50-10%) are in an orange palette while a pink palette was used for simulated users with increasing proportions of failed trials (0-40%). b) Accuracy values obtained for two test partitions, one with low and the other with high mental fatigue. The model was trained on data without fatigue effects. c) Topographic maps in the θ and α bands for the two test partitions with fatigue effects.

the differences in the maximum cross-correlation values for the pairs simulated-real and real-real are negligible, and in some cases, the similarity simulated-real exceeds that of the real-real pair. These results can be found in Fig. A4 in the Supplementary Material.

The classification performance of simulated MI-EEG data was also evaluated and compared with real data accuracy in a 10 fold cross-validation (CV) scenario. Eight folds were used for training, one fold was utilized for validation and the remaining fold for testing. Only session 1 (200 trials) of real data was part of this analysis. Regarding simulated data, the 200 trials for each one of the users of Table 1 were considered.

Fig. 4 a) shows the results of this CV analysis for users b and g of Dataset-1 and for the nine simulated users described in Table 1. Users with decreasing percentages of ERD are represented in an orange palette while a pink palette is used for simulated users with increasing percentages of trials without ERD. As can be seen, simulated data can yield similar classification performance as real MI-EEG data. Further, as expected, lower percentages of ERD and higher proportions of bad trials lead to decreased classification performance.

To show how the effects of fatigue may influence decoding performance, signals with simulated fatigue can be used as training and testing data. In our analysis, the first half of the session described in Section 3.1, where no fatigue effects were present, was used as training data. Data from the second half of the session were divided into two test partitions: one with low and one with high fatigue effects. Fig. 4 b) shows the accuracy obtained for the two test sessions. A notable decrease in accuracy is observed when mental fatigue effects are simulated over a prolonged session. Further, Fig. 4 c) illustrates the topographic maps in the θ and α bands. As expected, these plots show that, as simulated mental fatigue level increases throughout the session, powers in θ and α bands in the frontal and parietal areas increase as well.

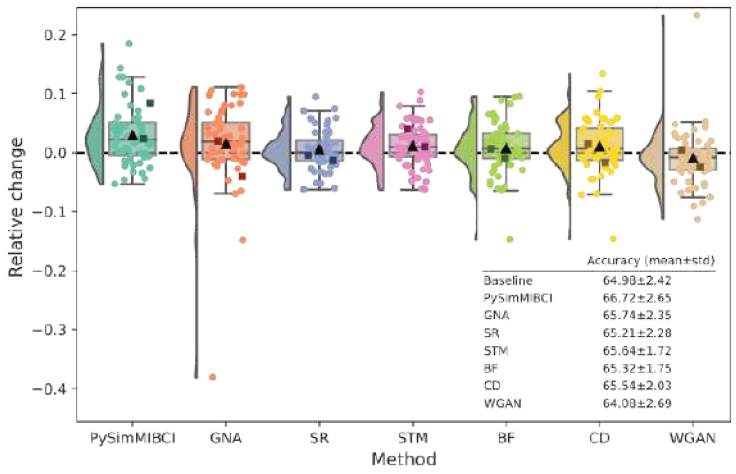

Figure 5: Data augmentation results in two real MI-EEG datasets. Each point corresponds to the mean RC values across 10 random seeds for one real user (dark squares: Dataset-1, bright circles: Dataset-2). Average RC value across all the users is shown by a black triangle.

4.2 PySimMIBCI as a plausible data augmentation strategy

In this section, we show the impact of using model-based user-specific simulated data as a data augmentation strategy for MI-BCI decoding based on DL. The results correspond to a within-subject cross-session analysis. In this analysis, for each subject, the data from session 1 was used for model training while session 2 data was used for model testing. The training data was further divided into a training and a validation set (25 trials per class). To account for variations in model initialization, this process was repeated 10 times using different random seeds. Fig. 5 shows the RC yielded by each tested data augmentation method in the within-subject cross-session scenario. The mean RC values across 10 random seeds are plotted for a total of 56 users (dark squares: Dataset-1, light circles: Dataset-2). PySimMIBCI achieves the highest average accuracy value (66.72% ± 2.65%), followed by GNA (65.74% ± 2.35%), STM (65.64% ± 1.71%), CD (65.54% ± 2.03%), BF (65.32% ± 1.75%), SR (65.21% ± 2.28%) and WGAN (64.08% ± 2.69%). The baseline accuracy (i.e. without data augmentation) was 64.98% ± 2.42%. Remarkably, PySimMIBCI is the only approach that does not result in a performance decrease of more than 5% for any of the users.

The statistical analysis supports the effectiveness of the proposed method. The Friedman test [54] rejected the null hypothesis (p-value < 0.05) that there is no difference in the central tendency of the methods. Additionally, the post-hoc Nemenyi test [55] revealed that PySimMIBCI, GNA and STM methods are significantly better (p-value < 0.05) than the baseline. It also showed that PySimMIBCI is significantly better than SR, WGAN, BF, CD and STM, but similar to GNA. An evaluation of PySimMIBCI across four additional EEG decoding models with varying degrees of complexity and diverse input types can be found in Supplementary Material, Section 6.

5 Discussions and conclusions

In this work, we introduced PySimMIBCI, a novel approach to generate realistic and user-specific MI-EEG-like signals by embedding neurophysiologically meaningful activity into

MRI-based forward models. Different user capabilities to generate distinguishable brain MI patterns were simulated and for the first time the effect of fatigue in EEG signals was considered in artificial data generation. By leveraging these user-specific simulated data for data augmentation we were able to significantly improve the training of DL models with application in MI-BCI decoding.

A complete framework to generate realistic MI-EEG data for right vs. left hand MI scenarios was presented in the first part of this work. Our results convincingly demonstrated the electrophysiological similarity between our simulated data and real MI-EEG data. The temporal traces of the simulated signals exhibited comparable amplitudes and visual appearance to actual EEG data (Fig. 3 a)). Moreover, when analyzing the CSP patterns, we observed a clear correspondence in the spatial properties of the simulated and real data (Fig. 3 b)). These patterns further reinforced the neurophysiological plausibility of the simulations, as it can be verified by the amplitudes in the hand motor areas where ERDs take place in both real and simulated MI-EEG data. Additionally, the spectral characteristics of the simulated signals closely resemble those of real signals, with both exhibiting a typical $1/f^{\lambda}$ shaped spectrum and a noticeable peak in the α band (Fig. 3 c). It is worth clarifying that the α peak is not always as prominent as shown in Fig. 3 c in real MI-EEG data. It should be mentioned also that despite our simulations simply use the typical and well-described ERD in the α band of the contralateral hand motor area, effective results were achieved. In fact, by inspecting the spectrum of real EEG data, a clear task-related α peak was visualized in most of the cases (Fig. A1, Supplementary Material), exhibiting a variety of central frequencies and bandwidths and allowing us to parameterize it. On the contrary, that was not the case for the β band, as only 7 out of the 56 users exhibited a characterizable peak within this frequency range. Naturally, the inclusion of more complex patterns and their impact in the MI-EEG data generation process is undoubtedly an interesting topic to explore in the future.

It is timely to mention here that in our simulations two sets of user-specific parameters are considered: a) the central frequency and bandwidth in the α peak, and b) the exponent of the $1/f^{\lambda}$ aperiodic activity. This approach is in line with recent findings that emphasize the significance of non-oscillatory activity as a predictor of individual motor abilities, complementing the well-studied sensorimotor characteristics [56]. By estimating the periodic parameters we can identify user-specific variation in the task-related ERD activity. Additionally, the user-specific fitting of the aperiodic activity enables to preserve cognitive function capability and retain valuable information about individual visuomotor skills [56].

In the development of an EEG-BCI decoding model, it is essential to assess its performance with varying degrees of data distinguishability and account for typical neurophysiological changes that occur between BCI sessions. To address these issues, users with different modulation capabilities and levels of attention were modeled by employing a range of ERD percentages and introducing different proportions of failed trials, respectively. Through our experiments, we demonstrated that these two strategies effectively simulate users that achieve different classification performances. However, as it becomes evident from the results shown in Fig. 4 a), the two scenarios have different effects from the classification perspective. When the percentage of ERD decreases, the classes become almost completely overlapped, dropping the accuracy to near by-chance levels. In contrast, adding trials without ERD produces data from a new different class, which are randomly assigned to any MI class, causing, as expected, a less pronounced loss in model performance.

When trying to control a BCI, other mental states and cognitive processes might affect the ability to maintain voluntary control of the EEG activity. In particular, during long MI-BCI sessions, the required high concentration and mental effort, often lead to mental fatigue, which results in a decline in system performance [57]. This impact of mental fatigue on EEG patterns has also been observed in patients undergoing MI-BCI stroke rehabilitation, who are even more susceptible to mental fatigue than users without neurological pathologies [58]. In this work, we simulated these effects on the EEG by progressively increasing power in frontal θ and parietal α, as task duration increases. As expected, the inclusion of fatigue resulted in a noticeable decrease in classification performance (Fig. 4 b)), which was consistently accompanied by increases in frontal θ and parietal α powers, as shown in the topography maps in the bands of interest (Fig. 4 c). Although we simulated fatigue effects considering the changes in the α and θ bands, patterns that were observed in previous EEG studies, it is worth noting that other alternative approaches can also be explored. In fact, there is still a lack of consensus within the community regarding the EEG effects of mental fatigue [40]. Moreover, further experiments are crucial to assess if the synthetic fatigue effect resembles the genuine mental fatigue impact on the EEG. Importantly, our simulation of fatigue effects in the EEG highlights the possibility of introducing other task-unrelated cognitive states into artificial data generation. This implies that any additional cognitive state can be simulated, provided that the underlying brain dynamics are well-known in the temporal, spectral, and spatial domains. Addressing this aspect is part of our future plans to enhance the realism of the simulated EEG.

Finally, a data augmentation strategy based on our user-specific simulated MI-EEG data was proposed. Experimental results on FBCNet exhibited that the performance of DL models can be effectively improved when these simulated data is introduced within the training phase. It is timely to mention that such improvements were found by adding only 2 simulated samples in each training batch, a quantity three times smaller than the lowest found in the literature [14]. Moreover, our method yields a significant improvement over the baseline and the SR, BF, CD, STM and WGAN methods. This difference could be attributed to the fact that SR, BF, CD and STM generate data by simply corrupting the original trials, and WGAN focuses on generating data with the closest distribution as possible to the original dataset. Given this, neither of these methods guarantees the preservation of relevant neurophysiological information for accurate classification in the generated data. The lack of significant difference between PySimMIBCI and GNA suggests that similar classification performances can be achieved by either of these two strategies. Nevertheless, a major advantage of PySimMIBCI is that it takes into consideration neurophysiological principles and it is based on a model that enables simulation of many realistic scenarios, as opposed to GNA, which merely adds noise to real data. Moreover, an evaluation of PySimMIBCI was conducted across four additional EEG decoding models with varying degrees of complexity and diverse input types (refer to Supplementary Material, Section 6). The results suggest that the proposed method holds promise as a data augmentation strategy, especially in models proficient at extracting band power features in relevant frequency bands. These findings align with the essence of PySimMIBCI, which generates MI-EEG data by incorporating user-specific spectral information.

When moving towards the particular application of MI-BCIs for rehabilitation, effective data augmentation strategies would enable the training of decoding models with just a few trials from the target user. The beginning of the feedback phase could be acceler-

ated and thus the applicability and usability of these systems could be greatly improved. In addition, PysimMIBCI for data augmentation could help reduce the levels of BCI inefficiency and alleviate frustration in users unable to generate discriminable patterns necessary for controlling an MI-BCI system [59]. However, analyses with real patient data and under rehabilitation conditions are needed to assess the continued efficacy of our method in this particular scenario.

It is important to mention that the proposed framework can be further improved or expanded in different aspects, some of which are mentioned below. Firstly, only hand MI classes were tested throughout this study. Nevertheless, it is important to highlight that PySimMIBCI can be easily extended to other MI-BCI tasks as long as the spatial, waveform and events' properties are known. Hence, in the future, we will study the generation of other types of MI that are relevant in the field of motor rehabilitation, such as hand MI vs. rest [60] and multi-class scenarios. Moreover, other user-specific information, such as personalized spatial location of generator dipoles in the cortex and the use of personalized head models will be included in the data generation process. Furthermore, it is worth noting that during our data augmentation experiments, the training batches were structured to include a fixed number of real and simulated trials. This might lead to a learning problem caused when training batches are constructed with multi-domain data. That is, training batches have a mix of simulated and real data as in a multi-domain problem. From this perspective, future studies will explore ideas to improve gradients' interference during multi-domain data training such as those of gradient surgery [61].

In conclusion, we presented and validated PySimMIBCI, a framework for generating user-specific and meaningful MI-EEG-like data. These data might be useful not only during the development of new decoding models but also as a plausible data augmentation strategy for DL-based MI-BCIs. Therefore, we strongly believe that this approach holds significant potential for advancing the development of more accurate decoding models for MI-BCI systems. In addition, our framework is open source, freely available, and compatible with implementations in MNE-Python [20], thereby promoting accessibility for a wide community of researchers.

References

[1] Simanto Saha et al. "Progress in brain computer interface: Challenges and opportunities". In: *Frontiers in Systems Neuroscience* 15 (2021), p. 578875.

[2] Jelena Mladenović et al. "Towards identifying optimal biased feedback for various user states and traits in motor imagery BCI". In: *IEEE Transactions on Biomedical Engineering* 69.3 (2021), pp. 1101–1110.

[3] LEH Van Dokkum, T Ward, and Isabelle Laffont. "Brain computer interfaces for neurorehabilitation-its current status as a rehabilitation strategy post-stroke". In: *Annals of Physical and Rehabilitation Medicine* 58.1 (2015), pp. 3–8.

[4] Yannick Roy et al. "Deep learning-based electroencephalography analysis: a systematic review". In: *Journal of Neural Engineering* 16.5 (2019), p. 051001.

[5] Fabien Lotte. "Generating artificial EEG signals to reduce BCI calibration time". In: *5th International Brain-Computer Interface Workshop*. 2011, pp. 176–179.

[6] Sung-Jin Kim, Dae-Hyeok Lee, and Yeon-Woo Choi. “CropCat: Data Augmentation for Smoothing the Feature Distribution of EEG Signals”. In: *2023 11th International Winter Conference on Brain-Computer Interface (BCI)*. IEEE. 2023, pp. 1–4.

[7] Yu Pei et al. “Data augmentation: Using channel-level recombination to improve classification performance for motor imagery EEG”. In: *Frontiers in Human Neuroscience* 15 (2021), p. 645952.

[8] Mostafa Neo Mohsenvand, Mohammad Rasool Izadi, and Pattie Maes. “Contrastive representation learning for electroencephalogram classification”. In: *Machine Learning for Health*. PMLR. 2020, pp. 238–253.

[9] Fang Wang et al. “Data augmentation for EEG-based emotion recognition with deep convolutional neural networks”. In: *International Conference on Multimedia Modeling*. Springer. 2018, pp. 82–93.

[10] Aaqib Saeed et al. “Learning from heterogeneous eeg signals with differentiable channel reordering”. In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, pp. 1255–1259.

[11] Yuanzhe Dong et al. “An Approach for EEG Data Augmentation Based on Deep Convolutional Generative Adversarial Network”. In: *2022 IEEE International Conference on Cyborg and Bionic Systems (CBS)*. IEEE. 2023, pp. 347–351.

[12] Fatemeh Fahimi et al. “Generative adversarial networks-based data augmentation for brain-computer interface”. In: *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[13] Sujit Roy et al. “MIEEG-GAN: generating artificial motor imagery electroencephalography signals”. In: *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2020, pp. 1–8.

[14] Qiqi Zhang and Ying Liu. “Improving brain computer interface performance by data augmentation with conditional deep convolutional generative adversarial networks”. In: *arXiv preprint arXiv:1806.07108* (2018).

[15] Yonghao Song et al. “Common Spatial Generative Adversarial Networks based EEG Data Augmentation for Cross-Subject Brain-Computer Interface”. In: *arXiv preprint arXiv:2102.04456* (2021).

[16] Nik Khadijah Nik Aznan et al. “Simulating brain signals: Creating synthetic EEG data via neural-based generative models for improved SSVEP classification”. In: *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2019, pp. 1–8.

[17] Rongrong Fu, Yaodong Wang, and Chengcheng Jia. “A new data augmentation method for EEG features based on the hybrid model of broad-deep networks”. In: *Expert Systems with Applications* 202 (2022), p. 117386.

[18] Elham Barzegaran et al. “EEGSourceSim: A framework for realistic simulation of EEG scalp data using MRI-based forward models and biologically plausible signals and noise”. In: *Journal of Neuroscience Methods* 328 (2019), p. 108377.

[19] Jussi T Lindgren et al. “simBCI—a framework for studying BCI methods by simulated EEG”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26.11 (2018), pp. 2096–2105.

[20] Alexandre Gramfort et al. “MEG and EEG Data Analysis with MNE-Python”. In: *Frontiers in Neuroscience* 7.267 (2013), pp. 1–13. DOI: 10.3389/fnins.2013.00267.

[21] David Sabbagh et al. “Predictive regression modeling with MEG/EEG: from source power to signals and cognitive states”. In: *NeuroImage* 222 (2020), p. 116893.

[22] Lucas C Parra et al. “Recipes for the linear analysis of EEG”. In: *Neuroimage* 28.2 (2005), pp. 326–341.

[23] Scott Makeig et al. “Independent component analysis of electroencephalographic data”. In: *Advances in Neural Information Processing Systems* 8 (1995).

[24] Sylvain Baillet, John C Mosher, and Richard M Leahy. “Electromagnetic brain mapping”. In: *IEEE Signal processing magazine* 18.6 (2001), pp. 14–30.

[25] Christoph M Michel and Denis Brunet. “EEG source imaging: a practical review of the analysis steps”. In: *Frontiers in Neurology* 10 (2019), p. 325.

[26] Andrea Zimmermann-Schlatter et al. “Efficacy of motor imagery in post-stroke rehabilitation: a systematic review”. In: *Journal of Neuroengineering and Rehabilitation* 5.1 (2008), pp. 1–10.

[27] H Chris Dijkerman et al. “Does motor imagery training improve hand function in chronic stroke patients? A pilot study”. In: *Clinical Rehabilitation* 18.5 (2004), pp. 538–549.

[28] Gert Pfurtscheller and FH Lopes Da Silva. “Event-related EEG/MEG synchronization and desynchronization: basic principles”. In: *Clinical Neurophysiology* 110.11 (1999), pp. 1842–1857.

[29] Gert Pfurtscheller and Christa Neuper. “Motor imagery activates primary sensorimotor area in humans”. In: *Neuroscience letters* 239.2-3 (1997), pp. 65–68.

[30] Yongwoong Jeon et al. “Event-related (De) synchronization (ERD/ERS) during motor imagery tasks: Implications for brain–computer interfaces”. In: *International Journal of Industrial Ergonomics* 41.5 (2011), pp. 428–436.

[31] Gert Pfurtscheller et al. “Mu rhythm (de) synchronization and EEG single-trial classification of different motor imagery tasks”. In: *NeuroImage* 31.1 (2006), pp. 153–159.

[32] Christophe Destrieux et al. “Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature”. In: *Neuroimage* 53.1 (2010), pp. 1–15.

[33] Sophocles J Orfanidis. *Introduction to signal processing*. Prentice-Hall, Inc., 1995.

[34] Michael Tangermann et al. “Review of the BCI competition IV”. In: *Frontiers in Neuroscience* 6 (2012), p. 55.

[35] Maliha Hossain. “Whitening and coloring transformations for multivariate Gaussian data”. In: *A lecture partly based on the ECE662 Spring* (2014).

[36] Anna Rita Bentivoglio et al. “Analysis of blink rate patterns in normal subjects”. In: *Movement disorders* 12.6 (1997), pp. 1028–1034.

[37] Kristine B Walhovd, Hannah Rosquist, and Anders M Fjell. “P300 amplitude age reductions are not caused by latency jitter”. In: *Psychophysiology* 45.4 (2008), pp. 545–553.

[38] Yvonne Tran et al. “The influence of mental fatigue on brain activity: Evidence from a systematic review with meta-analyses”. In: *Psychophysiology* 57.5 (2020), e13554.

[39] Teng Cao et al. “Objective evaluation of fatigue by EEG spectral analysis in steady-state visual evoked potential-based brain-computer interfaces”. In: *Biomedical Engineering online* 13.1 (2014), pp. 1–13.

[40] Thomas Jacquet et al. “Mental fatigue induced by prolonged motor imagery increases perception of effort and the activity of motor areas”. In: *Neuropsychologia* 150 (2021), p. 107701.

[41] Saroj KL Lal and Ashley Craig. “Driver fatigue: electroencephalography and psychological assessment”. In: *Psychophysiology* 39.3 (2002), pp. 313–321.

[42] JO Phipps-Nelson, Jennifer R Redman, and Shantha MW Rajaratnam. “Temporal profile of prolonged, night-time driving performance: breaks from driving temporarily reduce time-on-task fatigue but not sleepiness”. In: *Journal of Sleep Research* 20.3 (2011), pp. 404–415.

[43] Leonard J Trejo et al. “EEG-based estimation and classification of mental fatigue”. In: *Psychology* 6.05 (2015), p. 572.

[44] Upasana Talukdar and Shyamanta M Hazarika. “Estimation of mental fatigue during EEG based motor imagery”. In: *Intelligent Human Computer Interaction: 8th International Conference, IHCI 2016, Pilani, India, December 12-13, 2016, Proceedings 8*. Springer. 2017, pp. 122–132.

[45] Thomas Donoghue et al. “Parameterizing neural power spectra into periodic and aperiodic components”. In: *Nature Neuroscience* 23.12 (2020), pp. 1655–1665.

[46] Connor Shorten and Taghi M Khoshgoftaar. “A survey on image data augmentation for deep learning”. In: *Journal of big data* 6.1 (2019), pp. 1–48.

[47] Donald B Percival and Andrew T Walden. *Spectral analysis for physical applications*. cambridge University press, 1993.

[48] Ravikiran Mane et al. “FBCNet: A Multi-view Convolutional Neural Network for Brain-Computer Interface”. In: *arXiv preprint arXiv:2104.01233* (2021).

[49] Min-Ho Lee et al. “EEG dataset and OpenBMI toolbox for three BCI paradigms: An investigation into BCI illiteracy”. In: *GigaScience* 8.5 (2019), giz002.

[50] Adam Paszke et al. “Pytorch: An imperative style, high-performance deep learning library”. In: *Advances in Neural Information Processing Systems* 32 (2019).

[51] Fabien Lotte. “Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain–computer interfaces”. In: *Proceedings of the IEEE* 103.6 (2015), pp. 871–890.

[52] Herbert Ramoser, Johannes Muller-Gerking, and Gert Pfurtscheller. “Optimal spatial filtering of single trial EEG during imagined hand movement”. In: *IEEE Transactions on Rehabilitation Engineering* 8.4 (2000), pp. 441–446.

[53] Yijun Wang, Shangkai Gao, and Xiaorong Gao. “Common spatial pattern method for channel selection in motor imagery based brain-computer interface”. In: *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*. IEEE. 2006, pp. 5392–5395.

[54] Milton Friedman. “The use of ranks to avoid the assumption of normality implicit in the analysis of variance”. In: *Journal of the American Statistical Association* 32.200 (1937), pp. 675–701.

[55] Peter Bjorn Nemenyi. *Distribution-free multiple comparisons.* Princeton University, 1963.

[56] Maarten A Immink et al. “Resting-state aperiodic neural dynamics predict individual differences in visuomotor performance and learning”. In: *Human Movement Science* 78 (2021), p. 102829.

[57] Eleanor A Curran and Maria J Stokes. “Learning to control brain activity: A review of the production and control of EEG components for driving brain–computer interface (BCI) systems”. In: *Brain and Cognition* 51.3 (2003), pp. 326–336.

[58] Ruyi Foong et al. “Assessment of the efficacy of EEG-based MI-BCI with visual feedback and EEG correlates of mental fatigue for upper-limb stroke rehabilitation”. In: *IEEE Transactions on Biomedical Engineering* 67.3 (2019), pp. 786–795.

[59] Guido Dornhege et al. “An Introduction to Brain-Computer Interfacing”. In: *Toward Brain-Computer Interfacing.* 2007, pp. 1–25.

[60] Kai Keng Ang et al. “Facilitating effects of transcranial direct current stimulation on motor imagery brain-computer interface with robotic feedback for stroke rehabilitation”. In: *Archives of Physical Medicine and Rehabilitation* 96.3 (2015), S79–S87.

[61] Lucas Mansilla et al. “Domain generalization via gradient surgery”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2021, pp. 6630–6638.

Supplementary material to: Neurophysiologically meaningful motor imagery EEG simulation with applications to data augmentation

C. M. Galván, R. D. Spies, D. H. Milone and V. Peterson

1 Distribution of periodic parameters in real MI-EEG data

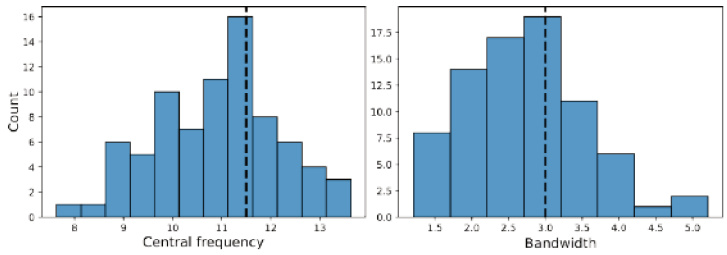

Figure A1: Histograms of central frequency and bandwidth of α peak across all the real users in Dataset-1 and Dataset-2. The dotted lines indicate the mode of each parameter.

2 PySimMIBCI results for different sampling methods

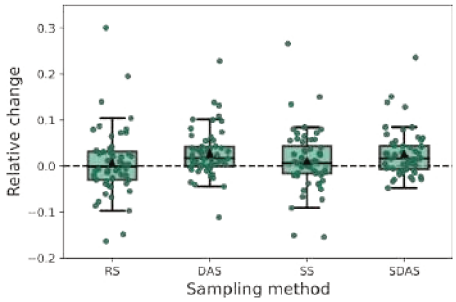

Figure A2: Data augmentation results in two real MI-EEG datasets for the FBCNet model employing different sampling methods for the construction of mini-batches. RS: random sampling, DAS: domain-aware sampling (all mini-batches have the same ratio of real/augmented samples), SS: stratified sampling (all mini-batches are stratified by the output class), SDAS: stratified domain-aware sampling (mini-batches are stratified by output class and the ratio of real/augmented samples is fixed). According to the Nemenyi test, the SDAS and DAS are significantly better than the baseline and than RS and SS.

3 PySimMIBCI results for different N_a

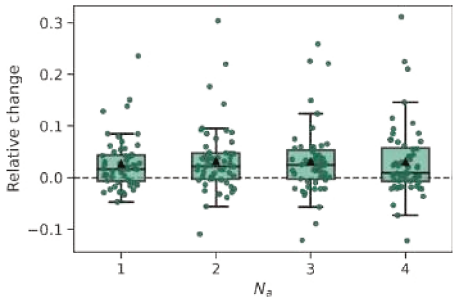

Figure A3: Data augmentation results in two real MI-EEG datasets for the FBCNet model employing different values for N_a (number of simulated trials from each class included in each batch). According to the Friedman test, there is no statistically significant difference between the median values of the different methods.

4 Similarity between real and simulated data by cross-correlation

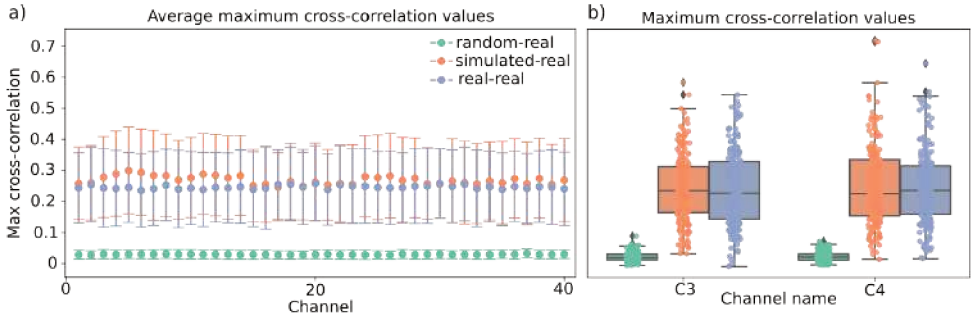

Figure A4: Results of the cross-correlation analysis between different pairs of data: random-real (green), simulated-real (orange) and real-real (blue). To focus the analysis in the band where the ERDs were simulated, data was filtered in the α band. Thus, the cross-correlation was computed between pairs of α-band trials from the same MI class and between corresponding channels. a) Average of maximum cross-correlation values for each pair of EEG channels. b) Maximum cross-correlation values for each pair of MI-EEG trials in channels C3 and C4.

5 Ablation study

The performance achieved by the augmentation method when each of the new blocks is introduced to the state-of-the-art method simBCI [1] was measured:

- PySimMIBCI only periodic: it includes only the user-specific periodic information block.
- PySimMIBCI only aperiodic: it includes only the user-specific aperiodic information block.
- PySimMIBCI only em: it includes only the eye-movement artifact block.

The results obtained for a simBCI-like version and for the PySimMIBCI are also provided here. This ablation study was conducted on data from both subjects of Dataset-1 and FBCNet model was used. As it can be observed, when including each one of the blocks to base simBCI model, better data augmentation performances are obtained for both subjects.

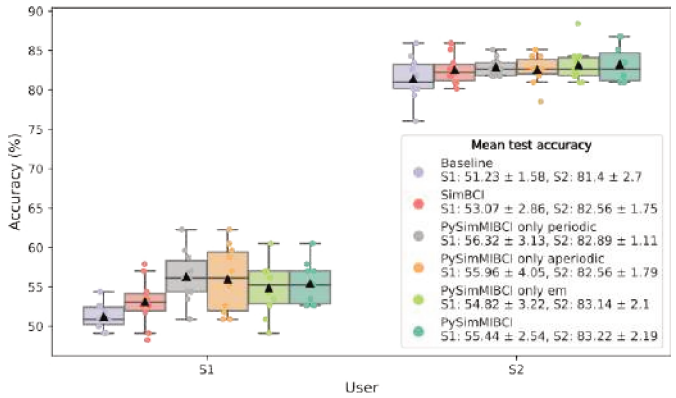

Figure A5: Ablation study results in Dataset-1 for the FBCNet model. Each point corresponds to the test accuracy obtained for one model initialization seed. The average accuracy across all seeds users is shown by a black triangle.

6 Data augmentation results for other deep learning models

Deep learning models with varying characteristics were selected for this analysis: 1) different levels of complexity, and 2) different input type, that is, models that process either the raw signal or its spectral characteristics (filtered signal in bands). In the following, we briefly described them.

- EEGNet [2]: This is a compact convolutional neural network architecture for EEG-based BCIs. It comprises the following blocks: 1) temporal convolution layer to learn frequency filters, 2) depthwise convolution, connected to each feature map individually, to learn frequency-specific spatial filters, 3) separable convolution, which is a combination of a depthwise convolution that learns a temporal summary for each feature map individually, followed by a pointwise convolution, which learns how to optimally mix the feature maps together and 4) classification block, where features are directly passed to a softmax classification layer.

- Deep ConvNet [3]: This model is inspired by successful architectures in computer vision and it was designed as a generic architecture with minor expert knowledge, that extracts a wide range of features and is not restricted to specific feature types. This models comprises a special first block that performs a temporal convolution and a spatial filtering, followed by three standard convolution-max-pooling blocks and a dense softmax classification layer.

- Shallow ConvNet [3]: It is inspired in [4]. The first two layers of this network perform a temporal convolution and a spatial filter, as in the deep ConvNet. These steps are comparable to the bandpass and CSP spatial filter steps in [4]. After that, a squaring nonlinearity, a mean pooling layer and a logarithmic activation function

follow; together these steps are analogous to the trial log-variance computation in [4].

- FB Shallow ConvNet: It is a modified version of the Shallow ConvNet, replacing the first layer of this model (temporal convolution), with the original filter bank of FBCSP and FBCNet [5]. In this way, we ensure that the model processes spectral features of the signal filtered in different bands of interest.

The experimental setup, the datasets and the training criteria were the same as in the cross-session within-subject scenario described in the main manuscript. The results are presented in Table A1.

	EEGNet	Shallow ConvNet	Deep ConvNet	FB Shallow ConvNet
Baseline	57.89 ± 3.51	59.62 ± 3.33	58.62 ± 3.27	61.39 ± 2.98
PySimMIBCI	58.24 ± 3.94	53.98 ± 3.67	58.38 ± 3.24	59.01 ± 2.89
GNA	58.47 ± 3.84	61.17 ± 3.21	59.40 ± 3.37	62.55 ± 2.73
SR	57.80 ± 3.67	60.54 ± 3.31	58.43 ± 3.14	62.36 ± 2.70
STM	58.53 ± 3.82	61.55 ± 3.30	59.81 ± 3.53	62.96 ± 2.42
BF	58.33 ± 3.77	61.62 ± 2.74	59.68 ± 3.29	61.99 ± 2.21
CD	59.25 ± 3.68	61.85 ± 2.59	60.55 ± 3.40	61.85 ± 2.40
WGAN	58.36 ± 3.93	53.84 ± 3.61	58.24 ± 2.93	58.92 ± 3.00

Table A1: Data augmentation results in two real MI-EEG datasets in a cross-session within-subject scenario. The values in each cell correspond to the average test accuracy and the standard deviation across the 56 real users.

7 FBCNet architecture

Block	Layer type	# Kernels	Kernel size	Output shape	# Parameters
Input				$(1, C, T)$	
Filter-Bank				(N_b, C, T)	
Spatial Convolution	Depthwise Conv2D	$m \cdot N_b$	$(C, 1)$	$(m \cdot N_b, 1, T)$	$m{\cdot}N_b{\cdot}C{+}m{\cdot}N_b$
	BatchNorm			$(m \cdot N_b, 1, T)$	$2 \cdot m \cdot N_b$
	Activation - ELU			$(m \cdot N_b, 1, T)$	
Temporal Feature Extraction	Variance Layer		$(1, w)$	$(m \cdot N_b, 1, T/w)$	
Classifier	Flatten			$m \cdot N_b \cdot T/w$	
	Linear	N_c		N_c	$m \cdot N_b \cdot T/w \cdot N_c + N_c$
	Log-Softmax			N_c	

Table A2: Implementation details of FBCNet model. C: number of EEG channels, T: number of time points, N_b: number of frequency bands, m: number of convolution filters per frequency band, N_c: number of output classes, w: temporal window length. The optimal setting recommended by the authors was employed, i.e. $N_b = 9$, $m = 32$, $w = 125$. In all the cases, N_c was equal to 2 and T was 500. This led to 14978 parameters for Dataset-1 ($N_c = 41$) and 8930 parameters for Dataset-2 ($N_c = 20$) analysis.

References

[1] Jussi T Lindgren et al. "simBCI—a framework for studying BCI methods by simulated EEG". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26.11 (2018), pp. 2096–2105.

[2] Vernon J Lawhern et al. "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces". In: *Journal of neural engineering* 15.5 (2018), p. 056013.

[3] Robin Tibor Schirrmeister et al. "Deep learning with convolutional neural networks for EEG decoding and visualization". In: *Human brain mapping* 38.11 (2017), pp. 5391–5420.

[4] Kai Keng Ang et al. "Filter bank common spatial pattern (FBCSP) in brain-computer interface". In: *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*. IEEE. 2008, pp. 2390–2397.

[5] Ravikiran Mane et al. "FBCNet: A Multi-view Convolutional Neural Network for Brain-Computer Interface". In: *arXiv preprint arXiv:2104.01233* (2021).

Anexo B

No need to re-train for never seen subjects: cross-subject domain adaptation in rehabilitative BCIs via backward optimal transport

No need to re-train for new users: cross-subject domain adaptation in rehabilitative BCIs via backward optimal transport

C. M. Galván, D. H. Milone, R. D. Spies, and V. Peterson

C. M. Galván, V. Peterson and R. D. Spies are at Instituto de Matemática Aplicada del Litoral, IMAL, UNL, CONICET, Santa Fe, Argentina
(e-mail: cgalvan@santafe-conicet.gov.ar, vpeterson@santafe-conicet.gov.ar, rspies@santafe-conicet.gov.ar)
D. H. Milone is at Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional, sinc(i), FICH-UNL, CONICET, Santa Fe, Argentina
(email: dmilone@sinc.unl.edu.ar)

Abstract

Objective: This work addresses the inter-subject variability of motor imagery (MI)-based brain-computer interfaces (BCIs) for rehabilitation in order to minimize the calibration phase required for new users. *Methods*: We propose cross-subject backward optimal transport (XS-BOT), a cross-subject and cross-dataset domain adaptation framework that requires minimal recalibration data from the target subject. Starting from a model trained on a large group of source subjects, XS-BOT aligns the distribution of target subject trials with that of training (source) data. In rehabilitation, the BCI cues the user when and which MI task to perform, but the user generated EEG patterns might not match the cued label. Our method exploits the availability of this cued label to guide the adaptation at feature level, avoiding model retraining. We validate XS-BOT on five public datasets (193 subjects) with widely used decoding models, representing traditional machine learning and deep learning approaches. *Results*: XS-BOT consistently outperforms state-of-the-art transfer learning and classical approaches across all datasets. High decoding accuracy is obtained with as few as 3 EEG channels and 20 adaptation trials, demonstrating robust performance under minimal calibration and channel setup. *Conclusions*: The proposed framework mitigates inter-subject and inter-dataset variability in MI-BCIs while avoiding the need for model retraining and substantially reducing calibration requirements. *Significance*: By minimizing calibration time and channel configuration, XS-BOT provides an efficient adaptation strategy that promotes the practical

use of MI-BCIs in rehabilitation and preserves valuable therapy time.

Keywords

Brain-computer interface, Deep Learning, Domain Adaptation, Electroencephalography, Motor Imagery, Optimal Transport.

1 Introduction

Addressing signal variability is a crucial step to build brain-computer interfaces (BCIs) for long-term and widespread use [1]. Due to significant inter-subject variability in electroencephalography (EEG) signals, typical machine learning (ML) models result in poor user intention decoding when used across subjects for motor imagery (MI) classification [2]. Consequently, most MI-BCI systems either rely on training a subject-specific model (within-subject approach) or necessitate re-calibrating previously trained models for each new user [3]. In either of these two scenarios, a considerable amount of labeled training data from the target subject is required. When used as rehabilitation neurotechnologies for motor function recovery after stroke [4–6], the acquisition of adaptation data and the followed-up model set-up should consume the minimum time of the therapy session. In this way, feedback resembling the user's intention can be delivered to the patient as soon as possible [7], resulting in more effective and user-friendly rehabilitation sessions.

Several MI-EEG datasets are publicly available [**cho2017eeg**, 8, 9], registered across different conditions. Leveraging large available data, across-subjects MI decoding models can be built. Taking into account the inter-subject variability, transfer learning techniques can be imposed to sort domain changes between source data (training set composed of multiple subjects's data) and data from an unseen target subject. In this work, we present cross-subject backward optimal transport (XS-BOT), a framework built on the principles of backward optimal transport for domain adaptation (BOTDA) [10]. Considering rehabilitation scenarios, where the instructed task is always known but the actual mental task performed by the user is uncertain and must be decoded, XS-BOT uses this cued label as a valuable supervision signal to "guide" the adaptation. Starting from a pre-trained MI decoding model at a large group of source subjects, XS-BOT is applied at the features' space, bridging the distribution of unseen target subject's features closer to the features distribution of the source domain, avoiding recalibration of both the feature extraction and classification blocks. Our framework uses only a small number of labeled trials from the target subject to estimate the optimal transport plan, coming from as few as three EEG channels. Thus, by means of XS-BOT we i) reduce to the minimum the EEG set-up and calibration time while ii) cross-subject variability is addressed. Results in different and large datasets comprising 139 test subjects show that XS-BOT can substantially improve MI decoding, being over 19% better than state-of-the-art adaptation methods, for both statistical-ML and deep learning (DL) pipelines. This makes XS-BOT a suitable method for fast calibration and follow-up model set-up in the context of MI-BCI rehabilitation, fostering the practical use of these systems.

2 Related works

MI decoding from EEG signals has traditionally relied on band power features extracted through spatial filtering, i.e., channel-based demixing methods used to estimate putative brain sources [11]. One of the most established methods is the common spatial patterns (CSP) [12], a supervised feature extraction method that maximizes variance differences between MI classes. CSP features are then typically input of linear classifiers, such as the linear discriminant analysis (LDA). Although CSP+LDA is widely accepted as the gold-standard for MI decoding in within-subject scenarios, it shows limited cross-subject generalization capability due to high variability in EEG signals [11].

More recently, DL approaches emerged as powerful alternatives in the context of EEG-BCIs. Among them, EEGNet [13], a convolutional architecture designed to efficiently capture EEG patterns, became widely adopted due to its strong performance in MI and other BCI tasks. Although numerous other DL architectures and approaches have been proposed in recent years for MI decoding [14, 15], many lack reproducible implementations [16, 17], or have not been evaluated in cross-subject scenarios [18, 19]. Additionally, none have consistently surpassed the widespread adoption of EEGNet [20, 21] or its precursors, DeepConvNet and ShallowConvNet [22–24], establishing them as DL baselines in MI-BCI context. Thus, here we focus on EEGNet [13] and for completeness provide in the supplementary material results on DeepConvNet and ShallowConvNet [22], as well as on two more recent networks [15, 18] that have publicly available implementations. It is important to mention here that the original EEGNet publication includes a cross-subject analysis, with mean accuracies below 40% for a 4-class classification problem. Additionally, a more recent work [20] studied the use of EEGNet for transfer learning based on fine-tuning and achieved an average accuracy of 66.36% for a 2-class MI decoding, showing the difficulty of cross-subject transfer learning. To address the challenge of inter-subject variability in MI decoding, several techniques have been explored. Many CSP variants [25–32] have been investigated. In this line, the recently introduced generic recentering CSP (GR-CSP) [33] applies affine transformations to the trial-specific covariance matrices utilized by the CSP algorithm. On the other hand, for DL architectures, fine-tuning strategies have been the prevalent framework for adapting pre-trained models to new subjects [20, 23, 24]. Complementary to model adaptation, data alignment techniques offer a general approach applicable across different decoding pipelines. Among them, euclidean alignment (EA) [34] stands out for its simplicity and effectiveness [35]. This unsupervised technique aligns data by transforming covariance matrices into the Euclidean space using the arithmetic mean of covariance matrices. More recently, unsupervised approaches based on projections using optimal transport barycenters have also been explored, such as convolution monge mapping normalization (CMMN) [36], originally proposed for sleep staging.

Although these techniques represent advancements in tackling inter-subject variability and enhancing the practicability of MI-BCI systems, it should be highlighted that the vast majority of MI decoding pipelines have been evaluated and validated on EEG data from a large number of channels, usually exceeding 20 electrodes [12, 13, 22]. Consequently, the performance and generalizability of these established pipelines in scenarios with a small channel count, which is more convenient for real-world applications, remain unexplored. This work aims not only to enhance minimal calibration MI decoding, but also to show that accurate performance is achievable with a reduced three-channel setup, ultimately advancing towards user-friendly and readily deployable MI-BCI systems.

3 Addressing inter-subject variability in MI-BCIs via optimal transport

One of the biggest challenges in MI-BCIs is the inherent variability between subjects and datasets, driven by individual differences in brain anatomy, cognitive stages, electrode placement and recording conditions [2]. These factors hinder the generalization of models trained on one subject or group of subjects to others not seen during training [37]. In this section, first we describe the foundational BOTDA approach [10] for domain adaptation in MI-BCIs. Then, we present how XS-BOT adapts its core concepts to achieve effective domain adaptation in cross-subject and cross-dataset scenario, reducing the need for extensive subject-specific adaptation data and contributing to the development of more practical and user-friendly MI-BCI systems.

The EEG variability in MI-BCIs makes them naturally suited for domain adaptation, where the goal is to address the performance drop that occurs when a model trained on source data is applied to target data with similar but different distribution [38]. Among domain adaptation techniques, BOTDA [10] has shown remarkable results in the context of MI-BCIs. BOTDA leverages principles from optimal transport theory to align data distributions by applying a backward mapping from target to source feature space. Originally proposed for within-subject cross-session adaptation [10], it represents a flexible framework that can be extended to a variety of domain shifts. This section provides a brief introduction to BOTDA formulation, which serves as the basis for our framework.

Here, the source domain Ω_s contains labeled samples to train the decoding model, whereas the target domain Ω_t is where testing data come from. Both Ω_s and Ω_t have an associated dataset:

$$\mathcal{S} \doteq \left\{\left(\mathbf{x}_j^s, y_j^s\right)\right\}_{j=1}^{N_s} \subset \Omega_s \times \mathcal{K}, \quad \mathcal{T} \doteq \left\{\left(\mathbf{x}_i^t, y_i^t\right)\right\}_{i=1}^{N_t} \subset \Omega_t \times \mathcal{K}, \tag{1}$$

where N_s and N_t denote the number of trials in the source and target datasets, respectively. Vectors $\mathbf{x}_j^s \in \Omega_s$, $\mathbf{x}_i^t \in \Omega_t$ are the features extracted from EEG signals, with $\Omega_s, \Omega_t \subset R^d$, and $y_j^s, y_i^t \in \mathcal{K}$ represent the indicated mental tasks belonging to K possible MI classes, with $\mathcal{K} \doteq \{k_l\}_{l=1}^K$. It is important to note that, due to the inherent nature of MI-BCIs, these labels do not constitute the ground truth of the mental activity actually performed, but rather the task that was cued [39].

BOTDA addresses the distribution mismatch between Ω_t and Ω_s by estimating an optimal transportation plan γ^* that allows mapping target features into the source feature space. Here, the discrete BOTDA Sinkhorn group-LASSO formulation is considered:

$$\gamma^* \doteq \arg\min_{\gamma} \langle \gamma, M \rangle_F + \lambda W_e(\gamma) + \nu W_c(\gamma), \tag{2}$$

where $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product and $M = (m_{i,j})$, $m_{i,j} \geq 0$ a matrix such that $m_{i,j} = m\left(\mathbf{x}_i^t, \mathbf{x}_j^s\right)$ is the cost of transporting a unit probability mass from $\mathbf{x}_i^t$ to $\mathbf{x}_j^s$, given a cost function m. A common choice for m is the squared Euclidean distance [40]. In (2), $W_e(\gamma)$ and $W_c(\gamma)$ act as regularizers and λ and ν are their respective hyperparameters. Specifically, $W_e(\gamma) \doteq \sum_{ij} \gamma_{i,j} \log(\gamma_{i,j})$ is the negentropy, which favors smoother and less sparse transport plans, whereas $W_c(\gamma) \doteq \sum_j \sum_k \|\gamma_{\mathcal{I}_k,j}\|_2$, with $\mathcal{I}_k$ denoting the set of indices in the target domain that belongs to class $k \in \mathcal{K}$, is the group-lasso term, which encourages samples of the same class to be transported as a group.

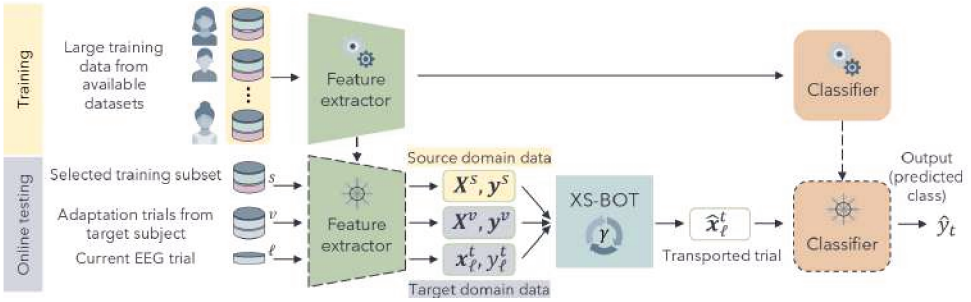

Figure 1: Training and online testing phases of XS-BOT. The decoding model is trained on a large dataset from multiple subjects. During online testing for a trial ℓ, an optimal transport plan γ^* is learned using adaptation trials from the target subject, along with the current trial and its cued label. The current trial's features are transformed using γ^* and then fed into the previously trained classifier.

Estimating γ^* requires labeled data from both target and source domain. Let us assume that a small adaptation set of N_v labeled target trials is available. Using this adaptation set, BOTDA selects a source subset of the same size, N_v, along with its optimal regularization parameters λ and ν in a data-driven manner, aiming to maximize transport performance. Both source subset and regularization parameters remain fixed during adaptation. For each trial ℓ, BOTDA performs online adaptation by constructing a target transportation set $\mathcal{V}_\ell$, that combines the adaptation set with the current trial:

$$\mathcal{V}_\ell \doteq \left\{\left(\mathbf{x}_i^t, y_i^t\right)\right\}_{i=1}^{N_v} \cup \left\{\left(\mathbf{x}_\ell^t, y_\ell^t\right)\right\} \subset \mathcal{T}, \tag{3}$$

After γ^* is estimated, and since the current trial ℓ is part of the transportation set, its transformed feature vector $\hat{\mathbf{x}}_\ell^t$ is directly obtained by the following weighted barycenter mapping:

$$\hat{\mathbf{x}}_\ell^t = \sum_j \gamma_{\ell,j}^* \mathbf{x}_j^s, \tag{4}$$

where the weights $\gamma_{\ell,j}^*$ determine how much mass from the target trial $\mathbf{x}_\ell^t$ is transported to each source trial $\mathbf{x}_j^s$. Finally, $\hat{\mathbf{x}}_\ell^t$ can be used to feed an already trained classifier to predict its class label.

In this approach, the cued label for the current trial ℓ is exploited when learning γ^*, specifically only within the group-lasso regularization term. Nevertheless, although this term promotes target samples categorized on the same group to be transported together, such transportation may fail if there is a mismatch between the cued and performed mental task [39].

For cross-subject and cross-dataset transfer learning, the source domain refers to a large and diverse dataset containing labeled samples from various subjects, used to train the decoding model, while the target domain corresponds to a new subject, from a different dataset, unseen during training. The goal of XS-BOT is to adapt the target subject's features so that they align with the source domain's feature distribution, using only a small number of target-domain adaptation trials. A key advantage of XS-BOT is that it avoids model retraining, enabling computationally efficient, real-time adaptation.

The framework, as shown in Figure 1, operates by first training a decoding model using data from multiple subjects, typically from public datasets. This model is trained on large-scale source data to learn generalizable features across a variety of individuals

[23, 41]. Then, for each target subject, a small adaptation set (typically 10 trials per class) is used to select a source subset and the optimal regularization parameters, following a data-driven approach similar to BOTDA (see Supplementary Material for more details). During the online testing phase, the source subset, adaptation set and regularization parameters remain fixed. For each testing trial from the target subject, XS-BOT applies BOTDA to adapt its features, allowing real-time adaptation. The followed steps are:

1. The trained feature extractor is used to compute $\mathbf{x}_\ell^t$, the features vector corresponding to the current target trial.

2. By using the transportation set $\mathcal{V}_\ell$ as defined in (3) and the selected source subset, the optimal transport plan γ^* is computed.

3. The features vector $\mathbf{x}_\ell^t$ is transported according to this optimal plan γ^* to match the feature distribution from the source training data following (4).

4. The (fixed) classifier, which was trained only with the source data, is used to predict the class of the trial using the transformed features vector $\hat{\mathbf{x}}_\ell^t$ as input.

4 Data and experimental setup

Datasets and preprocessing The experiments were conducted using five publicly available EEG datasets of right versus left hand MI tasks from non-impaired subjects. Lee2019_MI [8] dataset was used for training, based on a previous study that showed that this dataset was the best "donator" when tested across a variety of MI-BCI datasets [41]. However, to demonstrate the robustness with respect to training data, in the Supplementary Material we included the results obtained by training on each available dataset and testing on the others. Testing was performed on Cho2017 [**cho2017eeg**], Dreyer2023A, Dreyer2023B, and Dreyer2023C [9]. Lee2019_MI consists of recordings from 54 users, each participating in two sessions with 100 trials per class. EEG signals were recorded at 1000 Hz using 62 electrodes following the international 10-20 system. Cho2017 includes one session data, with 200 to 240 trials from 52 participants. EEG was acquired using 64 channels from the international 10-10 system and sampled at 512 Hz. Dreyer2023 dataset comprises three subsets (A, B, and C) that follow the same MI protocol but address different research questions. It has 240 trials from one session for each subject, with EEG recorded at 512 Hz from 27 electrodes of the international 10-20 system. Subsets A, B, and C include 60, 21, and 6 participants, respectively.

All datasets underwent the same preprocessing. Raw signals were epoched from 0 to 3 seconds after the MI instruction cue. To ensure a common reference across datasets and recording setups, signals were re-referenced to the Fz channel. Each epoch was then band-pass filtered between 4 and 40 Hz using an Butterworth filter and resampled to 128 Hz. Finally, epochs were cropped to the interval between 0.5 and 2.5 seconds, which served as input to the decoding model. For analysis, only C3, Cz, and C4 channels were used to focus on MI-relevant areas while minimizing electrode count [41].

MI decoding pipelines and evaluation setup We evaluated XS-BOT at two types of MI decoding pipelines: a traditional approach based on CSP followed by LDA, and a DL-based approach defined by EEGNet. For CSP+LDA, EEG trials were bandpass filtered between 8–30 Hz [12]. Then, the trials were spatially filtered using two CSP

components and features were computed as the log-average power of the projected signals. The resulting two-dimensional features were used to feed an LDA classifier. For DL learning models, each trial was z-normalized using its own mean and standard deviation prior to model input. For EEGNet, we used the default EEGNet-8,2 architecture [13]. Instead of the original classification block, a shallow multilayer perceptron was used for classification. Latent features before classification were extracted. EEGNet was implemented in PyTorch and trained with AdamW ($lr = 0.001$) and a cross-entropy loss for 500 epochs. Further details on model architectures and hyperparameters are provided in the Supplementary Material.

XS-BOT was evaluated in a cross-subject and cross-dataset scenario, with minimal target adaptation data. For each target subject in Cho2017 and Dreyer2023 datasets, a small labeled subset of trials $N_v = 20$ (first 10 trials from each class) was used as adaptation set. This value for N_v matchs the one used in the original BOTDA work [10]. The remaining trials were reserved for testing, to simulate online real-life implementation. The source data consisted of the Lee2019_MI dataset (54 subjects), used entirely for training, with no data held out for validation.

Metrics We evaluated decoding performance using classification accuracy (proportion of correctly predicted labels over the test trials). We also used class distinctiveness (D) [42] to assess the separability of MI classes. This Riemannian geometry-based metric is given by

$$D = \frac{\delta_R\left(\overline{C}_A, \overline{C}_B\right)}{\frac{1}{2}\left(\sigma_{C_A} + \sigma_{C_B}\right)} \tag{5}$$

where $\overline{C}_k$ and σ_{C_k} are respectively the mean and mean absolute deviation of the covariance matrices for class k. Unlike accuracy, D does not rely on any classifier, offering an intrinsic measure of how discriminable the data are in terms of their statistical properties. Following [42], we computed D using 20 sensorimotor electrodes (10–20 system) and bandpass-filtered trials (8–30 Hz).

Competing methods We compared XS-BOT against the following baselines and state-of-the-art adaptation strategies:

- Source data training (SDT): The model is trained on source data from multiple subjects without any adaptation to the target subject.
- Adaptation data training (ADT): The model is trained exclusively on the few adaptation trials available from the target subject.
- Euclidean alignment (EA) [34]: Both source and target data are aligned in the Euclidean space prior to training and testing. For each source subject, alignment is performed independently using the mean covariance matrix computed over all of that subject's trials. For the target subject, each trial is aligned using the arithmetic mean of the adaptation trials.
- Generic recentering CSP (GR-CSP)[33]: It aligns trial covariance matrices to the training set's Riemannian mean. At test time, features are extracted from aligned covariance matrices using an affine transformation with an incrementally updated centering matrix.

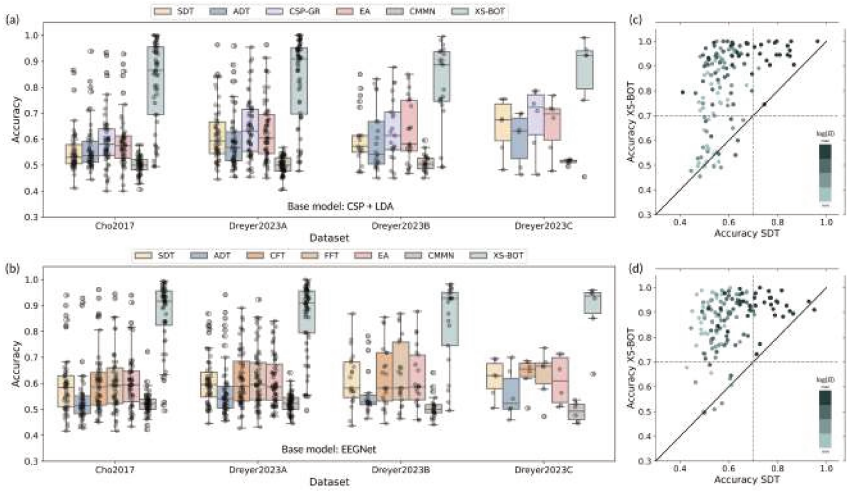

Figure 2: Performance of different adaptation methods in four MI-EEG datasets. Accuracy of SDT, ADT, CSP-GR, EA, FFT, CFT and XS-BOT applied to (a) CSP+LDA and (b) EEGNet. Each point represents one test subject. Panels (c) and (d) compare the accuracy of XS-BOT against the nonadaptive baseline, SDT, on (c) CSP+LDA and (d) EEGNet. In these panels, again each point corresponds to a single subject. The diagonal line means equal performance between the two methods and the dashed grey lines mark the 0.7 accuracy thresholds. The color of each point encodes the log-transformed class distinctiveness of the subject, divided into quartiles, where lighter markers indicate less discriminable MI signals and darker markers represent more distinct signals.

- Convolution monge mapping normalization (CMMN) [36]: It normalizes EEG signals by aligning their power spectral density to a Wasserstein barycenter computed from training data.
- Fine-Tuning strategies for DL models [20, 24]: Models are pretrained on source data and fine-tuned with target adaptation data. Full model fine-tuning (FFT) updates all layers whereas at classifier fine-tuning (CFT) only the classifier layers are updated.

In order to ensure fair comparison, all competing methods, both in the traditional and deep learning frameworks, were evaluated using the same number of adaptation trials from the target subject. The only exception is SDT, which was included as a reference where any form of adaptation is performed.

5 Results

5.1 Boosting MI decoding performance on classical and deep learning models

To assess the effectiveness of XS-BOT, we applied it to both classical (CSP+LDA) and DL-based (EEGNet) MI decoding pipelines, comparing it to competing adaptation methods across the datasets and subjects. Figure 2 (a) shows the results for CSP+LDA. It can be seen that XS-BOT outperforms all competitors, with an average accuracy of 0.820 $\pm$ 0.163 across all subjects (mean $\pm$ standard deviation). The baseline method SDT reaches

an accuracy of 0.594 ± 0.105, whereas the other adaptive methods yield in comparison substantially lower performances: ADT (0.584 ± 0.107), CSP-GR (0.627 ± 0.118), EA (0.617 ± 0.115) and CMMN (0.501 ± 0.034). The Friedman test [43] rejected the null hypothesis of equal central tendencies among the methods ($p \ll 0.001$), and a subsequent post-hoc Nemenyi test [44] confirmed that XS-BOT performance is significantly better than SDT, ADT, EA, CMMN and CSP-GDR, with $p \ll 0.001$ for all corresponding pairwise comparisons. Overall, the relative gain in accuracy over existing methods when XS-BOT is applied is more than 30%. To evaluate if these benefits extend to DL-based decoding, we repeated the analysis but using EEGNet (Figure 2 (b)). XS-BOT again shows a remarkable improvement, with an average accuracy of 0.866 ± 0.109, which far exceeds the nonadaptive baseline (SDT, 0.610 ± 0.110) as well as all other adaptation strategies: ADT (0.555 ± 0.105), AD-CFT (0.617 ± 0.112), AD-FFT (0.616 ± 0.119), EA (0.612 ± 0.106) and CMMN (0.519 ± 0.042). Statistical tests confirmed the robustness of these results ($p \ll 0.001$). Note that all competing methods, in both ML approaches consistently underperform the 0.7 accuracy threshold for effective MI-BCI control [8], a baseline for practical usability. This suggests that all these methods would yield unusable systems.

Despite the consistent improvement in average accuracy across subjects, individual performance of XS-BOT varied from subject to subject. To better understand these differences, we investigated how the characteristics of the target subject relate to the adaptation performance. In particular, we analyzed whether metrics such as class distinctiveness or baseline decoding accuracy on SDT can be associated with the variability in the effectiveness of XS-BOT. Figure 2 (c) and (d) illustrate the impact of adaptation on individual subject performance, comparing decoding accuracy achieved with the nonadaptive baseline (SDT) and the proposed adaptive method (XS-BOT). Each point represents one subject, with the x-axis indicating SDT accuracy and the y-axis XS-BOT accuracy. The solid diagonal line corresponds to equal performance between both methods and grey lines show the 0.7 practical threshold [8]. The color of each point reflects the subject's class distinctiveness, divided into four categories based on the quartiles of the log-transformed class distinctiveness values. Lighter tones indicate lower class distinctiveness (i.e., lower discriminability between MI classes), whereas darker tones represent higher class distinctiveness. This visual encoding allows us to assess how class separability relates to both baseline and adaptive performance.

Panel (c) presents results for CSP+LDA. Most subjects are above the diagonal, indicating consistent gains from adaptation. The few exceptions are subjects with poor initial SDT performance and low class distinctiveness, suggesting that their MI signals may lack sufficient class-discriminative information to support learning an effective transportation plan. For SDT, only MI subjects with very high class distinctiveness (darker points) tend to achieve good performance, shedding light on the idea that in cross-subject BCIs only outstanding MI performing users can decently use the system without data adaptation. Importantly, those subjects with lower distinctiveness benefit more from adaptation, with XS-BOT enabling many of them to surpass the 0.7 accuracy threshold. Panel (d) presents the analogous analysis for EEGNet. Although the overall trends remain consistent with those in panel (c), the improvements introduced by XS-BOT are even more pronounced in this setting. Here, not only does a higher number of subjects surpass the 0.7 threshold, but also the accuracy gains over SDT are often larger, creating a clear gap between XS-BOT and the nonadaptive SDT baseline.

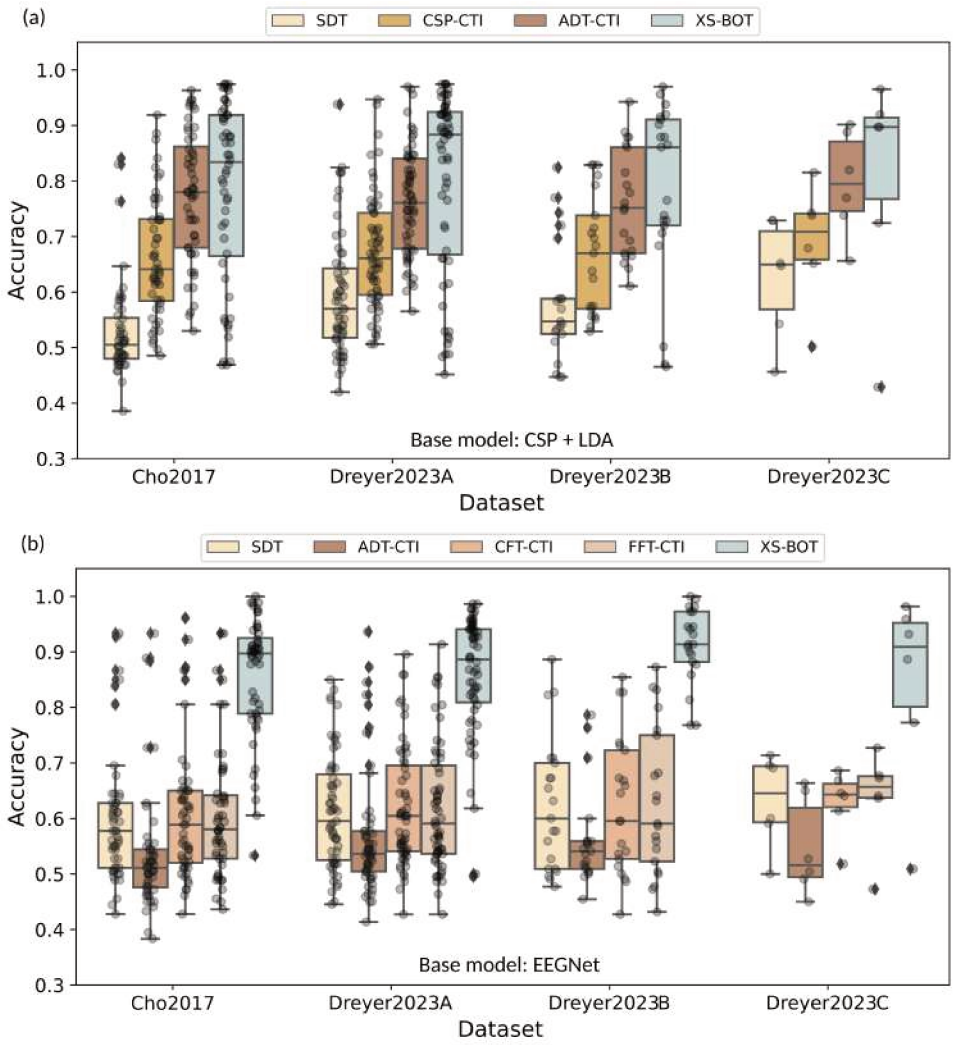

Figure 3: Comparison of XS-BOT against re-training CTI baselines in different MI-EEG datasets. Each point represents one test subject. (a) Results for CSP+LDA. (b) Results for EEGNet.

5.2 Impact of current-trial information usage: adaptation vs re-training

To further assess the performance of XS-BOT, we compared it against the same competing methods as before, but now including the cued label of the current testing trial during model training or fine-tuning, namely current-trial-included (CTI) variants. Although such approaches might not be viable for online scenarios, they can provide useful upper bound performance references.

In particular, for adaptation data training with inclusion of the current trial (ADT-CTI), a new decoding model is trained at each testing trial by means of stacking the adaptation data of the subject with the current trial data. In this scenario, both the feature extractor and the classifier incorporate task information from the current cued label during model optimization. For the DL model, CTI was additionally applied into the fine-tuning strategies. As before, in FFT-CTI the entire model is fine-tuned, whereas CFT-CTI restricts the fine-tuning to the classifier layers only. It is timely to note that these CTI approaches have an inherent advantage over XS-BOT: they directly adjust

classifier parameters using the cued label of the current trial. However, XS-BOT keeps the classifier fixed and uses the cued label only to guide feature transformations in label-based groups.

We also considered a variant (CSP-CTI), in which the information from the current trial is integrated in an intermediate step for the CSP+LDA pipeline. Here, the feature extractor is trained using both the adaptation data and the current trial, whereas the classifier is trained only on the adaptation data. In other words, while CSP is adjusted for every new trial, the LDA classification boundary is learned without direct access to the current trial. This setting allows a more aligned comparison with XS-BOT, where the current trial influences how CSP features are transformed but not the classifier itself.

Figure 3 (a) shows the results for the CSP+LDA pipeline. XS-BOT consistently outperforms CSP-CTI, suggesting that using the current trial's information to guide the adaptation process, as XS-BOT does, is more effective than including the current trial in the feature extractor training, as in CSP-CTI. Interestingly, although ADT-CTI retrains the whole model for each trial, XS-BOT achieves better performance. Figure 3 (b) shows that for EEGNet, unlike CSP+LDA, ADT-CTI performs worse than SDT. This could be explained by the fact that the size of the adaptation set combined with the current trial is not large enough to train a DL model from scratch. However, fine-tuning approaches such as CFT-CTI and FFT-CTI also fail to surpass SDT. Notably, XS-BOT significantly outperforms all these CTI approaches, highlighting its ability to effectively exploit current-trial information without retraining the model. Our approach not only achieves higher performance but also offers practical feasibility in a real-time BCI context. For EEGNet, CTI strategies require training or fine-tuning parts of the model for each trial, which is unfeasible in real-life scenarios due to the associated computational time. In contrast, XS-BOT avoids these time constraints, making it well suited for closed-loop MI-BCIs, where feedback must be provided immediately after the user performed the MI task in each trial. Note that average adaptation time per trial was under 40 ms for both CSP+LDA and EEGNet models when using a standard desktop computer (Intel i7, 16 GB RAM).

6 Discussions

In this work, we introduced XS-BOT, a domain adaptation framework for rehabilitative MI-BCIs that enables cross-subject and cross-dataset transfer learning with minimal adaptation data from the target subject. XS-BOT aligns target subject's features to match source subjects' features distribution using optimal transport. Our method allows adaptation on a trial-by-trial basis and uses the cued label of the current trial to guide the transport learning. Importantly, this is done without retraining the feature extractor or classifier itself and keeping a minimal computational cost. Our results demonstrate that XS-BOT significantly outperforms both classical and state-of-the-art adaptation strategies across multiple datasets and decoding models, achieving reliable performance even when using only three EEG channels.

XS-BOT consistently improves decoding accuracy over the nonadaptive baseline (SDT), both when using spatio-spectral features (CSP+LDA) and DL features (EEGNet). Performance is boosted in most subjects, exceeding the 0.7 accuracy threshold for effective BCI control. These improvements are especially marked in subjects with initially low accuracy, highlighting the method's ability to compensate for challenging decoding sce-

narios. In cases where no improvement is observed, XS-BOT does not introduce significant performance drops. These cases are systematically linked to subjects with low class distinctiveness and already low SDT performance, suggesting that the limiting factor lies in the fact that signals may not contain sufficient class-discriminative information to support learning an effective transportation plan. Together, these results emphasize that XS-BOT is not only effective but also safe to deploy, as it does not significantly degrade performance in any case and tends to offer the largest gains where adaptation is most needed.

Compared to other adaptation strategies, XS-BOT shows clear advantages across all datasets and decoding models. Unlike ADT, which relies solely on the small adaptation set from the target subject to train the model, XS-BOT leverages information from the source domain and current trial, achieving consistently higher accuracy. It also outperforms both unsupervised (EA and CMMN) and supervised (GR-CSP) alignment methods. Similarly, fine-tuning strategies (FFT and CFT) applied on EEGNet perform significantly worse, likely due to insufficient data for effective adaptation [45, 46]. In contrast, XS-BOT avoids model retraining and still achieves superior performance with the same amount of adaptation data. These results show that XS-BOT is a more robust and reliable alternative in the practically desirable situation of minimal amount of adaptation data from the new target subject. We also compared it against competitors with current trial-included (CTI) strategies. Although these approaches also have direct access to the trial to be decoded, they are not in any case more effective than XS-BOT. In addition, in the case of DL models, the high computational cost of retraining or fine-tuning makes these approaches infeasible in real-life implementations.

XS-BOT was designed to enhance the usability of MI-BCIs in rehabilitation settings by considering: i) short calibration, ii) non-delayed feedback delivery, and iii) fast electrode cap setup. A short calibration was not only addressed by cross-subject and cross-dataset transfer learning, but also by using a limited number of adaptation trials from the target subject. As in [10], only 20 trials are required, which in practical terms means no more than 5 minutes at the beginning of the session. Additionally, XS-BOT was built to allow fast adaptation, enabling rapid trial-by-trial feedback delivery, improving accuracy without burdening real-time data processing schemes. Lastly, XS-BOT achieves high-performance results only 3 channels, a minimal setup (15% the typical channel number consisting of 20 electrodes covering the sensorimotor cortex [12–14, 22]). These factors have the potential to make MI-BCIs for rehabilitation more practical, user-friendly and accessible. Simpler channels setups result in quicker preparation and simplified operation; fewer adaptation trials minimize user fatigue and accelerate the transition to real-time control, where the rehabilitation process takes place; and real-time feedback is crucial for effective therapy [4].

Although these results demonstrate the robustness and efficiency of XS-BOT, some limitations must be considered. First, our approach relies on the availability of cued labels during adaptation, which is possible in rehabilitation settings, but may not be the case for other BCI applications. Second, XS-BOT assumes that the data from the target subject meet two key conditions: (i) the adaptation subset is representative of the whole test distribution, and (ii) the user generates discriminable and stable MI patterns. These assumptions may not always be held, particularly in clinical stroke populations. Moreover, the applicability of XS-BOT to newer and emerging DL models remains an open avenue for exploration. Similarly, more investigation is needed to understand the impact of the number and location of EEG channels into model decoding performance. Another

promising direction for future work is to examine the impact of training with different groups of source subjects, such as selecting a small group of highly skilled users or identifying the minimum number of subjects required for effective pretraining. Finally, future studies could explore the feasibility of transferring models trained on healthy individuals to motor-impaired populations [23], enabling faster transitions to real-time MI-BCI control in therapeutic settings.

Taken together, these findings highlight XS-BOT as an impactful approach for rehabilitative MI-BCIs. By combining minimal calibration data, trial-by-trial adaptation, and a highly reduced electrode setup, XS-BOT has the potential to bring these systems closer to ultimate users, who will benefit from easy to use BCI systems in clinical scenarios.

7 Conclusion

Despite their potential for rehabilitation, the practical adoption of MI-BCIs is markedly hindered by lengthy subject-specific calibration phases before real-time use. This work addresses this challenge by introducing XS-BOT, a cross-subject and cross-dataset domain adaptation framework that enables robust MI decoding with minimal adaptation data from the new user. XS-BOT leverages large public datasets for model training and effectively utilizes cued labels to guide the adaptation at the feature level, avoiding retraining. The features of the target subject are aligned with the training distribution in a trial-by-trial manner, with minimal computational cost, enabling meaningful real-time feedback. XS-BOT demonstrates consistently superior performance than state-of-the-art approaches in a scenario with as few as 20 adaptation trials from the target user and using data from just 3 EEG channels. From the usability perspective, shorter calibration phases and minimal electrode setups not only simplify preparation and reduce user fatigue but also enable a faster transition to real-time control where rehabilitation effects take place.

In summary, XS-BOT offers a robust and practical approach for cross-subject adaptation in MI-BCIs, particularly in rehabilitation scenarios. It represents a meaningful step toward plug-and-play operation, addressing key barriers to real-world use, and advancing user-friendly MI-BCI systems.

References

[1] Simanto Saha and Mathias Baumert. “Intra-and inter-subject variability in EEG-based sensorimotor brain computer interface: a review”. In: *Frontiers in computational neuroscience* 13 (2020), p. 87.

[2] Bradley J Edelman et al. “Non-invasive brain-computer interfaces: state of the art and trends”. In: *IEEE Reviews in Biomedical Engineering* (2024).

[3] Fatemeh Fahimi et al. “Inter-subject transfer learning with an end-to-end deep convolutional neural network for EEG-based BCI”. In: *Journal of neural engineering* 16.2 (2019), p. 026007.

[4] Andrea Biasiucci et al. “Brain-actuated functional electrical stimulation elicits lasting arm motor recovery after stroke”. In: *Nature communications* 9.1 (2018), p. 2421.

[5] Ander Ramos-Murguialday et al. “Brain-machine interface in chronic stroke: randomized trial long-term follow-up”. In: *Neurorehabilitation and neural repair* 33.3 (2019), pp. 188–198.

[6] Alexander A Frolov et al. “Post-stroke rehabilitation training with a motor-imagery-based brain-computer interface (BCI)-controlled hand exoskeleton: a randomized controlled multicenter trial”. In: *Frontiers in neuroscience* 11 (2017), p. 400.

[7] Chaofei Fan et al. “Plug-and-play stability for intracortical brain-computer interfaces: a one-year demonstration of seamless brain-to-text communication”. In: *Advances in neural information processing systems* 36 (2023), pp. 42258–42270.

[8] Min-Ho Lee et al. “EEG dataset and OpenBMI toolbox for three BCI paradigms: An investigation into BCI illiteracy”. In: *GigaScience* 8.5 (2019), giz002.

[9] Pauline Dreyer et al. “A large EEG database with users’ profile information for motor imagery brain-computer interface research”. In: *Scientific Data* 10.1 (2023), p. 580.

[10] Victoria Peterson et al. “Transfer learning based on optimal transport for motor imagery brain-computer interfaces”. In: *IEEE Transactions on Biomedical Engineering* 69.2 (2021), pp. 807–817.

[11] Fabien Lotte et al. “A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update”. In: *Journal of neural engineering* 15.3 (2018), p. 031005.

[12] Benjamin Blankertz et al. “Optimizing spatial filters for robust EEG single-trial analysis”. In: *IEEE Signal processing magazine* 25.1 (2007), pp. 41–56.

[13] Vernon J Lawhern et al. “EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces”. In: *Journal of neural engineering* 15.5 (2018), p. 056013.

[14] Hamdi Altaheri et al. “Deep learning techniques for classification of electroencephalogram (EEG) motor imagery (MI) signals: A review”. In: *Neural Computing and Applications* 35.20 (2023), pp. 14681–14722.

[15] Martin Wimpff et al. “EEG motor imagery decoding: A framework for comparative analysis with channel attention mechanisms”. In: *Journal of neural engineering* 21.3 (2024), p. 036020.

[16] Ce Zhang et al. “EEG-inception: an accurate and robust end-to-end neural network for EEG-based motor imagery classification”. In: *Journal of Neural Engineering* 18.4 (2021), p. 046014.

[17] Mouad Riyad et al. “Incep-EEGNet: a convnet for motor imagery decoding”. In: *Image and Signal Processing: 9th International Conference, ICISP 2020, Marrakesh, Morocco, June 4–6, 2020, Proceedings 9*. Springer. 2020, pp. 103–111.

[18] Thorir Mar Ingolfsson et al. “EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain–machine interfaces”. In: *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2020, pp. 2958–2965.

[19] Shiu Kumar et al. “OPTICAL+: a frequency-based deep learning scheme for recognizing brain wave signals”. In: *Peerj Computer Science* 7 (2021), e375.

[20] Fangzhou Xu et al. “A transfer learning framework based on motor imagery rehabilitation for stroke”. In: *Scientific Reports* 11.1 (2021), p. 19783.

[21] Xin Deng et al. “Advanced TSGL-EEGNet for motor imagery EEG-based brain-computer interfaces”. In: *IEEE access* 9 (2021), pp. 25118–25130.

[22] Robin Tibor Schirrmeister et al. “Deep learning with convolutional neural networks for EEG decoding and visualization”. In: *Human brain mapping* 38.11 (2017), pp. 5391–5420.

[23] Aarthy Nagarajan et al. “Transferring a deep learning model from healthy subjects to stroke patients in a motor imagery brain–computer interface”. In: *Journal of Neural Engineering* 21.1 (2024), p. 016007.

[24] Kaishuo Zhang et al. “Adaptive transfer learning for EEG motor imagery classification with deep convolutional neural network”. In: *Neural Networks* 136 (2021), pp. 1–10.

[25] Qingguo Wei and Xinjie Ding. “Intra-and inter-subject common spatial pattern for reducing calibration effort in MI-based BCI”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 31 (2023), pp. 904–916.

[26] Siamac Fazli et al. “Subject-independent mental state classification in single trials”. In: *Neural networks* 22.9 (2009), pp. 1305–1312.

[27] Hyohyeong Kang et al. “Composite common spatial pattern for subject-to-subject transfer”. In: *IEEE Signal Processing Letters* 16.8 (2009), pp. 683–686.

[28] Haiping Lu et al. “Regularized common spatial pattern with aggregation for EEG classification in small-sample setting”. In: *IEEE transactions on Biomedical Engineering* 57.12 (2010), pp. 2936–2946.

[29] Fabien Lotte and Cuntai Guan. “Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms”. In: *IEEE Transactions on biomedical Engineering* 58.2 (2010), pp. 355–362.

[30] Wojciech Samek et al. “Robust spatial filtering with beta divergence”. In: *Advances in Neural Information Processing Systems* 26 (2013).

[31] Daniel Bartz and Klaus-Robert Müller. “Covariance shrinkage for autocorrelated data”. In: *Advances in neural information processing systems* 27 (2014).

[32] Benjamin Blankertz et al. “Invariant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing”. In: *Advances in neural information processing systems* 20 (2007).

[33] Satyam Kumar et al. “Transfer learning promotes acquisition of individual BCI skills”. In: *PNAS nexus* 3.2 (2024), pgae076.

[34] He He and Dongrui Wu. “Transfer learning for brain–computer interfaces: A Euclidean space data alignment approach”. In: *IEEE Transactions on Biomedical Engineering* 67.2 (2019), pp. 399–410.

[35] Bruna Junqueira et al. “A systematic evaluation of euclidean alignment with deep learning for EEG decoding”. In: *Journal of Neural Engineering* 21.3 (2024), p. 036038.

[36] Theo Gnassounou et al. “Convolution monge mapping normalization for learning on sleep data”. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 10457–10476.

[37] Apolline Mellot et al. “Harmonizing and aligning M/EEG datasets with covariance-based techniques to enhance predictive regression modeling”. In: *Imaging Neuroscience* 1 (2023), pp. 1–23.

[38] Sinno Jialin Pan and Qiang Yang. “A Survey on Transfer Learning”. In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), pp. 1345–1359. DOI: 10.1109/TKDE.2009.191.

[39] Victoria Peterson et al. “Towards subject-centered co-adaptive brain-computer interfaces based on backward optimal transport”. In: *Journal of Neural Engineering* (2025).

[40] Leonid V Kantorovich. “On the translocation of masses”. In: *Dokl. Akad. Nauk. USSR (NS)*. Vol. 37. 1942, pp. 199–201.

[41] Pierre Guetschel and Michael Tangermann. “Transfer Learning between Motor Imagery Datasets using Deep Learning–Validation of Framework and Comparison of Datasets”. In: *arXiv preprint arXiv:2311.16109* (2023).

[42] Fabien Lotte and Camille Jeunet. “Defining and quantifying users’ mental imagery-based BCI skills: a first step”. In: *Journal of neural engineering* 15.4 (2018), p. 046030.

[43] Milton Friedman. “The use of ranks to avoid the assumption of normality implicit in the analysis of variance”. In: *Journal of the american statistical association* 32.200 (1937), pp. 675–701.

[44] Peter Bjorn Nemenyi. *Distribution-free multiple comparisons.* Princeton University, 1963.

[45] German I Parisi et al. “Continual lifelong learning with neural networks: A review”. In: *Neural networks* 113 (2019), pp. 54–71.

[46] Anthony Robins. “Catastrophic forgetting, rehearsal and pseudorehearsal”. In: *Connection Science* 7.2 (1995), pp. 123–146.

Supplementary material to No need to re-train for new users: cross-subject domain adaptation in rehabilitative BCIs via backward optimal transport

C. M. Galván, D. H. Milone, R. D. Spies, and V. Peterson

1 Results for additional deep learning models

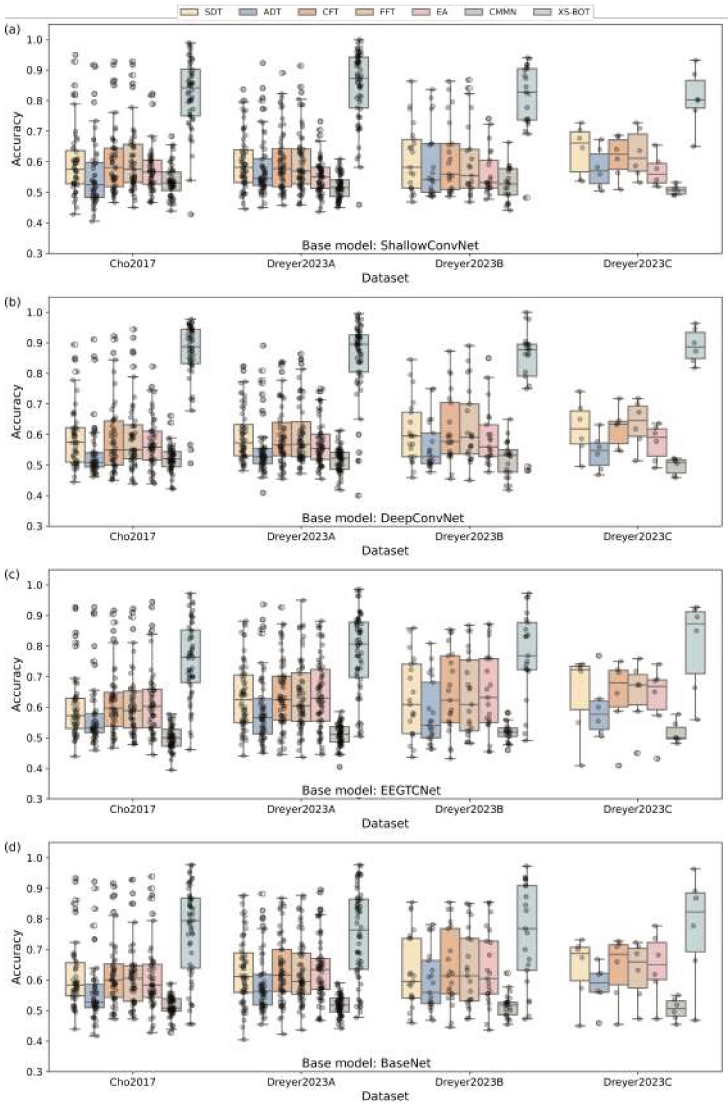

Figure A1: Performance comparison of adaptation methods using (a) ShallowConvNet, (b) DeepConvNet, (c) EEGTCNet and (d) BaseNet. In all the cases, XS-BOT outperforms all competing methods, including the nonadaptive baseline (SDT) and other adaptive strategies (ADT, CFT, FFT, EA and CMMN). These results resemble those presented with EEGNet in the main manuscript.

To further evaluate whether the performance gains of XS-BOT extend across different neural network architectures, we conducted additional experiments using four additional deep learning (DL) models: ShallowConvNet, DeepConvNet [1], EEGTCNet [2] and BaseNet [3]. All the models were trained and tested using the same experimental setup described in Section IV b of the main manuscript. The results are presented in Figure A1 for the different base models: (a) ShallowConvNet, (b) DeepConvNet, (c) EEGTCNet, and (d) BaseNet.

For ShallowConvNet, XS-BOT achieves an average accuracy of 0.830 $\pm$ 0.115 (mean $\pm$ std), substantially outperforming the nonadaptive baseline SDT, which reaches 0.602 $\pm$ 0.103. XS-BOT also significantly exceeds all competing adaptation methods: FFT

(0.606 ± 0.110), CFT (0.605 ± 0.107), ADT (0.575 ± 0.107), and EA (0.563 ± 0.069). The Friedman test rejects the null hypothesis of equal performance among methods ($p \ll 0.001$), and post-hoc Nemenyi tests confirm that XS-BOT performs significantly better than SDT and all other adaptive methods ($p \ll 0.001$ for all pairwise comparisons).

For DeepConvNet, XS-BOT reaches an average ±std accuracy of 0.854 ± 0.123, again greatly surpassing the nonadaptive baseline SDT at 0.600 ± 0.097. It also outperforms all other adaptation methods: FFT (0.606 ± 0.112), CFT (0.597 ± 0.109), EA (0.576 ± 0.082), and ADT (0.549 ± 0.089). The corresponding statistical tests confirm the significance of these differences ($p \ll 0.001$).

For EEGTCNet, XS-BOT achieves an average accuracy of 0.768 ± 0.137, clearly outperforming the nonadaptive baseline SDT (0.623 ± 0.114). It also surpasses all adaptive competitors: FFT (0.628 ± 0.117), CFT (0.635 ± 0.117), EA (0.635 ± 0.118), ADT (0.583 ± 0.109), and CMMN (0.508 ± 0.037). Statistical analysis confirms that these differences are highly significant, with the Friedman test rejecting the null hypothesis ($p \ll 0.001$) and post-hoc Nemenyi comparisons showing that XS-BOT consistently outperforms SDT and all other adaptation methods ($p \ll 0.001$).

For BaseNet, XS-BOT obtains an average accuracy of 0.756 ± 0.151, once again greatly exceeding the nonadaptive SDT baseline (0.624 ± 0.112). Competing adaptation methods also fall behind: FFT (0.625 ± 0.110), CFT (0.629 ± 0.111), EA (0.627 ± 0.112), ADT (0.580 ± 0.101), and CMMN (0.515 ± 0.035). As in the previous architectures, the Friedman test indicates significant differences across methods ($p \ll 0.001$), with Nemenyi post-hoc tests confirming the superiority of XS-BOT over both the nonadaptive baseline and all other adaptive approaches ($p \ll 0.001$).

These results extend and reinforce the findings of the main manuscript, demonstrating that XS-BOT consistently provides significant improvements over the nonadaptive baseline and all competing adaptive methods across different DL architectures.

2 Implementation details

2.1 MI-EEG preprocessing

Preprocessing was implemented using functions from MNE-Python [4], aiming to simulate an online processing scenario. The full preprocessing pipeline is available in the repository, along with the configuration files used.

The raw EEG data were first segmented into trials from 0 to 3 seconds after the cue onset. Then, the following steps were applied sequentially to each trials:

1. Re-referencing to the Fz electrode.
2. Bandpass filtering between 4 and 40 Hz using deafult IIR filter (4th order Butterworth).
3. Resampling to 128 Hz to standardize the sampling rate.
4. Cropping the trials to retain only the 0.5–2.5 s interval after cue onset, focusing on the time window most relevant to MI activity.

This pipeline is designed to mimic the constraints and characteristics of real-time BCI systems, where preprocessing must be efficient and performed independently for each trial, avoiding data leakage from future time points.

2.2 MI decoding models and training

This section describes the decoding models used in our experiments, including both classical and DL approaches. All implementations and configuration files used to run the experiments are provided in the associated repository.

- CSP+LDA: For the CSP model, we used the default implementation provided by MNE-Python [4]. Given that only three EEG channels were used in all experiments, the number of CSP components was set to 2. Features were computed as the mean ± sd power of each spatially filtered signal. For LDA, we employed the Scikit-learn implementation [5].
- DL models: Our Pytorch implementations were based on Braindecode [6]. For simplicity, sequential implementation was changed to forward-like implementation. The hyperparameters were set as the recommended in the original works. In the following we specify them for each model.
  - EEGNet [7]: The number of temporal filters ($F1$) was set to 8 and the number of spatial filters or depth multiplier (D) was 2. $F2$, the number of pointwise filters, was $F_1 \times D = 16$. Dropout probability was 0.25, as recommended for the cross-subject analysis. A temporal kernel of 64 samples was considered. Temporal pooling was implemented using mean ± sd pooling.
  - ShallowConvNet [1]: 40 temporal filters and 40 spatial filters were used. The length of the temporal filters was 25. The length of the temporal pooling filter was 75, while the stride between temporal pooling filters was 15. Batch nomalization with momentum of 0.1 was applied. The dropout probability was 0.5. Temporal pooling was implemented using mean ± sd pooling.
  - DeepConvNet [1]: 25 temporal filters and 25 spatial filters were used in the first block, with the length of the temporal filters set to be 10. The length of the temporal pooling filter was 3 while the stride between the temporal pooling filters was 3. The number of temporal filters in the second block was 50. Here, the pooling stride was changed to 2 instead of the original value of 3 to match the length of the input signal. In the third block, 100 temporal filters were considered. Again, here, the pooling stride was set to 2. 200 temporal filters were applied in the fourth block. In this block, the pooling stride was 1. Temporal pooling was implemented using max pooling in all cases. Batch nomalization with momentum of 0.1 was applied. The dropout probability was 0.5.

The three DL models were trained using the AdamW optimizer [8], with PyTorch's default parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 0.01). The learning rate was set to 1×10^{-3}, and the models were trained to minimize cross-entropy loss. A batch size of 64 was used, and to ensure balanced class distribution within each mini-batch, a stratified sampler was employed. Each model was trained for 500 epochs without a validation set, and the model from the final epoch was used for evaluation, following the protocol described in [9]. Training was performed using PyTorch with CUDA 11.8 on an NVIDIA RTX 3060 GPU. The training time on the full Lee2019_MI dataset [10] was approximately 3 min for EEGNet and ShallowConvNet and 4 min for Deep4Net.

3 Source subset and hyperparameter selection in XS-BOT

To select the optimal source subset together with its regularization parameters in XS-BOT, we follow a three-step process: (1) source subsets construction, (2) regularization hyperparameters optimization, and (3) final source subset selection. In the following, each step is described:

We use a unified adaptation criterion across all steps, which combines classification accuracy with a measure of confidence. For DL models, we define the confidence of a sample as the probability assigned to the predicted class. For LDA, confidence is defined as the distance to the decision boundary.

1. Construction of source subsets. We generate M source subsets from the full source domain (i.e., the entire training dataset). Each subset consists of N_v samples with balanced class distribution. To construct the first source subset, we proceed as follows: for each class, the sample with the highest confidence, as previously defined by the predicted class probability for deep learning models or the distance to the LDA decision boundary, is selected as the prototype. The corresponding class cluster is then formed by selecting the $N_v/2 - 1$ nearest neighbors of the prototype in feature space. This procedure is repeated M times, each time using the next most confident sample as prototype.

2. Selection of optimal hyperparameters. For each source subset, we perform a grid search over the regularization parameters λ and ν to identify the pair that yields the best adaptation performance. Specifically, for each parameter configuration, we learn a transport plan between the fixed adaptation set and the source subset. We then transform the adaptation features using the resulting transport map and predict their labels using the pretrained classifier. The optimal parameters for each source subset are those that maximize the combined adaptation criterion: classification accuracy and sum of confidence on the transformed adaptation samples.

3. Final source subset selection. Once the optimal hyperparameters have been determined for each source subset, we select the source subset (along with its associated parameters) that achieves the best performance according to the same adaptation criterion used in Step 2.

In practice, this selection process is performed at the beginning of the session, once the adaptation trials have been collected, and takes approximately 10 seconds for CSP+LDA and 1 minute for EEGNet on a standard computer (Intel i7 CPU, 16 GB RAM), without using GPU acceleration. However, these delays does not pose an issue for online processing, since once the optimal source subset and regularization parameters are selected before the start of real-time control of the system, they remain fixed throughout the adaptation.

4 Robustness of XS-BOT across training datasets

To evaluate the robustness of XS-BOT across different training datasets, we conducted an experiment in which each test dataset (Cho2017, Dreyer2023A, Dreyer2023B, and Dreyer2023C) was used, one at a time, for training while testing was performed on the remaining three datasets. The results for the base models CSP+LDA and EEGNet are presented in Table A1 and A2, respectively. As shown in these tables, XS-BOT again achieves markedly higher performance than SDT and the other competitors, consistent

with the findings obtained when Lee2019_MI was used for training.

Training in Cho2017 dataset						
Testing	**SDT**	**ADT**	**CSP-GR**	**EA**	**CMMN**	**XS-BOT**
Lee2019_MI	0.627 ± 0.108	0.595 ± 0.113	0.646 ± 0.119	0.631 ± 0.114	0.504 ± 0.036	**0.837** ± 0.140
Dreyer2023A	0.618 ± 0.114	0.594 ± 0.112	0.643 ± 0.117	0.632 ± 0.118	0.519 ± 0.044	**0.830** ± 0.162
Dreyer2023B	0.595 ± 0.096	0.594 ± 0.122	0.634 ± 0.122	0.626 ± 0.126	0.517 ± 0.031	**0.831** ± 0.151
Dreyer2023C	0.664 ± 0.107	0.602 ± 0.102	0.676 ± 0.122	0.646 ± 0.113	0.520 ± 0.060	**0.834** ± 0.217
mean ± std	0.620 ± 0.109	0.595 ± 0.112	0.644 ± 0.118	0.631 ± 0.116	0.513 ± 0.040	**0.833** ± 0.153
Training at Dreyer2023A						
Lee2019_MI	0.592 ± 0.095	0.595 ± 0.113	0.645 ± 0.116	0.627 ± 0.114	0.508 ± 0.046	**0.862** ± 0.151
Cho2017	0.556 ± 0.076	0.567 ± 0.097	0.597 ± 0.113	0.593 ± 0.110	0.516 ± 0.037	**0.774** ± 0.178
Dreyer2023B	0.568 ± 0.065	0.594 ± 0.122	0.631 ± 0.118	0.621 ± 0.125	0.528 ± 0.033	**0.870** ± 0.144
Dreyer2023C	0.597 ± 0.075	0.602 ± 0.102	0.670 ± 0.120	0.660 ± 0.105	0.517 ± 0.048	**0.830** ± 0.177
mean ± std	0.574 ± 0.084	0.584 ± 0.108	0.625 ± 0.117	0.614 ± 0.116	0.515 ± 0.041	**0.827** ± 0.166
Training at Dreyer2023B						
Testing	**SDT**	**ADT**	**CSP-GR**	**EA**	**CMMN**	**XS-BOT**
Lee2019_MI	0.554 ± 0.064	0.595 ± 0.113	0.649 ± 0.117	0.626 ± 0.118	0.514 ± 0.037	**0.806** ± 0.159
Cho2017	0.500 ± 0.000	0.567 ± 0.097	0.606 ± 0.112	0.593 ± 0.104	0.505 ± 0.045	**0.735** ± 0.179
Dreyer2023A	0.528 ± 0.049	0.594 ± 0.112	0.647 ± 0.117	0.632 ± 0.117	0.511 ± 0.049	**0.791** ± 0.168
Dreyer2023C	0.527 ± 0.045	0.602 ± 0.102	0.677 ± 0.118	0.645 ± 0.109	0.548 ± 0.026	**0.845** ± 0.052
mean ± std	0.528 ± 0.051	0.587 ± 0.107	0.636 ± 0.116	0.619 ± 0.113	0.511 ± 0.044	**0.781** ± 0.168
Training at Dreyer2023C						
Testing	**SDT**	**ADT**	**CSP-GR**	**EA**	**CMMN**	**XS-BOT**
Lee2019_MI	0.581 ± 0.065	0.595 ± 0.113	0.652 ± 0.116	0.624 ± 0.111	0.514 ± 0.038	**0.807** ± 0.168
Cho2017	0.504 ± 0.013	0.567 ± 0.097	0.603 ± 0.113	0.599 ± 0.109	0.507 ± 0.049	**0.771** ± 0.177
Dreyer2023A	0.570 ± 0.080	0.594 ± 0.112	0.650 ± 0.117	0.629 ± 0.116	0.511 ± 0.049	**0.811** ± 0.161
Dreyer2023B	0.576 ± 0.084	0.594 ± 0.122	0.644 ± 0.121	0.627 ± 0.126	0.538 ± 0.053	**0.837** ± 0.124
mean ± std	0.555 ± 0.071	0.587 ± 0.109	0.637 ± 0.117	0.619 ± 0.113	0.514 ± 0.047	**0.802** ± 0.164

Table A1: Results from training on different datasets using CSP+LDA as the base model.

Training at Cho2017							
Testing	**SDT**	**ADT**	**CFT**	**FFT**	**EA**	**CMMN**	**XS-BOT**
Lee2019_MI	0.613 ± 0.117	0.554 ± 0.109	0.622 ± 0.119	0.632 ± 0.124	0.608 ± 0.122	0.515 ± 0.039	**0.790** ± 0.157
Dreyer2023A	0.588 ± 0.089	0.571 ± 0.109	0.599 ± 0.099	0.607 ± 0.105	0.596 ± 0.092	0.510 ± 0.040	**0.768** ± 0.153
Dreyer2023B	0.595 ± 0.106	0.561 ± 0.089	0.617 ± 0.125	0.621 ± 0.132	0.607 ± 0.108	0.517 ± 0.040	**0.753** ± 0.186
Dreyer2023C	0.593 ± 0.084	0.558 ± 0.094	0.599 ± 0.080	0.613 ± 0.105	0.601 ± 0.074	0.508 ± 0.018	**0.803** ± 0.131
mean ± std	0.599 ± 0.103	0.562 ± 0.105	0.610 ± 0.110	0.619 ± 0.116	0.602 ± 0.105	0.513 ± 0.039	**0.775** ± 0.158
Training at Dreyer2023A							
Testing	**SDT**	**ADT**	**CFT**	**FFT**	**EA**	**CMMN**	**XS-BOT**
Lee2019_MI	0.624 ± 0.115	0.554 ± 0.109	0.621 ± 0.118	0.631 ± 0.121	0.625 ± 0.116	0.498 ± 0.038	**0.786** ± 0.177
Cho2017	0.585 ± 0.102	0.540 ± 0.106	0.590 ± 0.111	0.599 ± 0.116	0.595 ± 0.106	0.513 ± 0.041	**0.760** ± 0.176
Dreyer2023B	0.615 ± 0.111	0.561 ± 0.089	0.630 ± 0.123	0.632 ± 0.130	0.619 ± 0.107	0.515 ± 0.035	**0.783** ± 0.180
Dreyer2023C	0.605 ± 0.090	0.558 ± 0.094	0.627 ± 0.095	0.646 ± 0.098	0.619 ± 0.096	0.483 ± 0.047	**0.788** ± 0.161
mean ± std	0.606 ± 0.108	0.550 ± 0.104	0.611 ± 0.111	0.619 ± 0.116	0.612 ± 0.106	0.506 ± 0.040	**0.775** ± 0.175
Training at Dreyer2023B							
Testing	**SDT**	**ADT**	**CFT**	**FFT**	**EA**	**CMMN**	**XS-BOT**
Lee2019_MI	0.602 ± 0.102	0.554 ± 0.109	0.605 ± 0.111	0.620 ± 0.114	0.617 ± 0.104	0.514 ± 0.048	**0.779** ± 0.166
Cho2017	0.577 ± 0.095	0.540 ± 0.106	0.576 ± 0.108	0.584 ± 0.115	0.585 ± 0.115	0.498 ± 0.042	**0.805** ± 0.143
Dreyer2023A	0.586 ± 0.088	0.571 ± 0.109	0.588 ± 0.092	0.601 ± 0.102	0.601 ± 0.102	0.508 ± 0.033	**0.809** ± 0.152
Dreyer2023C	0.602 ± 0.096	0.558 ± 0.094	0.598 ± 0.093	0.587 ± 0.097	0.607 ± 0.098	0.542 ± 0.037	**0.839** ± 0.152
mean ± std	0.589 ± 0.096	0.556 ± 0.104	0.591 ± 0.101	0.598 ± 0.107	0.603 ± 0.106	0.508 ± 0.042	**0.799** ± 0.153
Training at Dreyer2023C							
Testing	**SDT**	**ADT**	**CFT**	**FFT**	**EA**	**CMMN**	**XS-BOT**
Lee2019_MI	0.599 ± 0.092	0.554 ± 0.110	0.599 ± 0.099	0.609 ± 0.110	0.587 ± 0.091	0.494 ± 0.037	**0.824** ± 0.122
Cho2017	0.559 ± 0.090	0.540 ± 0.106	0.557 ± 0.098	0.570 ± 0.110	0.559 ± 0.076	0.498 ± 0.040	**0.778** ± 0.160
Dreyer2023A	0.597 ± 0.090	0.571 ± 0.109	0.601 ± 0.103	0.606 ± 0.107	0.589 ± 0.093	0.500 ± 0.032	**0.845** ± 0.129
Dreyer2023B	0.584 ± 0.094	0.561 ± 0.089	0.586 ± 0.103	0.605 ± 0.122	0.585 ± 0.095	0.510 ± 0.033	**0.831** ± 0.137
mean ± std	0.585 ± 0.092	0.556 ± 0.106	0.586 ± 0.101	0.596 ± 0.111	0.580 ± 0.088	0.499 ± 0.036	**0.819** ± 0.139

Table A2: Results from training on different datasets using EEGNet as the base model.

References

[1] Robin Tibor Schirrmeister et al. “Deep learning with convolutional neural networks for EEG decoding and visualization”. In: *Human brain mapping* 38.11 (2017), pp. 5391–5420.

[2] Thorir Mar Ingolfsson et al. “EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain–machine interfaces”. In: *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2020, pp. 2958–2965.

[3] Martin Wimpff et al. “EEG motor imagery decoding: A framework for comparative analysis with channel attention mechanisms”. In: *Journal of neural engineering* 21.3 (2024), p. 036020.

[4] Alexandre Gramfort et al. “MEG and EEG Data Analysis with MNE-Python”. In: *Frontiers in Neuroscience* 7.267 (2013), pp. 1–13. DOI: 10.3389/fnins.2013.00267.

[5] Fabian Pedregosa et al. “Scikit-learn: Machine learning in Python”. In: *the Journal of machine Learning research* 12 (2011), pp. 2825–2830.

[6] Robin Tibor Schirrmeister et al. “Deep learning with convolutional neural networks for EEG decoding and visualization”. In: *Human Brain Mapping* (Aug. 2017). ISSN: 1097-0193. DOI: 10.1002/hbm.23730. URL: http://dx.doi.org/10.1002/hbm.23730.

[7] Vernon J Lawhern et al. “EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces”. In: *Journal of neural engineering* 15.5 (2018), p. 056013.

[8] Ilya Loshchilov and Frank Hutter. “Decoupled weight decay regularization”. In: *arXiv preprint arXiv:1711.05101* (2017).

[9] Aarthy Nagarajan et al. “Transferring a deep learning model from healthy subjects to stroke patients in a motor imagery brain–computer interface”. In: *Journal of Neural Engineering* 21.1 (2024), p. 016007.

[10] Min-Ho Lee et al. “EEG dataset and OpenBMI toolbox for three BCI paradigms: An investigation into BCI illiteracy”. In: *GigaScience* 8.5 (2019), giz002.

Anexo C

Towards subject-centered co-adaptive brain-computer interfaces based on backward optimal transport

Towards subject-centered co-adaptive brain-computer interfaces based on backward optimal transport

Victoria Peterson[1,2*], Valeria Spagnolo[1], Catalina Galván[1,2], Nicolás Nieto[1,3], Rubén D. Spies[1,2], Diego H. Milone[3]

[1]Instituto de Matemática Aplicada del Litoral, IMAL, UNL, CONICET, Santa Fe, Argentina.
[2]Departamento de Matemática, Facultad de Ingeniría Química, UNL, Santa Fe, Argentina.
[3]Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional, sinc(i), UNL, CONICET, FICH, Santa Fe, Argentina.
e-mail: vpeterson@santafe-conicet.gov.ar

Abstract

Objective. Controlling a motor imagery brain-computer interface (MI-BCI) can be challenging, requiring several sessions of practice. Electroencephalography (EEG)-based BCIs are particularly affected by cross-session variability. In this scenario, it is crucial to implement co-adaptive systems, where the machine adapts the decoding algorithm while the user learns how to control the BCI. To support the user learning process, it is essential to measure and provide real-time feedback on self-modulation skills. This study aims to develop a method for online assessment of MI modulation capability to build co-adaptive BCIs that improve both user performance and system accuracy. *Approach.* Backward optimal transport for domain adaptation allows across-session MI-BCI usage without classifier retraining. Using the cued label to guide the adaptation, a supportive backward adaptation (SBA) method is defined. The required model effort to perform a trial adaptation is proposed as an online metric of MI modulation skills. We conducted experiments on both real and simulated data to demonstrate that this metric effectively informs about the the discriminability and stability of the EEG patterns related to the MI task. The proposed metric is validated by means of Riemannian distinctiveness metrics. *Main Results.* Our findings show that the associated effort when applying SBA provides a meaningful way of evaluating EEG patterns discriminability, being significantly correlated with Riemannian distinctiveness metrics. *Significance.* This study introduces a novel framework for co-adaptive BCI learning that performs data adaptation while assessing the MI-BCI skills of the user. The proposed SBA approach can enhance BCI performance by facilitating session-to-session adaptation and empowering users with

valuable feedback based on their current MI modulation strategy. This framework represents a significant advancement in developing user-centered, co-adaptive MI-BCIs that effectively support and enhance user capabilities.

Keywords

Domain adaptation, co-adaptive BCI, BCI skills, user-centered BCI, Optimal Transport.

1 Introduction

Brain-computer interfaces (BCIs) provide a digital link between a brain and a device. In particular, BCIs based on the mental imagination of movements, i.e. motor imagery (MI), strongly rely on the subject's ability to self-regulate their brain activity [1]. In this context, machine learning (ML) models built upon brain-recorded data should be able to identify both the optimal feature space and the classifier decision boundary to effectively decode the brain activity related to the intended mental task [2]. While the acquisition of MI-BCI control skills requires practice over several sessions of use [3], the ML decoding algorithm should adapt and support the subject's learning process [4]. This simultaneous subject-machine adaptation defines what is known as co-adaptive learning [5].

From the ML perspective, co-adaptive learning involves adjusting the parameters of the decoding algorithm to address changes in data distribution between training and testing datasets. In BCI, training and testing datasets may be referred to as calibration and application sessions, which typically are registered at different days. Several aspects may influence the stability between sessions of MI-BCI. Psychological and physiological factors, such as level of fatigue, arousal and workload can influence the user capability to control the BCI [2, 6, 7]. Moreover, due to potential electrode-cap misalignment and electrode impedance values drift between sessions [8], non-invasive MI-BCI systems based on surface electroencephalography (EEG) are particularly susceptible to data distribution changes. Thus, from the ML decoding algorithm viewpoint, MI-BCIs based on EEG define a non-stationary learning environment [9].

Domain adaptation provides a suitable framework to address changes in observable non-stationary data [10]. In the context of BCIs, the variability across sessions can be managed by "adapting" the feature distribution from one domain (calibration session) to another domain (application session). Recently, we presented a backward formulation of optimal transport for domain adaptation (BOTDA) [11]. This method transforms data distribution of an application session to match the calibration data distribution, avoiding the traditional need for classifier retraining. Experiments mimicking online adaptation in MI-BCI for motor rehabilitation applications, showed that the use of a supervised BOTDA implementation significantly improved decoding performance of an already trained classifier, as well as state-of-the-art data alignment methods, including complete model retraining [11]. Interestingly, for some subjects, decoding performance resulted in similar fashion to a model without any adaptation. These results open a unique opportunity to study the relationship between adaptation success and the user capability to control MI-BCIs [12].

However, unveiling MI-BCI user skills is still an important challenge in the field. Efforts have been made in order to anticipate users' cognitive skills to self-regulate sensorimotor rhythms by using neurophysiological markers [1, 13], studying visual rotation

capabilities [14] as well as other cognitive and personality profiles tests ([14–16]). Nevertheless, these metrics may not reflect the changes related to the user learning process. Performance metrics from fixed (non-adaptive) ML classifiers can be calculated across sessions; however, they fail to capture the user's MI-BCI control skills. In this direction, the authors in [4] proposed new metrics to measure MI-BCI skills based on covariance matrices in the Riemannian space. These metrics were designed to be independent of the given classifier, and thus only reflect the MI patterns in the EEG data. Among them, class distinctiveness was proposed to measure how distinguishable within a session MI-EEG patterns are from each other, showing to be correlated with the spatial visual abilities of the participants, the mayor predictor of MI-BCI skills [17]. Nevertheless, user's MI-BCI skills should be measured online to provide appropriate feedback that would help them in better self-modulate their brain patterns.

While in MI-BCIs the classifier's performance depends on both the user's capability to generate distinct and consistent EEG patterns and the optimized ML model parameters [18], most current closed-loop BCIs offer feedback to the user based solely on the model output. Nonetheless, the classifier's effectiveness will only reflect the user's ability to operate the BCI if the model is reliable, that is, its complexity, number of trainable parameters, training strategy and used data are appropriate for the classification task. For building co-adaptive closed-loop BCIs, not only the method should adapt to the user as they learn how to control the BCI, but also the delivered feedback should be based on their generated EEG patterns. As such, feedback is expected to be informative about users' current MI skill levels, helping them develop better BCI control capabilities [12]. In this regard, in co-adaptive BCIs we argue that the algorithmic support, that is, the effort the model needs to make to perform the adaptation, should be measured and delivered as feedback to the BCI user to inform them in real-time about the quality of their EEG patterns. In this context, let us consider the following example, in which the performance of two users in MI-BCI utilizing a co-adaptive algorithm is compared. For a "high-performing" user, the co-adaptive method achieved a final accuracy of 90%, starting from a baseline accuracy of 80% using a basic (no adaptive) ML model. In contrast, for a "lower-performing" user, the same final accuracy of 90% was reached, but the initial accuracy of the basic ML model was only 60%. In the latter case, the effort in performing the adaptation was higher to that of the high-performing user. Thus, we argue that by measuring the adaptation cost, real-time feedback could be delivered to the users to help them improve their MI modulation strategies effectively.

In the present work, we show that our supervised backward domain alignment method constitutes a reliable online adaptive framework. In particular, we demonstrate that the effort required to transform a given trial so as to match the distribution of the calibration data offers a novel way for online assessment of the discriminability and stability of EEG patterns. We shall refer to this method as supportive backward adaptation (SBA). SBA not only facilitates successful adaptation between sessions in MI-BCIs, but it also enables the online evaluation of the user's ability to produce distinguishable and stable EEG patterns. This opens a new venue in co-adaptive BCI learning, where meaningful feedback based on adaptation-related information can be provided to the users, guiding them throughout the BCI learning process. The present study represents a step forward in developing user-centered co-adaptive closed-loop MI-BCIs.

2 Co-adaptive learning based on backward optimal transport

This section provides a brief review of backward optimal transport for domain adaptation (BOTDA) between sessions of BCI. Implementation details for supervised transfer learning are also given. The new metric for MI-BCI skills assessment based on the associated effort in performing the data adaptation is then presented. The supervised BOTDA implementation together with the MI-BCI skills assessment metric based on the adaptive cost constitute SBA.

2.1 Backward supervised online adaptation method

Backward adaptation consists of coping with changes in data distribution between two BCI sessions, while avoiding classifier retraining [11]. In this context, it is said that data used to train the classifier belongs to the source domain (Ω_s), while the application or testing data are drawn from the target domain (Ω_t). The goal of BOTDA is to adapt or transform features of the target domain so that the distribution become similar to the source domain feature distribution, i.e., where the classifier was trained. Assuming that the feature distribution drift is given by a certain transformation $\mathcal{B} : \Omega_t \to \Omega_s$, BOTDA consists of estimating $\mathcal{B}$ in a cost-effective manner [19].

BOTDA is applied at the feature space level, that is, after features are extracted. Regardless of the method used for learning the feature representation, BOTDA seeks to match the distribution of two datasets, the source ($\mathcal{S}$) and the target ($\mathcal{T}$) datasets. Mathematically, these two datasets can be described as a set of pair data:

$\mathrm{S} \doteq \{(\mathbf{x}_i^s, y_i^s), i = 1, \ldots, N_s\} \subset \Omega_s \times \mathcal{K}$,
$\mathcal{T} \doteq \{(\mathbf{x}_i^t, y_i^t), i = 1, \ldots, N_t\} \subset \Omega_t \times \mathcal{K}$, where N_s and N_t denote the number of trials in the source and target domain, respectively, $\mathbf{x}_i^s \in \Omega_s, \mathbf{x}_i^t \in \Omega_t$ are the feature vectors, with Ω_s, $\Omega_t \subset R^d$, $\mathcal{K} \doteq \{k_j\}_{j=1}^K$ and $y_i^s, y_i^t \in \mathcal{K}$ represent the indicated mental tasks belonging to K possible classes.

When working with discrete variables, estimating $\mathcal{B}$ results in finding the optimal transportation plan γ^* to minimize a certain cost function m. Then, defining $M = (m_{i,j})$, $m_{i,j} \geq 0$, we have:

$$\gamma^* \doteq {}_{\gamma}\langle \gamma, M \rangle_F, \tag{1}$$

where $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product.

Note that M is a matrix such that $m_{i,j} = m(\mathbf{x}_i^t, \mathbf{x}_j^s)$ represents the cost of moving a unit probability mass from $\mathbf{x}_i^t \in \Omega_t$ to $\mathbf{x}_j^s \in \Omega_s$. Generally, m is chosen as the square of the Euclidean distance [20].

To relax the optimization procedure and to include label information during the transportation plan estimation, two regularization terms $W_e(\cdot)$ and $W_c(\cdot)$ are added to the functional in (1), as follows:

$$\gamma^* \doteq {}_{\gamma}\langle \gamma, M \rangle_F + \lambda W_e(\gamma) + \nu W_c(\gamma), \tag{2}$$

where $\lambda > 0$ and $\nu > 0$ are regularization parameters that weight the contribution of each penalization term. Here $W_e(\gamma) \doteq \sum_{ij} \gamma_{i,j} \log(\gamma_{i,j})$ accounts for the negentropy, while $W_c(\gamma) \doteq \sum_j \sum_k \|\gamma_{\mathcal{I}_k,j}\|_2$, where $\mathcal{I}_k$ denotes the set of indices that belongs to class $k \in \mathcal{K}$, promotes transporting "together" samples with the same label. The problem defined by (2) is known as the Sinkhorn group-LASSO optimal transport problem.

The optimal transport plan γ^*, estimated by means of (2), is a matrix of the same size as the cost matrix M, which describes how to distribute the mass probability between the domains. In particular, in the backward formulation, each element $\gamma^*_{i,j}$ indicates how much probability mass from the target sample $\mathbf{x}^t_i$ is transferred to the source sample $\mathbf{x}^s_j$. Thus, in the context of online adaptation, and noting that the last two terms in (2) are independent of $\mathbf{x}$, for a given ℓ^{th} trial from the application session, transportation is the result of the following mapping:

$$\hat{\mathbf{x}}^t_\ell = \mathcal{B}_{\gamma^*}(\mathbf{x}^t_\ell) \doteq_{\mathbf{x}\in R^d} \sum_{j=1}^{N_s} \gamma^*_{\ell,j}\, m(\mathbf{x}, \mathbf{x}^s_j). \tag{3}$$

When the ℓ^{th} target sample is used during the optimal transport plan estimation, the transformation defined in (3) results in a simple weighted barycenter mapping. That is, $\hat{\mathbf{x}}^t_\ell = \sum_{j=1}^{N_s} \gamma^*_{\ell,j}\, \mathbf{x}^s_j$.

Considering real-life BCI applications, the changes between the source and the target domains must be estimated by using only a portion of data (as small as possible) of the current application session. In most BCIs setups, a few trials at the beginning of the session are registered without feedback; this is usually referred to as recalibration phase. The resulting recalibration data is used to learn the transportation plan from the target to the source domain. Therefore, in the sequel we shall refer to the recalibration data as the *transportation set*, and will be denoted by $\mathcal{V} \doteq \{(\mathbf{x}^v_i, y^v_i)\}_{i=1}^{N_v} \subset \mathcal{T}$, where $N_v \ll N_t$. Online adaptation based on BOTDA was proposed by adding the current ℓ^{th} trial to the transportation set, that is:

$$\mathcal{V}_\ell \doteq \left\{(\mathbf{x}^t_i, y^t_i)\right\}_{i=1}^{N_v} \cup \left\{(\mathbf{x}^t_\ell, y^t_\ell)\right\} \subset \mathcal{T}, \quad \forall \ell = N_v + 1, \ldots, N_t. \tag{4}$$

From a transfer-learning viewpoint, finding the minimizer in (2) for a transportation set as defined in (4) constitutes a supervised domain adaptation problem [10]. In the context of the present study, we shall refer to this supervised BOTDA implementation, as supportive backward adaptation (SBA). It is timely to note that the class label of the current trial, as reported by the BCI system, does not necessarily reflect the mental task performed by the user. Therefore, accounting for the trial cued label does not necessarily imply that after adaptation such a trial will be correctly classified, as we argue and demonstrate next.

2.2 Supportive backward adaptation to guide the user learning process

As previously explained, SBA performs supervised domain adaptation at the feature space level making use of the trial label to estimate the optimal transport plan. Nevertheless, in synchronous MI-BCIs, the "label" represents the instruction (cue), rather than the actual class label of the user-generated EEG patterns, which may or may not align with the instructed task.

Based on the hypothesis that SBA can successfully perform the adaptation when the user-generated EEG patterns belong to the cued mental task, it is reasonable to think that the associated cost of transporting a trial from the current application session to the calibration one, contain valuable information about the stability and discriminability of the EEG trial *per-se*. In fact, let us closely inspect (2). When the optimal plan

$\gamma^* \in R_+^{N_v+1 \times N_s}$ is computed for $\mathcal{V}_\ell$ as defined in (4), the fidelity term of the optimal transport functional in (2), known as the Wasserstein distance, reads as follows:

$$W = \sum_{i,j} \gamma^*_{i,j} m_{i,j}, \tag{5}$$

where $\gamma^*_{i,j}$ are the components of γ^*. Thus, every sample i in the target domain is transported as indicated by the optimal plan $\gamma^*_{i,j}$ given a cost $m_{i,j}$ to the j sample in the source domain. In other words, W quantify the amount of "work" done in order to make the sample distribution of the target domain resemble that of the source domain. Considering the way the transportation plan is learned (see (4)), we propose to quantify the algorithmic support in adapting the current ℓ^{th} trial by h, defined by:

$$h \doteq \sum_j \gamma^*_{\ell,j} m_{\ell,j} =< \gamma^*_\ell, \mathbf{m}_\ell >, \tag{6}$$

where γ^*_ℓ and $\mathbf{m}_\ell$ are the ℓ^{th} rows of the optimal transportation plan γ^* and of the cost matrix M, respectively.

By means of h, we are able to measure the effort exerted by SBA when performing the data distribution matching of a given trial. We shall refer to this measure as *algorithmic support*. It is expected that when the EEG patterns generated by the user are aligned with the indicated mental class, the class label information provided by the system helps the adaptation to be successful. That is, the class of the current trial after adaptation is correctly predicted by the classifier. In this context, it is expected that h be higher for less distinguishable and stable EEG patterns, resulting in more effort made by the model during the online adaptation process. The proposed algorithmic support index works as an online user skills metric which does not require of extra methods or layers of analysis upon the adaptive algorithm itself. Since it is applied at the feature space level, it can be used within any decoding pipeline, including end-to-end deep learning architectures.

3 Leveraging SBA for cross-session MI-BCI

In the present work we focus on a two-class subject-specific MI-BCI system used across sessions. The common spatial pattern (CSP) in conjunction with the linear discriminant analysis (LDA) are used to build the decoding algorithm [21]. As commonly done, the feature extractor as well as the discriminative model are trained following an intra-subject scheme using data from the calibration session (source domain). Data from a new BCI session, where the model is applied, defines the target domain. The subset of the first N_v trials of the target domain defines the recalibration set [11].

The initial step of SBA involves training both the feature extractor and the classifier using the complete calibration session data (training data). Then, both optimized models will always be kept fixed. To maintain the sample proportion between the source and the target domain when learning the optimal transport plan, a subset of the source data equal to N_v is first selected [22]. This subset of source data constitutes the source domain samples from where to learn the transportation plan. We have previously shown that with $N_v = 20$ is sufficient to perform the across sessions adaptation [11].

In order to find the training trials that best represent the calibration session, we sample multiple subsets of size N_v from the training data. For each subset, several backward transport plans are learned using combinations of regularization parameters, with the

recalibration data serving as target samples. For a given subset, the combination of regularization parameters that achieves the highest accuracy on the transported recalibration data is selected as the optimal. Then, the subset that leads to the highest accuracy on the recalibration data is chosen as the source subset. Thus, both the source subset and the regularization parameters are determined in a data-driven manner based on their performance with respect to the recalibration data [11]. For further implementation detail, please visit our GitHub repository (`https://github.com/vpeterson/otda-mibci`).

Once the optimum source subset and regularization parameters are chosen, online adaptation can be performed. That is, for every ℓ^{th} trial in the application session, the backward optimal transport plan is computed by adding the current ℓ^{th} trial to the transportation set (see (4)), and solving the backward OT problem as defined in (2) with the source subset serving as source samples and the transportation set $\mathcal{V}_\ell$ as target samples.

Figure 1 illustrates the adaptive process performed by SBA in a two-dimensional CSP space. While the full training set was used to learn the CSP features as well as the LDA decision boundary (Figure 1a), only the source subset is used to learn the transportation plan, as illustrated in Figure 1b. Testing trials without adaptation lie on the feature space as defined by the given discriminative patterns (Figure 1c). When SBA is applied, testing trials are transported towards one or the other class center of mass (Figure 1d). The algorithmic support for a set of trials is shown in Figure 1e. It is timely to note that the distance of a given trial from its corresponding class center of mass in the feature space does not necessarily correlate with the effort exerted by the model when performing the adaptation. Trials with features lying in close proximity to their corresponding class center of mass can get high algorithmic support values (e.g. trial 2).

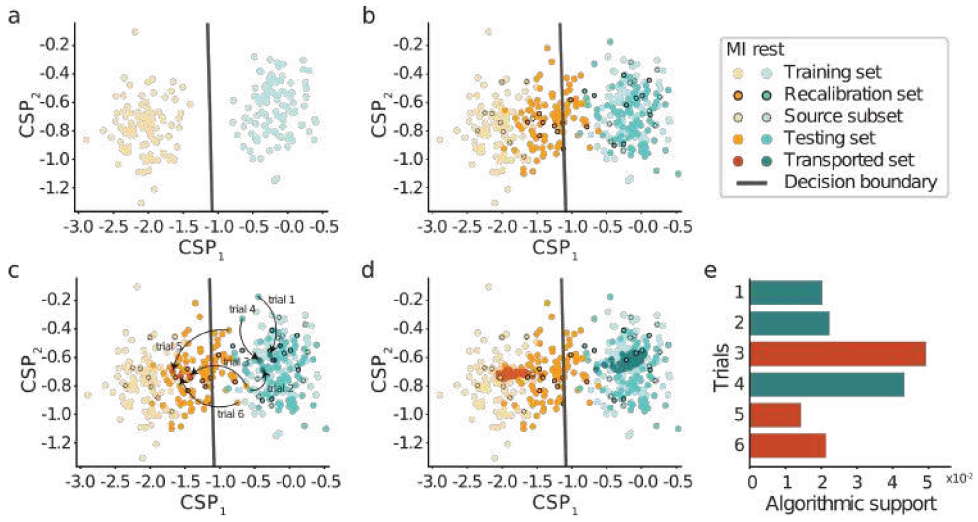

Figure 1: Schematic representation of SBA steps in the CSP feature space. **a.** Training data is used to learn the CSP+LDA decoding algorithm. **b.** By means of the recalibration data the optimal source subset is chosen from where to learn the backward transport plan. **c.** Example of few testing trials transported as indicated by the backward transport plan. Here arrows indicate the direction of the transportation. **d.** Comparison of the data with and without SBA application. **e.** Algorithmic support values for exemplary trials. Bar colors indicate the trial class. The higher the value, the stronger was the adaptive support performed by the algorithm.

4 Riemannian MI-BCI skills metrics

The Riemannian geometry has been widely adopted in the MI-BCI community to find a separable space in the covariance matrices manifold [23–25]. Given a raw EEG trial band-pass filtered in a frequency band of interest (e.g. 8 - 30 Hz), it is widely accepted that its spatial covariance matrix represents the band power of each channel in their first diagonal while the inter-channel covariance values are on the off-diagonal elements [24]. Based on the Riemannian distance between two covariance matrices, the distinctiveness metric D was proposed for measuring how apart patterns belonging to two different states A and B are [4]. That is,

$$D = \frac{\delta_R(\bar{C}_A, \bar{C}_B)}{\frac{1}{2}(\sigma_{C_A} + \sigma_{C_B})}, \tag{7}$$

where δ_R is the Riemannian distance, $\bar{C}$ is the Riemannian mean covariance matrix [24] and σ_C represents the mean absolute deviation of the covariance matrices, calculated as a measure of dispersion of the covariance matrices with respect to their mean. When comparing two MI classes, D has been shown to be positively correlated with the user's capability to visualize objects and understands their spatial relationship (a.k.a spatial ability). Spatial ability is known to be a major predictor of MI-BCI performance [14, 26], and thus, it is expected that the higher the distinctiveness value is, the more distinguishable the user-generated EEG patterns are, informing about the MI-BCI user skills.

In the present study, D is used as a validated tool to assess users' BCI skills, as done by others [27–29]. Note that when used within a session between classes, it is known as *class distinctiveness* [4]. When the states A and B belong to the same class but across different sessions, D defines a between-session user MI-BCI skills metric, to which we shall refer to as *session distinctiveness*. Similarly, but when computed trial-wise, it can measure how dissimilar a given ℓ^{th} trial for a given k^{th} class in the application session is to the its corresponding training peers of the calibration session. That is, $D_\ell = \delta_R(\bar{C}^{tr}_k, C^{te}_{k,\ell})$, where $\bar{C}^{tr}_k$ denotes the Riemannian mean covariance matrix calculated on the training data for the class k, and $C^{te}_{k,\ell}$ denotes the covariance matrix of the ℓ^{th} trial belonging to class k to be decoded during the application session. We shall refer to D_ℓ computed trial-by-trial as *trial distinctiveness*. Note that, the higher the distinctiveness values are, the more dissimilar the compared states are.

5 Data

Simulated and real EEG data were used in this study, comprising two-classes MI-BCI datasets with at least two recordings sessions.

5.1 Simulated MI-EEG data

Different sessions of hand MI vs. rest EEG data were generated with PySimMIBCI [30]. Each session was simulated to define a possible MI-BCI scenario from the user's self-regulation capability perspective. While rest was simulated as idle EEG α-band activity in both hemispheres, for MI the simulation considered the well-known Event-Related Desynchronization (ERD) in the α-band at the contralateral hand motor area. Changes in the percentage of the ERD (ERD%) represents the relative change with respect to

the idle state of the α-band during MI. The higher the ERD% is, the better the MI modulation capabilities are.

Each session comprised 200 trials of 4 seconds length with balanced class distribution. The EEG data was simulated with a sampling frequency of 1000 Hz from 41 electrodes localized according to the 10-5 electrode system. Data was then downsampled at 250 Hz and each trial was epoched from 0.5 to 2.5 s.

To study the extent at which the adaptation could be successful, we simulated data as coming from the same participant but in different possible scenarios with respect to both the self-regulation subject capability and the MI-BCI accuracy. While the former was simulated by changing the ERD%, the latter was simulated by including a percentage of trials where the participant failed to perform the task indicated by the system (fail%). In the following, we shall refer as a simulated session with a given percentage of ERD and failed trials as $S_{ERD\%,fail\%}$. Simulated data sessions can be found in Zenodo (`https://zenodo.org/records/13760210`).

Simulating different self-regulation capabilities. We simulated sessions with different ERD%, starting from 50%, up to 10% in steps of 5%, leading to 9 MI-BCI sessions ($S_{50,0}$-$S_{10,0}$) from high to low α-band self-regulation capability. For each session, the ERD% was kept fixed and equal to the given value across trials. Figure 2a shows how the α peak decreases as the ERD% increases. Lower ERD% values have peak amplitudes closer to the rest condition, resulting in less distinguishable classes. The distinctiveness metric D was calculated intra-session between-classes and inter-sessions within-class. For the latter, session $S_{50,0}$ was chosen as the comparison reference. As shown in Figure 2b, intra-session distinctiveness decreases when the ERD% decreases, confirming that the lower the ERD% value is the harder the session results for the ML viewpoint. Similarly, conforming the ERD% decreases, the intra-session for MI class increases, showing how different lower ERD% values are from the strongest one (ERD% = 50.) The spatial information also reveals that low ERD% values present a less discriminable topography map, as shown in Figure 2c.

Simulating wrong performed trials. New sessions were created with varying numbers of failed trials (fail%). Specifically, we adjusted the percentage of failed trials from 10% to 50%, in step of 10% by swapping the labels between classes in a simulated session at 45% of ERD. This procedure resulted in five new sessions, named from $S_{45,10}$ to $S_{45,50}$.

Simulating MI related and non-related brain oscillations. We created a new data session of MI vs. rest of 100 trial each by assigning to the MI trials a uniform distribution on the ERD% from 10 to 50. In this way, simulated trials comprised different MI modulations levels within the same BCI session. In addition, following the approach used in [30], we simulated mental fatigue effects by increasing frontal θ and parietal α powers. Considering that mental fatigue typically manifest towards the end of the session, the first half of the simulated trials were fatigue-free, while the second half gradually incorporated increasing fatigue levels, by means of linearly augmenting the amplitude in frontal θ and parietal α power over trials. To show the fatigue effect impact besides MI-BCI skills user capability, the MI trials were simulated as having ERD% equal to 50 with no failed trials.

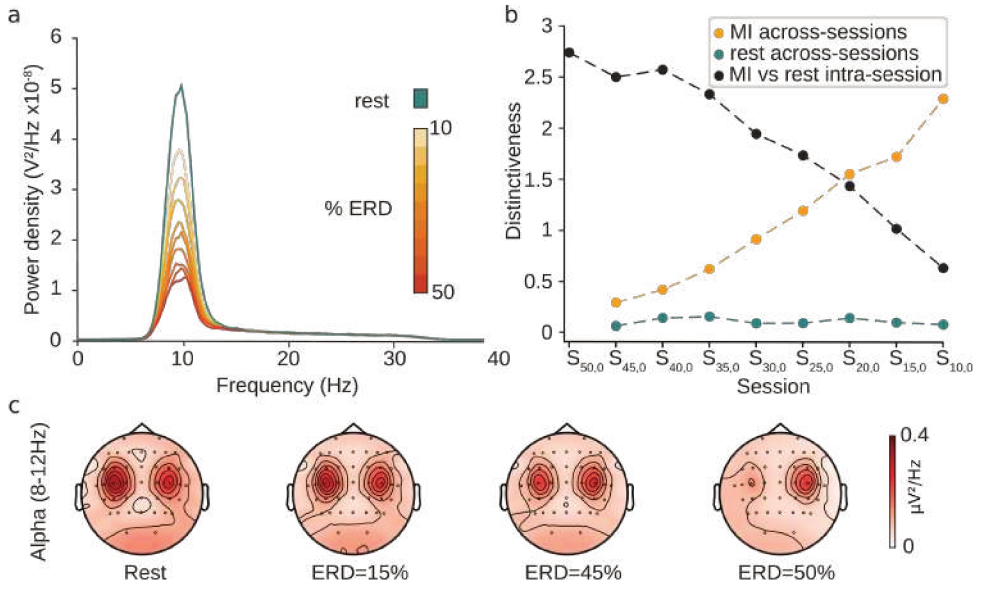

Figure 2: Simulated data main characteristics. **a.** MI self-regulation capabilities were simulated as different amplitude decrements of the alpha peak (10 Hz) with respect to baseline (rest). The higher the ERD% the higher the difference between rest and MI amplitudes. **b.** The distinctiveness metric is used to measure intra-session between classes and inter-session within class similarity. Session $S_{50,0}$ is used as reference for the session distinctiveness. Here dashed lines are used to connect adjacent dots, illustrating their tendency. **c.** Illustration of the alpha power in the channel-space for rest condition as well as for different ERD% values. The higher the ERD% value the more different the topography maps are from the rest condition.

5.2 Real data

We used two different EEG-based datasets with two MI-BCI sessions. Motivated by the ultimate goal of developing a co-adaptive BCI for motor rehabilitation, the typical two-classes set-up was followed, that is both datasets comprised two MI classes.

Dataset-1. This dataset comprised 10 inexperienced able-bodied BCI users (3 females, 4 left-handed individuals, with a mean age of 25.45 years ± 2.50) previously acquired in [31]. The experiment included two identical sessions, spaced 5 days apart. Approval for the experiment was granted by the local ethics committee (BASEC-Nr. Req-2017-00631, Cantonal Ethics Commission, Zurich, Switzerland). Brain signals were recorded using a portable 64-channel EEG system (eego™rt Ant Neuro, Netherlands) with a sampling frequency of 512 Hz. Electrodes were positioned based on the international 10-20 system, with CPz as the reference and AFz as the ground electrodes. EEG signals were filtered within the frequency range of 0.5 Hz to 40 Hz. Participants performed two mental tasks: i) imagining the movement of their dominant hand (grasping motion, MI) and ii) a rest/relaxation condition (rest). Each session consisted of four runs with short breaks in between. Every run comprised 40 trials (20 for each task), resulting in a total of 160 trials per session. No feedback was provided to the subjects during the sessions. For this study, data was downsampled to 128 Hz and epoched from 0.5 to 2.5 s after the onset of the visual cue. 28 electrodes covering the sensorimotor areas were selected, following [32].

Dataset-2. Known in the literature as $Lee2019\_MI$, is a large publicly available dataset[1] obtained from 54 able-bodied participants in two sessions of MI-BCI [33]. The EEG recordings were acquired with the BrainAmp (Brain Products, Germany) using the nasion as reference and the electro AFz as ground. 62 electrode signals at 1000 Hz were collected as positionated by the 10-20 international montage system. In each session, participants performed a calibration and a test phase. Each phase had 100 trials with balanced right and left hand imagery tasks. During the online test phase, real-time classifier outputs were used to delivered feedback. No feedback was provided during the calibration phase. As before, the EEG signals were windowed from 0.5 to 2.5 s after the onset of the visual cue. Signals were band-pass filtered between 0.5 and 40 Hz, and then downsampled to 125 Hz. 18 channels covering the sensorimotor area were selected, as done in [33]. Given that for the online test phase no information about the indicated mental task to be performed was given, in the current work we used only those trials from the calibration phase. That is, we ended up with two MI-BCI sessions of 100 trials each.

6 Experiments and Results

Experiments were designed to give support and address the following main hypothesis:

H. "In supervised backward online adaptation, the associated cost of transporting a testing trial to match the calibration data distribution reflects the MI user self-regulation capability."

Performing a backward domain adaptation undoubtedly convey the need of relying on a discriminative source domain to where perform the adaptation. In addition, given that such backward adaptation is based on the cued label, the extend at which such information bias the adaptation performance must be studied to fully understand the nuances of the proposed algorithmic support metric. In this way, to prove the aforementioned hypothesis, we first needed to understand whether the following assumptions were held:

Supervised backward adaptation based on the cued label information can successfully address across-sessions data distribution changes if:

A1. the user-generated EEG patterns are discriminative and aligned to the indicated mental task, and

A2. the source data, where the classifier is trained, defines a discriminative space.

In the following, we first present experiments and results to prove $A1$ and $A2$ assumptions. Then, experiments and results that support the main hypothesis H are presented. Throughout these experiments we always used as decoding algorithm the standard CSP+LDA pipeline. Given the way simulated data was generated, only two CSP components were extracted, defining a 2-dimensional feature space. For real data, the typical six components of spatial filters were used [21]. For evaluating the decoding performance, classification accuracy was used. In both simulated and real data, EEG signals were filtered in between 8 and 30 Hz before feature learning. To quantify the user MI-BCI skills, D was used between classes (class distinctiveness), between sessions per each class (session distinctiveness) and trial-wise (trial distinctiveness). For reference, the standard calibration (SC) framework, where not adaption is

[1] Available at `http://gigadb.org/dataset/100542`

applied between sessions, is also implemented. The simulations as well as the corresponding source codes implemented in this work are publicly available at GitHub (`https://github.com/NiCALab-IMAL/SupportiveBackwardAdaptation`).

6.1 Dependence of adaptation success on the user-generated EEG patterns

To evaluate the impact of the user-generated EEG patterns and their correspondence to the indicated mental task (assumption $A1$), two experiments were conducted here. In both experiments, to focus only on the data discriminability at the target domain level, the calibration session (source domain) was simulated as the most well-suited scenario. Thus, the decoding algorithm was trained on simulated data $S_{50,0}$, a session with clear discriminability between MI and rest conditions and no failed trials (see Figure 2). In the subsequent applications, both trained CSP and LDA models were kept fixed, and the recalibration set was defined by the first 20 trials of the testing session ($N_v = 20$).

For the first experiment, we evaluated the decoding performance of the already trained classifier using as testing data sessions from $S_{45,0}$ to $S_{10,0}$. In this way, all EEG patterns belong to the indicated mental task, but with different simulated MI decoding capability. The regularization parameters λ and ν in (2) were kept fixed to 0.1 and 1, respectively. These values were selected to reinforce with positive bias with respect to the indicated mental task information (the higher ν, stronger is the impact of accounting with the cued label to learn the backward transport plan). Results are shown in Figure 3a, demonstrating that the performance of the already trained classifier when SBA is used achieved perfect or near perfect classification values across all simulated sessions. The classification performance of the standard pipeline (SC) is shown for reference. To illustrate in the CSP space how the transportation works, Figure 3b shows for a given session ($S_{15,0}$) the training data, the original testing data as well as the resulting transported testing data. Notably, for most trials, SBA effectively compensates for data distribution shifts between testing and training sets, resulting in high classification accuracy. In the context of the present study, we shall refer to as *adaptation success* those cases at which testing trials are backwardly transported by means of SBA to the correct side of the decision boundary, and thus are correctly classified.

For the second experiment, we evaluated the performance of the perfect trained decoding model in testing sessions with high ERD% but increasing number of failed trials, namely $S_{45,0}$ and from $S_{45,10}$ to $S_{45,50}$. As before, to reinforce positive bias with respect to the system information, the regularization parameter of the group-LASSO penalty was kept fixed and equal to 1 across testing trials. The Sinkhorn regularization parameter was also kept to 0.1. Accuracy results are presented in Figure 3c, where is it clear that SBA achieved comparable classification performance than SC, i.e, a model without adaptation. Figure 3d illustrates for a given testing session ($S_{45,30}$) that failed trials were transported not based on the indicated mental task class but on the feature information. In other words, a trial with EEG patterns resembling MI but "labeled" as rest, was transported to their MI trials peers, regardless of having a rest label, and vice-versa. This result indicates that accounting for the cued label will effectively aid in backward adaptation only when the user-generated EEG patterns align with the indicated mental task. Understanding that SBA successfully achieves adaptation when the EEG patterns correspond to the indicated mental task, assures us that the predicted class of the model will reflect the user's mental intention.

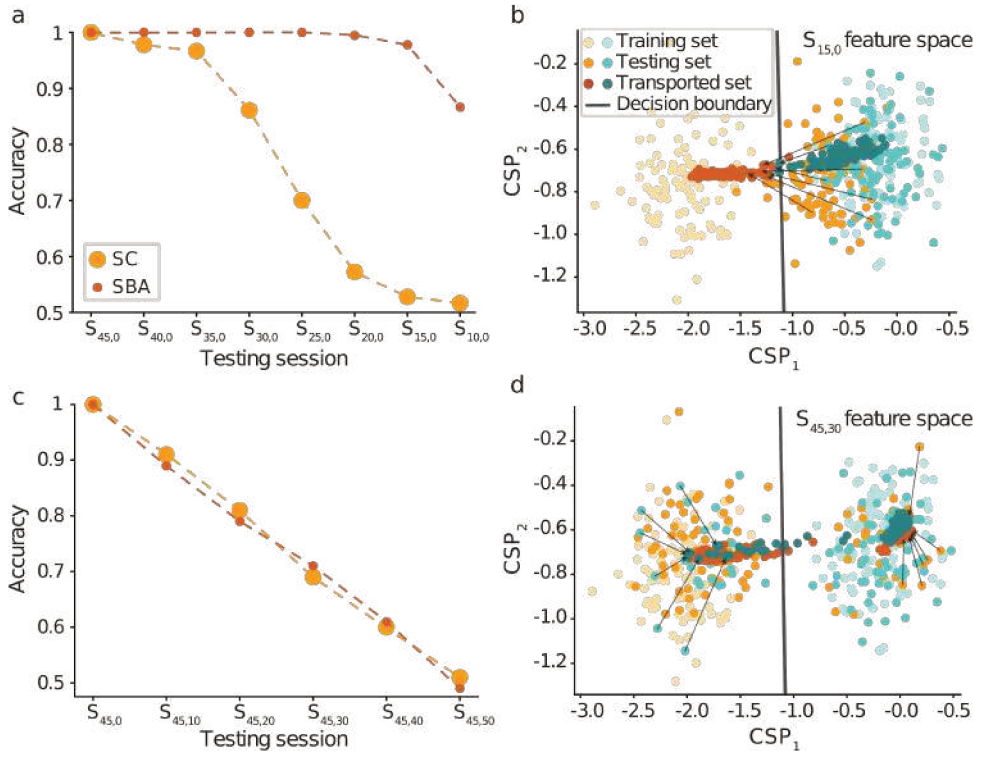

Figure 3: SBA adaptive success dependence on the ERD% and percentage of failed trials. Decoding model trained in a perfect session with high ERD% and no fail trials ($S_{50,0}$). **a.** Accuracy of SC and SBA across different testing sessions with varied ERD% but no failed trials. Dashes lines drawn to connect adjacent points. **b.** Illustration on SBA transportation for a given testing session with no failed trails but low ERD% ($S_{15,0}$). **c.** Accuracy of SC and SBA across different testing sessions with varied fail% but high ERD%. Performance across simulated session are connected with dashes lines to improve results readability. **d.** Illustration of SBA transportation for a given testing session with 30% of failed trails but ERD% equal to 45 ($S_{45,30}$).

.

Altogether, these results evidence the fact that supervised backward adaptation based on optimal transport can address the data distribution changes between the target (testing) and the source (calibration) set when the EEG patterns belongs to the indicated mental task.

6.2 The effect of the training data discriminability on adaptation

While in the previous experiments we focused on studying the testing sessions (target domain) impact in the backward adaptation to a perfect training session (source domain), here we want to evaluate source domains changes and its impact in the adaptation success. In other words, we want to prove assumption $A2$. The decoding algorithm obtained by different training sessions was tested with session $S_{50,0}$, that is, the perfect session at which all trials are aligned to their corresponding class and have ERD% value equal to 50.

First we evaluated the impact in the self-regulation capability by using as training

sessions $S_{45,0}$ to $S_{10,0}$. Class distinctiveness was calculated in each training session to account for a data-driven MI-BCI user's skills metric. As shown in Figure 4a, the class distinctiveness decreased as the ERD% decreased, but always yielding values highly above zero. Interestingly, training accuracy remained fairly high (above 80%). Note that the accuracy of SBA approached near-perfect classification across all trained models.

On the other hand, we used as training data sessions $S_{45,0}$ to $S_{45,50}$. Here, the class distinctiveness rapidly decreased from session to session, yielding values closer to zero when the number of failed trails was equal or above 40% (Figure 4c). Similarly, training accuracy tended to by-chance levels as the fail% increased. It can be observed that SBA failed to transport samples when the source domain class distinctiveness was too low. Therefore, such sessions seemed not to provide a reliable calibration basis for backward adaptation. Figure 4b and d show the feature space with their corresponding LDA decision boundary as well as the CSP topomaps when trained in $S_{10,0}$ and in $S_{45,40}$, respectively. While for $S_{10,0}$ the model was still able to learn discriminable patterns, that was not the case for $S_{45,40}$. Given that both, CSP and class distinctiveness rely on covariance matrices estimation, it is somewhat expected that EEG patterns with near zero distinctiveness result in a low discriminable space. Although this is an extreme scenario, low class distinctiveness can occur when the subject is not able to self-modulate their brain activity, and thus most of the MI trials overlap with the rest ones.

These results demonstrate that backward adaptation should be conducted using a training session with reasonable class discriminability. While this is obvious for any decoding algorithm, extra care must be taken when working in backward adaptation, as it strongly relies on train data distribution. These results indicate that despite of having perfectly performed trials during the application session, adaptation will fail when the training session lacks sufficient class discriminability. In real-life scenarios, this could lead to ambiguity and misleading feedback. To avoid user frustration and to properly support their learning process, the supervised backward adaptation must only be implemented after calibration data defines a discriminable feature space.

6.3 Relationship between algorithmic support and data discriminability

In the previous experiments we proved that if the calibration data is discriminative enough and the EEG patterns belong to the indicated mental task, SBA succeeds in performing the adaptation. But, while the adaptation seems to be possible regardless the ERD%, what is the *cost* of doing so? In other words, how much is the algorithmic support (h) made by SBA to facilitate the proper data alignment? In this section we investigate the relationship between the algorithmic support and the MI-BCI user skills with simulated data.

As before, CSP and LDA were used for EEG decoding. SBA regularization parameters were kept fixed in 0.1 and 1, for λ and ν, respectively. To ensure a perfect calibration session, session $S_{50,0}$ was used as training data. The algorithmic support h was measured at every testing trial as defined in (6). To understand whether a correlation exists between the algorithmic support and the simulated self-regulation capability, we used as testing data the session with ERD% uniformly sampled from 10 to 50, as described in Section 5.1. As shown in Figure 5a, there is a strong negative correlation in between the algorithmic support and the ERD% (Pearson correlation $r = -0.8$, p-value < 0.0001), showing that while adaptation was successfully possible (SC accuracy = 0.86, SBA accuracy = 0.98),

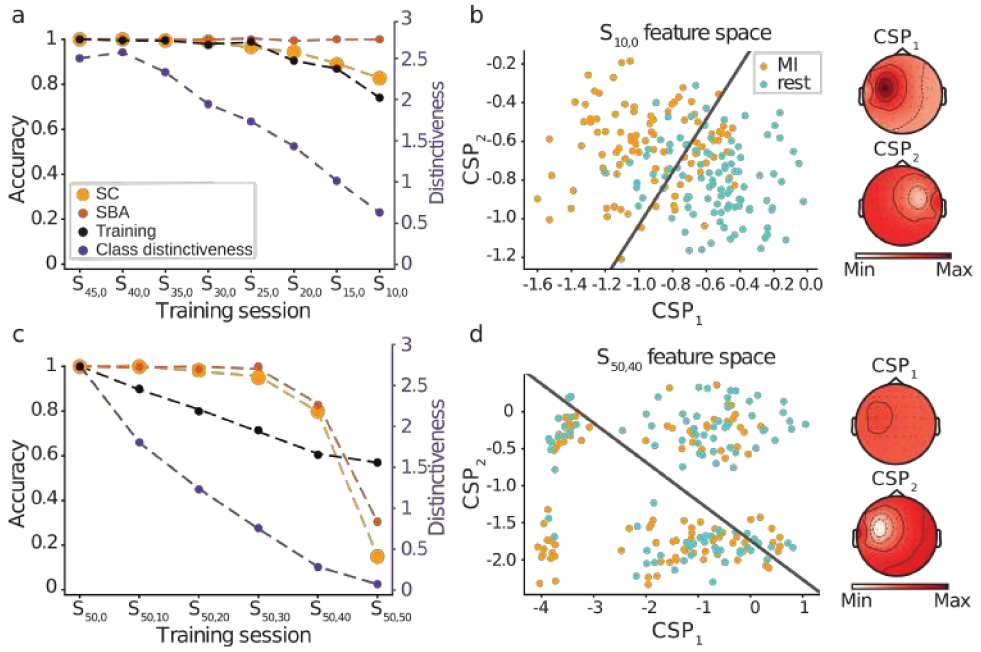

Figure 4: SBA adaptive success dependence on the training data discriminability. Testing data kept fixed to a perfect session with high ERD% and no failed trials ($S_{50,0}$). **a.** Accuracy reached by SC and SBA across different training sessions with varied ERD% but no failed trial. Training model accuracy as well as class distinctiveness is shown. Connecting lines are drawn to improve results readability. **b.** Illustration on the training feature space with the learned LDA decision boundary together with the CSP topomap for a session with low ERD% and no failed trials ($S_{10,0}$). **c.** Accuracy reached by SC and SBA across different training sessions with varied fail% but high ERD%. Training model accuracy as well as class distinctiveness is shown. Here again, connecting lines are drawn to improve results readability. **d.** Illustration on the training feature space with the learned LDA decision boundary together with the CSP topomaps for a session with high ERD% but 40% of failed trials ($S_{45,40}$).

the lower the ERD%, the higher the cost h. Interestingly, the algorithmic support also increased as long as the simulated fatigue effect increased, as shown in Figure 5b. Here it is observed that for the first half of the trials, where no fatigue was present, the algorithmic support was zero or near zero, while for the second half, the algorithmic support increased linearly, following the fatigue level increments as explained in Section 5.1.

While these two experiments were run based on a fixed but known ν parameter to accomplish with strong group-LASSO penalty in (2), in real-life scenarios the regularization parameters might be selected based on data-driven procedures. As explained in Section 3, the regularization parameters selection relies on the recalibration data and the source subset from where to learn the transportation plan [11]. As before, recalibration data conforms the first 20 trials of the testing session ($N_v = 20$). Thus, in the following experiment we let the model decide the best regularization parameters from a list of 20 values, ranging from 0.1 to 10. Similar to before, training data was $S_{50,0}$, but testing data varied from $S_{45,0}$ to $S_{10,0}$. h was calculated for each testing trial in each testing session. The results, depicted in Figure 5c show the mean algorithmic support value together with their standard deviation across all testing trial in each testing session. In line with the previous results, the algorithmic support increased as the ERD% in testing

data decreased. Interestingly, the selected group-LASSO regularization parameter (ν) started to increase from $S_{30,0}$, that is, when the ERD% was lower than 30%. Note that the Sinkhorn regularization parameter (λ) was always chosen to be 0.1.

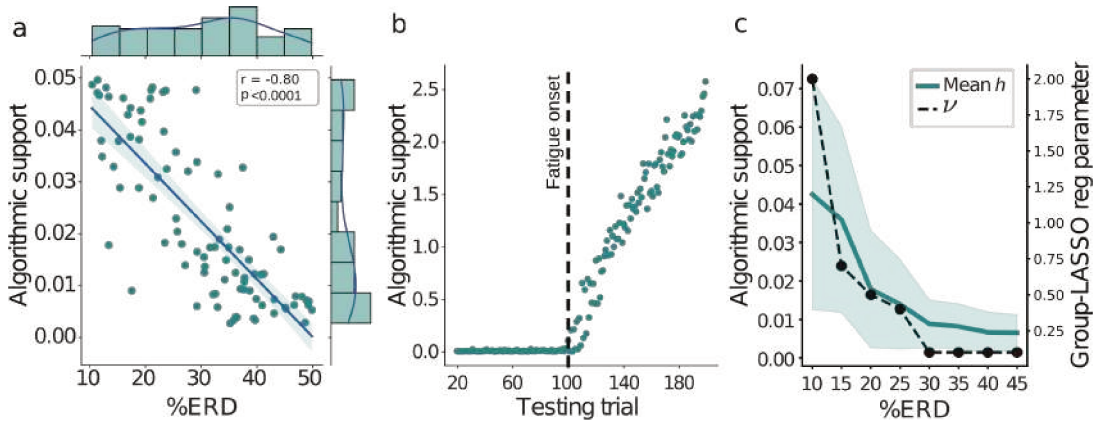

Figure 5: SBA algorithmic support informs about MI-EEG discriminability. **a.** Algorithmic support h as a function of the ERD% in a testing session with uniform ERD% distribution. Text aggregates Pearson correlation analysis between the algorithmic support h and the ERD%. **b.** Algorithmic support h in a testing session with linearly increasing fatigue levels in the second half of the trials. The dashed vertical line indicate the fatigue onset. Each dot represent a trial. **c.** Algorithmic support h at different testing sessions with different ERD% when the regularization parameters are selected by cross-validation. The thick continuous line shows the mean h value across trials, while the shaded area shows standard deviation. The value of the group-LASSO regularization parameter at each tested session is also shown. A connecting line is drawn between points to illustrate the tendency.

6.4 Co-adaptive learning in real data

In order to understand the potential of SBA as a co-adaptive method, real datasets in a subject-specific cross-session MI-BCI scenario were used. For each dataset, the first session was used as the calibration data, from where the decoding algorithm was trained. CSP+LDA was used by setting the number of spatial filters in CSP equal to six, as typically defined in the field [21]. The second session was used as the application session, from where the already trained decoding algorithm was used with SBA applied at each testing trial. To mimic real-life implementation, the regularization parameters were selected for each subject-data by cross-validation, using their corresponding recalibration data. That is, the first 20 trials of the new session were used to find both the source subset as well as the best regularization parameters.

During the application session, the trial distinctiveness as well as the the algorithmic support h was calculated for each testing trial. The session distinctiveness per each class was computed across the training and testing data per each subject in each database. Accuracy was used to assess decoding performance for the classifier with and without adaptation, namely SC and SBA, respectively.

Figure 6a shows the algorithmic support h, trial distinctiveness D_ℓ and accuracy of each subject in Dataset-1 sorted from higher to lower median h value. In general, the bigger the algorithmic support, the higher the trial distinctiveness; indicating that the

model needed more effort in the adaptation as more dissimilar the trial to their corresponding class training data was. But, while decoding accuracy is important to ensure a proper BCI control, that does not always reflect the MI user intention. In this way, it is interesting to observe the accuracy gained when performing the adaptation, shown in top panel of Figure 6a. As can be seen, a large classification performance improvement did not always imply a huge algorithmic support. Based on the results presented in simulated data, this phenomena could indicate that the subject was performing the indicated mental task correctly but with weak modulation (low ERD%). Similarly, cases at which strong algorithmic support was measured but with null or small relative classification performance increments between SC and SBA, can be explained by failed trials, that is, generated EEG patterns by the user that did not correspond to the indicated mental task by the system.

A similar analysis can be done for the results presented in Figure 6b for Dataset-2. In this case, due to the large number of subjects comprising this database, the median of h and D_ℓ value across trials is depicted per subject. Subjects with higher algorithmic support are highlighted with darker colors. Figure 6c aggregates accuracy information with and without data adaptation, showing, here again, that there is no clear correspondence in between algorithmic support and classification improvement.

Considering real-life applications, at which feedback should be delivered as soon as possible, we measured the computational time required to adapt and compute the logarithmic support metric of a given trial by means of SBA. Experiments were run on a standard PC with Intel Core^{TM} i7-6700K CPU. For both datasets, the average SBA computing time was of about 40 ms, around 2.5 times faster than the average time needed to compute the Riemannian trial distinctiveness metric.

Given that experiments were conducted in a cross-sessions scenario, it was important to understand whether correlation existed in between data similarity across classes and sessions, and the algorithmic support measured by means of h. Figure 6d and 6e show for Dataset-1 and Dataset-2, respectively, the median value across trials of h per each subject versus the session distinctiveness value found per each class. It is notable how positive and significantly correlated these two metrics were. This supports the idea that h can be used to measure the MI-BCI user skills.

7 Discussion

Motor imagery-based BCIs are promising technologies for neurorehabilitation [34, 35]. While around 25% of the users cannot command MI-BCIs [36], this may not necessarily indicate that they are not working towards being able to self-regulate their brain activity. Nevertheless, generating stable and distinguishable EEG patterns takes time. The session-to-session learning process must be adequately supported by the system, adapting to data changes. In the present work, we propose an index to assess the discriminability and stability of the user generated patterns across-sessions of BCI use. This measure, referred to as *algorithmic support*, is based on the cost of adapting the EEG trial to its corresponding training peers within a supportive backward adaptation (SBA) process. We showed that the algorithmic support h can be used for online assessing the discriminability and stability of EEG patterns, and thus, SBA has the potential to become a valuable tool for building co-adaptive MI-BCIs.

The algorithmic support metric measures the effort made by SBA when performing

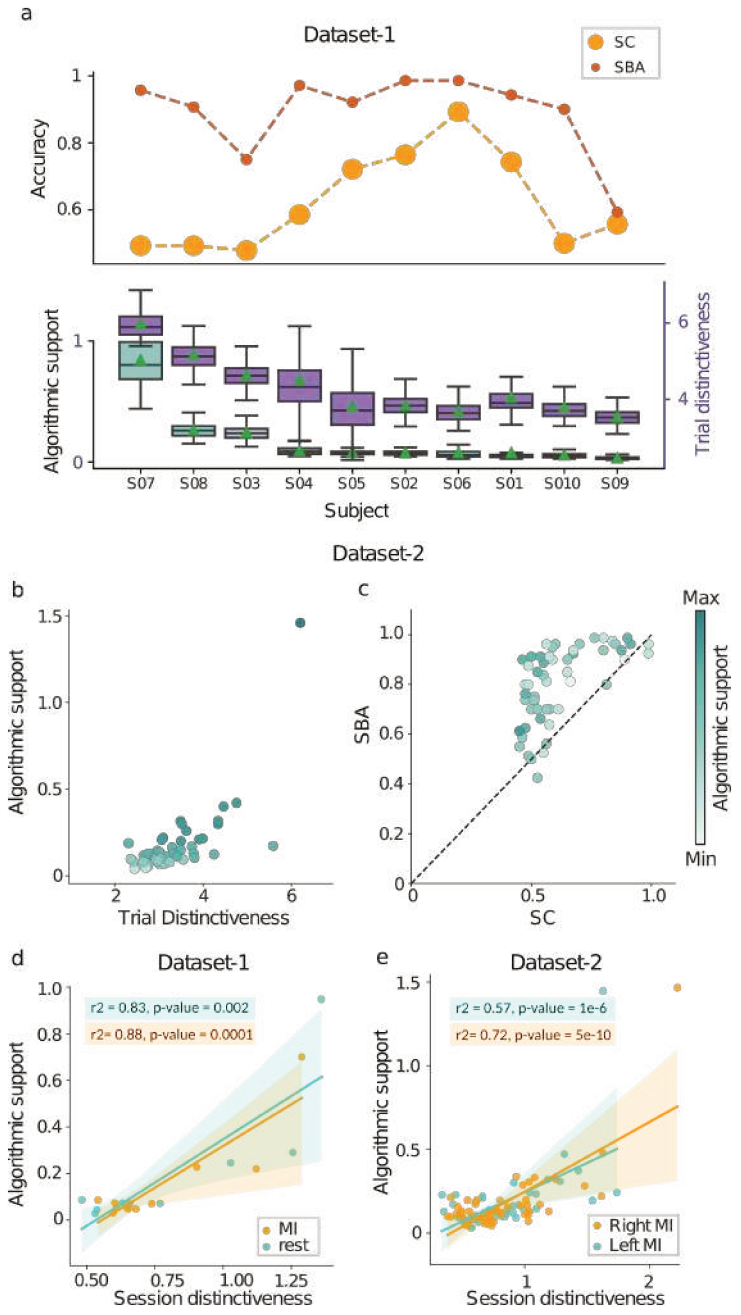

Figure 6: SBA algorithmic support is strongly correlated to Riemannian-based metrics. **a.** Algorithmic support, trial distinctiveness, and accuracy for each subject of Dataset-1 sorted in descending median h value. Lines in the accuracy plots are drawn to help results readability. Boxplots on the button reflect across trial computed metrics, with the left and right y-axes showing range values for the algorithmic support and the trial distinctiveness, respectively. **b.** Algorithmic support as a function of trial distinctiveness, where each dot represent the median value for each subject in Dataset-2. Subjects were colored-coded accordingly to their median h value, where darker colors correspond to higher algorithmic support values. **c.** Accuracy reached by SC and SBA per each subject in Dataset-2. Dots' color follows the increasing order of the corresponding h value per each subject. **d-e.** Algorithmic support as a function of session distinctiveness per each class for Dataset-1 and Dataset-2, respectively. Text aggregates Pearson correlation coefficient found and its p-value per class.

backward optimal transportation in a supervised manner. We studied SBA as a co-adaptive method for MI-BCI. SBA is based on BOTDA [11], an adaptive method designed to match the data distribution of a new session with the data distribution of a calibration session, avoiding classifier retraining. Considering MI-BCI for motor rehabilitation, the domain adaptation is done in a trial-basis using the indicated mental task provided by the system to guide the adaptation. In this work we showed that the backward adaptation based on the system information (cued label) is successful when the user-generated EEG patterns reflect the indicated mental task. That is, data alignment between sessions leads to high accuracy only when the user intention matches the cued label information. Most importantly, we showed that the effort the model exerts in performing such adaptation informs how distinguishable and stable the EEG patterns are.

Experiments on simulated data revealed the correspondence between patterns discriminability and the system information when performing SBA. Throughout extensive simulations, we proved that the adaptation is able to match the data distribution of the new session to the old one only when the EEG patterns belong to the indicated mental task, regardless of the level of ERD%, as shown in Figure 3. These results indicate that the system information will only guide the adaptation to be successful when the generated EEG patterns by the user match the cued mental task. In other words, when the user self-modulate their brain activity in a discriminative and task-related manner. Results on simulated data showed that the backward adaptation was possible also in extreme scenarios at which the simulated ERD% values were as low as 10% (see Figure 2b). While the accuracy of the adaptive model reached perfect classification, we found that the algorithmic support increases as the ERD% decreases (Figure 5c).

A relevant aspect to analyze when performing a backward adaptation is the quality, from the discriminative viewpoint, of the calibration data. Our results showed that data used to calibrate the model (source domain), should reflect data discriminabilty, as measured by the class distinctiveness and the accuracy in training data (see Figure 4). In this way, only "good" calibration sessions are reliable to act as source domain, that is, domains to where perform the backward data adaptation. This finding indicates that for multi-session MI-BCIs, the user might need to perform a good calibration session, as measured by both the training accuracy and class distinctiveness before passing to the application session. In real-life implementations, this could help avoiding user frustration during sessions with feedback. Considering most naive BCI users fail to perform MI in the very first session, different strategies could be asked to be done in different session days until reaching good calibration performance. Only then, user could pass to next stages of the MI-BCI controlling experience.

Frustration with respect to BCI controlling capabilities development can also be diminished if the feedback provided to the user is not solely based on the predicted class by the decoding algorithm. It has been argued that with a feedback that informs the user about the quality of the generated EEG patterns with respect to the indicated mental task, the participant might better modulate their brain activity and thus, improve their BCI controlling capabilities [4, 12]. Here, we showed that the effort put by SBA when performing the adaptation informs about the goodness on the generated EEG patterns. Results through simulations showed that the algorithmic support in performing the adaptation increases as the data discriminative diminishes, as simulated by low ERD% values in the data (see Figure 5a). Data discriminability can also be affected by other brain process that could co-exist during a MI-BCI session. This is the case of mental fatigue, that could appear during MI-BCI sessions perpetuating the underlying task-related brain

oscillations [37]. We showed that, as long as MI-related activity is preserved, data adaptation can be performed but with higher algorithmic support as the fatigue level increases (see Figure 5b). Although it is hard to account for online fatigue biomarkers, this result sheds light on the necessity to monitor fatigue levels within a MI-BCI session.

As claimed by the authors in [4], metrics based on the Riemannian geometry can help identifying user's MI ability beyond the classifier output. This involves measuring the stability and discriminability of the EEG patterns using a parameter-free and machine-independent algorithm. Such an approach can help delineate the decoding roles between the classifier and the user more effectively; however, it strongly relies on the estimation of covariance matrices, which are prone to issues in low electrode counting scenarios. In the current study we propose a novel way of assessing the user MI-BCI skills within an adaptive algorithm. It works at the feature space level, as extracted by any machine learning model or feature engineering process. Real data results show that the algorithmic support, as measured by h, is positive and significantly correlated to well-established metrics based on the Riemannian geometry. Given that class distinctiveness was proven to be positive correlated to mental rotation capabilities [4, 14], the found correlation across a large number of subjects' data suggest that h has the potential to become a valuable tool to measure, trial-by-trial, the effort made by the system to perform the adaptation, and thus explicitly indicate to the user how well the indicated mental task was performed. In this way, it is envisioned that by means of SBA not only across-sessions adaptation is performed but also meaningful feedback be delivered to the user, and thus support their MI-BCI learning process.

We envision SBA as powerful tool for users mastering their BCI control abilities. We aim at fostering independence and the sense of agency during MI-BCI sessions. Nevertheless, it is important to recognize a trade-off: while providing support may ease the user's experience, it can also lead to a lack of true learning and diminished engagement in actively controlling the MI-BCI. Conversely, if users navigate their sessions without adequate guidance, they may struggle to develop the necessary skills and self-awareness required for effectively command the system. With this in mind, and following what is commonly done in the literature [38], in multiple-sessions MI-BCIs, positive feedback can first be imposed. In the context of SBA, such positive bias can be performed by manually setting the group-LASSO regularization parameter to a high value. Then, as sessions progress, if the average algorithmic support decreases from session to session and offline quantifiers of ERD activity show better modulation capabilities, the regularization parameter can be adjusted at the beginning of the online session, managing the frustration-motivation ratio. In this line, our results in simulated data showed that there is a positive relationship between the selected regularization parameter and the algorithmic support, where both increase as long as the sessions become more and more difficult from the class-discriminative viewpoint (see Figure 5c). This last finding may indicate that the data-driven selected regularization parameter has the potential to inform about the cross-session adaptation complexity, where harder adaptive scenarios are more likely to have higher values of the group-LASSO regularization parameter. In closed-loop implementations, the data-driven selected regularization parameter can directly be used as users become more experienced, and thus no positive bias is needed to be imposed by the experimenter.

While the predicted class by the decoding algorithm affects the control efficiency of the BCI, it does not necessary resemble the generated pattern by the MI-BCI user [4]. In this way, the algorithmic support can provide another domain of quantification for

a given trial, beyond predicted class category. As seen in Figure 6, improvements in accuracy after applying data adaptation not always indicate a huge algorithmic support, and vice-versa. Delivering to the user feedback based on not only the predicted class but also on the algorithmic support is expected to lead to more transparency on how the model decoding works, and hopefully guide and support the user throughout the MI-BCI learning process. In this way, visual neurofeedback in a form of an energy bar can be delivered in the closed-loop stimulation protocol. Based on the algorithmic support such energy bar can be colored and updated trial by trial. It is expected that users realize that the better the performed mental task, the lower algorithmic support exerted by the model is; and thus, use that feedback to improve, change or maintain their modulation strategy within a session. The implementation of such a closed-loop stimuli is part of our future plans.

In the context of across-session adaptation, the proposed algorithmic support can also be influenced by non-physiological factors affecting the stability of the EEG. Currently, there is no form to disentangle how much the changes in data are due to modulation variability or non-BCI user-related factors, such as electrode-cap misalignment across the BCI sessions. Neither the algorithmic support nor the Riemannian distinctiveness metric calculated across sessions can inform whether the observed changes in a trial are related to variability due to MI-BCI user skills or any other type of source. Nevertheless, while we acknowledge that more research is needed in this direction, we argue that within a session, the algorithmic support value can only change due to user self-modulation, as other sources of cross-session variability remain constant.

The present study is based on two-sessions without feedback MI-BCI datasets. Undoubtedly, further research is needed to understand how to apply co-adaptation as more sessions become available. In addition, while SBA can be applied at any feature domain representation, the present study focused on the traditional CSP features, which rely on covariance matrices estimations, as Riemannian metrics do. Therefore, it remains to be elucidated whether the algorithmic support metric would exhibit similar behavior when applied in other feature spaces. Future plans involve studying the effects of applying SBA in more complex latent spaces, such as those encoded by deep learning architectures. Similarly, it is necessary to study the effect of the delivered feedback in both data adaptation and algorithmic support assessment. In this direction, our future research plans aim at studying the aforementioned study limitations as well as validating SBA as a co-adaptive method in multiple-sessions of MI-BCI with applications to motor rehabilitation. Next steps involve the use of SBA as an online adaptive algorithm as well as to provide meaningful feedback to the user in a trial-basis. We hope this induces a better user-machine experience, allowing more users to control a MI-BCI, and thus, ultimately potentiating the use of MI-BCI for motor rehabilitation.

8 Conclusion

To support the user's learning process during MI-BCI training, we introduced SBA, a method based on backward optimal transport for domain adaptation, which addresses across-sessions variability while assessing the quality of the user-generated EEG patterns. This study demonstrates the potential of SBA as a co-adaptive approach for MI-BCI systems, highlighting its envisioned capacity to enhance user experience during motor rehabilitation. By enabling online adaptation based on the data distribution of new

sessions, SBA can minimize the need for classifier retraining while providing critical feedback on the user's ability to perform the indicated mental tasks. Our findings reveal that successful adaptation strongly depends on the discriminability and stability of the user-generated EEG patterns in a task-related manner, emphasizing the importance of high-quality calibration data. Furthermore, we established a quantitative measure of adaptation effort through the algorithmic support h index, which offers insights into user performance. This dual functionality of SBA not only facilitates seamless session-to-session adaptation but also has the potential to provide users with meaningful feedback to enhance their MI-BCI control capabilities. As we advance the development of MI-BCI systems, integrating such adaptive methodologies is essential for fostering user confidence and engagement, ultimately improving clinical outcomes in motor rehabilitation.

Data availability statement

The simulated data created for this study is publicly available at Zenodo.

Acknowledgments

This work was partially supported by CONICET, through PIP 2021-2023 11220200100806CO and Universidad Nacional del Litoral, through CAI+D2020-50620190100069LI.

References

[1] Claudia Sannelli et al. "A large scale screening study with a SMR-based BCI: Categorization of BCI users and differences in their SMR activity". In: *PloS one* 14.1 (2019), e0207351.

[2] Aline Roc et al. *A review of user training methods in brain computer interfaces based on mental tasks*. Feb. 2021. DOI: 10.1088/1741-2552/abca17.

[3] Christa Neuper et al. "Imagery of motor actions: Differential effects of kinesthetic and visual-motor mode of imagery in single-trial EEG". In: *Brain Res Cogn Brain Res.* 25.3 (2005), pp. 668–677.

[4] Fabien Lotte and Camille Jeunet. "Defining and quantifying users' mental imagery-based BCI skills: a first step". In: *Journal of Neural Engineering* 15.4 (2018), p. 046030.

[5] Jan Saputra Müller et al. "A mathematical model for the two-learners problem". In: *Journal of Neural Engineering* 14.3 (2017), p. 036005.

[6] Gan Huang et al. "Discrepancy between inter-and intra-subject variability in EEG-based motor imagery brain-computer interface: Evidence from multiple perspectives". In: *Frontiers in Neuroscience* 17 (2023), p. 1122661.

[7] Neethu Robinson et al. "Design Considerations for Long Term Non-invasive Brain Computer Interface Training With Tetraplegic CYBATHLON Pilot". In: *Frontiers in Human Neuroscience* 15 (2021), p. 648275. ISSN: 16625161. DOI: 10.3389/fnhum.2021.648275.

[8] Wojciech Samek, Motoaki Kawanabe, and Klaus-Robert Müller. “Divergence-based framework for common spatial patterns algorithms”. In: *IEEE Reviews in Biomedical Engineering* 7 (2014), pp. 50–72.

[9] Gregory Ditzler et al. “Learning in Nonstationary Environments: A Survey”. In: *IEEE Computational Intelligence Magazine* 10.4 (2015), pp. 12–25.

[10] Sinno Jialin Pan and Qiang Yang. “A Survey on Transfer Learning”. In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), pp. 1345–1359.

[11] Victoria Peterson et al. “Transfer learning based on optimal transport for motor imagery brain-computer interfaces”. In: *IEEE Transactions on Biomedical Engineering* 69.2 (2022), pp. 807–817.

[12] Fabien Lotte, Florian Larrue, and Christian Mühl. “Flaws in current human training protocols for spontaneous Brain-Computer Interfaces: lessons learned from instructional design”. In: *Frontiers in human neuroscience* 7 (2013), p. 568.

[13] Benjamin Blankertz et al. “Neurophysiological predictor of SMR-based BCI performance”. In: *NeuroImage* 51.4 (2010), pp. 1303–1309.

[14] Camille Jeunet et al. “Predicting mental imagery-based BCI performance from personality, cognitive profile and neurophysiological patterns”. In: *PloS one* 10.12 (2015), e0143962.

[15] Eva Maria Hammer et al. “Psychological predictors of SMR-BCI performance”. In: *Biological psychology* 89.1 (2012), pp. 80–6.

[16] Valentina Gamboa von Groll et al. “Large scale investigation of the effect of gender on mu rhythm suppression in motor imagery brain-computer interfaces”. In: *Brain-Computer Interfaces* 11.3 (2024), pp. 87–97. DOI: 10.1080/2326263X.2024.2345449.

[17] Laura J Hagedorn, Nikki Leeuwis, and Maryam Alimardani. “Prediction of inefficient BCI users based on cognitive skills and personality traits”. In: *International Conference on Neural Information Processing*. Springer. 2021, pp. 81–89.

[18] Serafeim Perdikis and Jose del R Millan. “Brain-machine interfaces: a tale of two learners”. In: *IEEE Systems, Man, and Cybernetics Magazine* 6.3 (2020), pp. 12–19.

[19] Nicolas Courty et al. “Optimal Transport for Domain Adaptation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.9 (2017), pp. 1853–1865. DOI: 10.1109/TPAMI.2016.2615921.

[20] L. V. Kantorovich. “On the translocation of masses”. In: *Dokl. Akad. Nauk SSSR, 37* 37.7-8 (1942), pp. 227–229.

[21] Fabien Lotte. “Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain–computer interfaces”. In: *Proceedings of the IEEE* 103.6 (2015), pp. 871–890.

[22] Nicolas Courty, Rémi Flamary, and Devis Tuia. “Domain adaptation with regularized optimal transport”. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2014, pp. 274–289.

[23] Florian Yger, Maxime Berar, and Fabien Lotte. “Riemannian Approaches in Brain-Computer Interfaces: A Review”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25.10 (2017), pp. 1753–1762. DOI: 10.1109/TNSRE.2016.2627016.

[24] Alexandre Barachant et al. “Riemannian geometry applied to BCI classification”. In: *LVA/ICA 2010* (2010), pp. 629–636.

[25] Tingnan Qu et al. “Riemannian distance based channel selection and feature extraction combining discriminative time-frequency bands and Riemannian tangent space for MI-BCIs”. In: *Journal of Neural Engineering* 19.5 (Sept. 2022), p. 056025. DOI: 10.1088/1741-2552/ac9338. URL: https://dx.doi.org/10.1088/1741-2552/ac9338.

[26] Camille Jeunet, Bernard N’Kaoua, and Fabien Lotte. “Advances in user-training for mental-imagery-based BCI control: Psychological and cognitive factors and their neural correlates”. In: *Progress in brain research* 228 (2016), pp. 3–35. DOI: https://doi.org/10.1016/bs.pbr.2016.04.002.

[27] Satyam Kumar et al. “Transfer learning promotes acquisition of individual BCI skills”. In: *PNAS nexus* 3.2 (2024), pgae076. DOI: 10.1093/pnasnexus/pgae137.

[28] Maria Sayu Yamamoto et al. “Class-distinctiveness-based frequency band selection on the Riemannian manifold for oscillatory activity-based BCIs: preliminary results”. In: *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE. 2022, pp. 3690–3693.

[29] Paulo V Ascenção et al. “Evaluation of performance metrics for users of brain computer interfaces during motor imagery”. In: *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. IEEE. 2019, pp. 217–222.

[30] Catalina M Galván et al. “Neurophysiologically meaningful motor imagery EEG simulation with applications to data augmentation”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* (2024).

[31] Victoria Peterson et al. “A penalized time-frequency band feature selection and classification procedure for improved motor intention decoding in multichannel EEG”. In: *Journal of Neural Engineering* 16.1 (2019), p. 016019. DOI: 10.1088/1741-2552/aaf046.

[32] Silvia Marchesotti et al. “Cortical and subcortical mechanisms of brain-machine interfaces”. In: *Human Brain Mapping* 38.6 (2017), pp. 2971–2989. DOI: 10.1002/hbm.23566.

[33] Min-Ho Lee et al. “EEG dataset and OpenBMI toolbox for three BCI paradigms: an investigation into BCI illiteracy”. In: *GigaScience* 8.5 (Jan. 2019), giz002. ISSN: 2047-217X. DOI: 10.1093/gigascience/giz002.

[34] Ravikiran Mane, Tushar Chouhan, and Cuntai Guan. “BCI for stroke rehabilitation: motor and beyond”. In: *Journal of Neural Engineering* 17.4 (2020), p. 041001.

[35] Wenzhe Liao et al. “Motor imagery brain–computer interface rehabilitation system enhances upper limb performance and improves brain activity in stroke patients: a clinical study”. In: *Frontiers in Human Neuroscience* 17 (2023), p. 1117670.

[36] Carmen Vidaurre et al. “Co-adaptive calibration to improve BCI efficiency”. In: *Journal of Neural Engineering* 8.2 (Mar. 2011), p. 025009. DOI: 10.1088/1741-2560/8/2/025009. URL: https://dx.doi.org/10.1088/1741-2560/8/2/025009.

[37] Ruyi Foong et al. “Assessment of the efficacy of EEG-based MI-BCI with visual feedback and EEG correlates of mental fatigue for upper-limb stroke rehabilitation”. In: *IEEE Transactions on Biomedical Engineering* 67.3 (2019), pp. 786–795.

[38] Jelena Mladenović et al. “Towards identifying optimal biased feedback for various user states and traits in motor imagery BCI”. In: *IEEE Transactions on Biomedical Engineering* 69.3 (2021), pp. 1101–1110.

Doctorado en Ingeniería
mención Inteligencia Computacional, Señales y Sistemas

Título de la obra:
Métodos de aprendizaje automático de mínima calibración para interfaces cerebro-computadora basadas en imaginería motora

Autor: Catalina María Galván

Lugar: Santa Fe, Argentina

Palabras Claves:

Interfaz cerebro-computadora, Imaginería motora, Electroencefalografía, Simulación de datos, Aumentación de datos, Adaptación de dominio, Transporte Óptimo