Prediction of psychopathologies using resting-state EEG

Bachelorarbeit

Doroteya Nikolova
0480052

June 28, 2026

Supervisor: Prof. Dr. Benjamin Blankertz
Prof. Dr. Klaus-Robert Müller

Technische Universität Berlin
School of Electrical Engineering and Computer Science
Institute of Software Engineering and Theoretical Computer Science
Neurotechnology

Abstract

Psychiatric disorders are among the most complex conditions to diagnose. Electroencephalography (EEG) offers a promising non-invasive alternative, and objective biomarkers derived from it could support more reliable clinical assessment. This thesis investigates whether resting-state EEG contains discriminative information that allows the binary classification of participants as either subclinical or having at least one diagnosed mental disorder.

To address this question, a Riemannian geometry-based machine learning pipeline was implemented and evaluated. The pipeline extracts OAS-regularized spatial covariance matrices from short overlapping EEG windows, aggregates them to a subject-level representation via the Riemannian mean, projects them into tangent space, and applies elastic-net regularized logistic regression. The pipeline was applied to resting-state EEG recordings from two cohorts - a predominantly clinical group (AD, $n = 146$) and a predominantly subclinical group (WY, $n = 149$) - across both eyes-open and eyes-closed conditions. Model evaluation was performed using nested cross-validation with five outer folds, three inner folds, and five random seeds. Permutation tests were conducted to assess whether observed performance exceeded chance level.

The pipeline did not achieve reliable classification performance in any of the four investigated conditions. Mean outer-fold balanced accuracy ranged from 0.469 to 0.513 across all conditions, remaining close to chance level throughout. All permutation tests yielded non-significant p-values ($p > 0.05$), confirming that none of the observed performances significantly exceeded chance level. These results suggest that resting-state EEG covariance structure alone does not carry sufficient discriminative information to reliably separate a diagnostically heterogeneous clinical group from subclinical participants. Future work should consider disorder-specific clinical groupings, enriched feature representations, or task-evoked EEG paradigms that elicit more consistent disorder-related neural responses.

Zusammenfassung

Psychische Störungen gehören zu den schwierigsten Erkrankungen, die es zu diagnostizieren gilt. Die Elektroenzephalographie (EEG) bietet eine vielversprechende nicht-invasive Alternative, und aus ihr abgeleitete objektive Biomarker könnten eine zuverlässigere klinische Beurteilung unterstützen. Diese Arbeit untersucht, ob Ruhe-EEG diskriminative Informationen enthält, die eine binäre Klassifikation von Probanden als subklinisch oder als Personen mit mindestens einer diagnostizierten psychischen Störung ermöglichen.

Zur Beantwortung dieser Frage wurde eine auf Riemannscher Geometrie basierende maschinelle Lernpipeline implementiert und evaluiert. Die Pipeline extrahiert OAS-regularisierte räumliche Kovarianzmatrizen aus kurzen überlappenden EEG-Fenstern, aggregiert diese mithilfe des Riemannschen Mittelwerts zu einer subjektspezifischen Repräsentation, projiziert sie in den Tangentialraum und wendet eine Elastic-Net regularisierte logistische Regression an. Die Pipeline wurde auf Ruhe-EEG-Aufzeichnungen zweier Kohorten angewendet - einer überwiegend klinischen Gruppe (AD, $n = 146$) und einer überwiegend subklinischen Gruppe (WY, $n = 149$) - jeweils für die Augen-offen- und Augen-geschlossen-Bedingungen. Die Modellevaluation erfolgte mittels verschachtelter Kreuzvalidierung mit fünf äußeren Faltungen, drei inneren Faltungen und fünf zufälligen Seeds. Permutationstests wurden durchgeführt, um zu prüfen, ob die beobachtete Leistung das Zufallsniveau überschreitet.

Die Pipeline erzielte in keiner der vier untersuchten Bedingungen eine zuverlässige Klassifikationsleistung. Die mittlere ausgewogene Genauigkeit der äußeren Faltungen lag zwischen $0{,}469$ und $0{,}513$ über alle Bedingungen hinweg und blieb durchgehend nahe dem Zufallsniveau. Alle Permutationstests lieferten nicht-signifikante p-Werte ($p > 0{,}05$), was bestätigt, dass keine der beobachteten Leistungen das Zufallsniveau signifikant überschritt. Diese Ergebnisse legen nahe, dass die Kovarianzstruktur von Ruhe-EEG allein keine ausreichenden diskriminativen Informationen enthält, um eine diagnostisch heterogene klinische Gruppe zuverlässig von subklinischen Probanden zu trennen. Zukünftige Arbeiten sollten störungsspezifische klinische Gruppierungen, angereicherte Merkmalsrepräsentationen oder aufgabenbasierte EEG-Paradigmen in Betracht ziehen, die konsistentere störungsbezogene neuronale Reaktionen hervorrufen.

Contents

List of Abbreviations

Abbreviation	Full Term
AD	Predominantly clinical cohort label used in the DFG study (exact designation internal to the dataset)
ADHD	Attention-Deficit/Hyperactivity Disorder
BCI	Brain-Computer Interface
CSP	Common Spatial Patterns
CV	Cross-Validation
DFG	Deutsche Forschungsgemeinschaft (German Research Foundation)
EEG	Electroencephalography
FgMDM	Filter geodesic Minimum Distance to Mean
FIR	Finite Impulse Response
FN	False Negative
fNIRS	Functional Near-Infrared Spectroscopy
FP	False Positive
GAD	Generalized Anxiety Disorder
HPD	Hermitian Positive Definite
KDE	Kernel Density Estimation
MNE	Minimum Norm Estimates (EEG/MEG analysis library: MNE-Python)
OAS	Oracle Approximating Shrinkage
OCD	Obsessive-Compulsive Disorder
PHOB	Specific Phobia
PSD	Power Spectral Density
PTSD	Post-Traumatic Stress Disorder
SAD	Social Anxiety Disorder
SAGA	Stochastic variance-reduced gradient optimization solver (Defazio et al., 2014)
SPD	Symmetric Positive Definite
TN	True Negative
TP	True Positive
WY	Predominantly subclinical cohort label used in the DFG study (exact designation internal to the dataset)

1 Introduction

Mental disorders represent a major global health challenge. According to the data from the World Health Organization (WHO), in 2021 nearly one in seven people (approximately 1.1 billion individuals) worldwide were battling a mental disorder. Anxiety and depressive disorders are the most common [32]. These conditions impact emotional regulation, cognitive functioning, and physical well-being. Early and accurate diagnosis is therefore essential for effective treatment. However, traditional diagnostic procedures - clinical interviews, self-report questionnaires, neuropsychological assessments, and neuroimaging - face several limitations, including subjectivity, time-consuming administration, dependence on clinical expertise, and limited sensitivity to subtle early-stage changes [26].

In recent years, machine learning (ML) and deep learning (DL) methods have shown promising results in detecting and predicting mental health conditions [42]. Among neuroimaging techniques, electroencephalography (EEG) stands out due to its high temporal resolution, non-invasiveness, low cost, and direct measurement of neuronal activity, making it suitable for large-scale and clinical applications [22]. It is used to investigate neural correlates of psychiatric conditions.

The aim of this thesis is to evaluate whether resting-state EEG contains discriminative information that allows the classification of participants as either non-clinical or having at least one diagnosed mental disorder. To this end, the thesis implemented a Riemannian-geometry-based machine learning pipeline that extracts covariance features from short EEG windows, projects them into tangent space, and applies logistic regression within a nested cross-validation framework.

1.1 Literature Review

Early EEG analysis methods often relied on variance-based features, such as power spectral density (PSD) or spatial filters like Common Spatial Patterns (CSP) [40, 44, 37]. While these approaches capture important frequency-specific or channel-specific information, they do not fully represent the spatial covariance structure of multichannel EEG signals [3]. This motivated a shift toward covariance-based representations [5, 4].

As demonstrated by Barachant et al., short segments of ongoing EEG can be classified directly through their spatial covariance matrices, enabling reliable detection of mental states without requiring precisely timed stimuli [5]. In subsequent work, they showed that treating covariance matrices as geometric objects on the manifold of symmetric positive-definite (SPD) matrices yields more discriminative features than Euclidean approaches, outperforming CSP-based pipelines in motor imagery BCI tasks [3]. In their later work, the authors further introduced Riemannian-based kernel methods tailored to the SPD manifold, achieving a mean classification accuracy of 86% on a

motor imagery BCI task compared to 80% for the conventional CSP pipeline [4].

Congedo et al. provided a comprehensive review of Riemannian geometry for EEG analysis, highlighting its robustness to noise, suitability for short-window covariance estimation, and strong performance across diverse BCI tasks [12]. Collectively, these studies established Riemannian geometry as a powerful framework for extracting discriminative information from EEG covariance matrices.

Previous work in the DFG project has shown that variance-based EEG features can predict anxiety dimensions. Extending this approach to covariance-based Riemannian representations may therefore capture additional clinically relevant information, which motivates the present work.

1.2 Database

In the present thesis, the EEG data is collected within a large DFG-funded research project investigating the neural correlates of two latent anxiety dimensions: anxious arousal (physiological hyperarousal) and anxious apprehension (cognitive worry and rumination). In total, 322 participants were recruited, comprising both subclinical and clinical samples. Data collection for the subclinical part was completed between February 2018 and June 2019, while the clinical part began in July 2021 under strict COVID-19 hygiene protocols. All participants completed computer-based tasks, including resting-state EEG, flanker, picture-viewing, threat-processing, and fear-conditioning paradigms. This thesis focuses exclusively on the resting-state EEG recordings.

The subclinical group (n = 174) included healthy individuals differing in their anxiety dimension profiles, which could be convergent (high or low in both anxious arousal and anxious apprehension) or divergent (one dimension more pronounced than the other). The clinical group (n = 148) consisted of participants diagnosed with at least one mental disorder, including obsessive-compulsive disorder (OCD), social anxiety disorder (SAD), specific phobia (PHOB), generalized anxiety disorder (GAD), panic disorder, agoraphobia, post-traumatic stress disorder (PTSD), depressive disorders, somatic symptom disorders, separation anxiety, sleep-wake disorders, body dysmorphic disorder, tic and related disorders (including trichotillomania and excoriation disorder), hoarding disorder, eating disorders, and attention-deficit/hyperactivity disorder (ADHD) as well as intermittent explosive disorder. In this thesis, diagnostic status is modeled as a binary classification between participants with at least one diagnosed mental disorder and those without one.

In the dataset used for this thesis, each participant was assigned to one of two task groups: AD, which contains mostly clinical participants with at least one disorder, and WY, which contains mostly non-clinical participants. The AD group spans a broad age range from 18 to 63 years, whereas the WY group is considerably younger and more homogeneous, ranging from 18 to 30 years. The two groups differ in the duration and structure of their resting-state EEG task: the AD group completed an 8-minute session, whereas the WY group completed a 4-minute session. Both sessions were structured as alternating one-minute eyes-open (O) and eyes-closed (C) periods, but the specific O/C order differed between groups: the AD group followed either OCCOCOOC or COOCOCCO, while the WY group followed either OCOC or COCO.

According to the clinical metadata, the AD group originally contained 163 participants, of whom 36 were classified as healthy cases, and the WY group contained 159 participants, of whom 138 were healthy. These numbers reflect the full set of participants for whom diagnostic information was available for this thesis.

However, the number of subjects available for analysis in this thesis is determined by the intersection of the clinical metadata and the resting-state EEG recordings stored as `.mat` files. During preprocessing, the following step was applied:

```
data_index = mat_df.merge(labels_df, on="EEG_ID", how="inner")
```

This inner merge retains only those participants for whom both a valid `.mat` file and a corresponding clinical label exist. As a result, the effective sample sizes differ from the original counts in the metadata file that was provided. After merging, the dataset used in the analysis consisted of 146 AD subjects and 149 WY subjects. This merged dataset forms the basis for all subsequent preprocessing, feature extraction, and classification analyses.

EEG recordings were acquired using a 64-channel BrainCap system (Easycap GmbH, M10 layout) following the international 10-10 electrode placement standard. Electrodes covered frontal, central, parietal, occipital, and temporal regions, with reference and ground electrodes positioned at REF and GND, respectively. The layout provided dense spatial coverage, enabling analysis of hemispheric asymmetries and connectivity patterns relevant to anxiety dimensions.

1.3 Structure of the Thesis

Chapter 1 introduces the research problem and its relevance within the broader context of mental-health diagnostics. It reviews previous literature on EEG-based prediction of psychopathologies, outlines the motivation and objectives of the present work, and formulates the main research question and hypotheses. It also describes the dataset used in this thesis, including participant recruitment, diagnostic grouping, and EEG acquisition procedures. It provides an overview of the clinical and subclinical samples, the recording setup, and the experimental design of the resting-state paradigm.

Chapter 2 presents the methodological framework. It explains the preprocessing pipeline, feature extraction procedures, and classification approach in detail. To give the reader a clear overview, Figure 2.1 summarizes all preprocessing and classification steps applied in the analysis.

Chapter 3 reports the experimental results obtained from the classification analyses. It presents the performance metrics for each task condition and rest order, including balanced accuracy, confusion matrices, and generalization gaps, and summarizes the outcomes across all evaluated folds and seeds.

Chapter 4 interprets the findings in the context of existing literature, discusses their implications for clinical EEG research, and outlines potential directions for future work.

Chapter 5 concludes the thesis by summarizing the main contributions, answering the research question, and highlighting the significance of the results for advancing data-driven approaches in mental-health diagnostics.

2 Methods

The present thesis investigates whether resting-state EEG covariance features can distinguish clinical from subclinical participants in the AD and WY cohorts. To address this question, a Riemannian geometry-based classification pipeline was developed, combining covariance estimation, tangent-space projection, and penalized logistic regression. Nested cross-validation was employed to obtain unbiased performance estimates, and a permutation test was used to assess whether results exceeded chance level. Figure 2.1 provides an overview of the full preprocessing and classification pipeline. The following sections describe each step in detail.

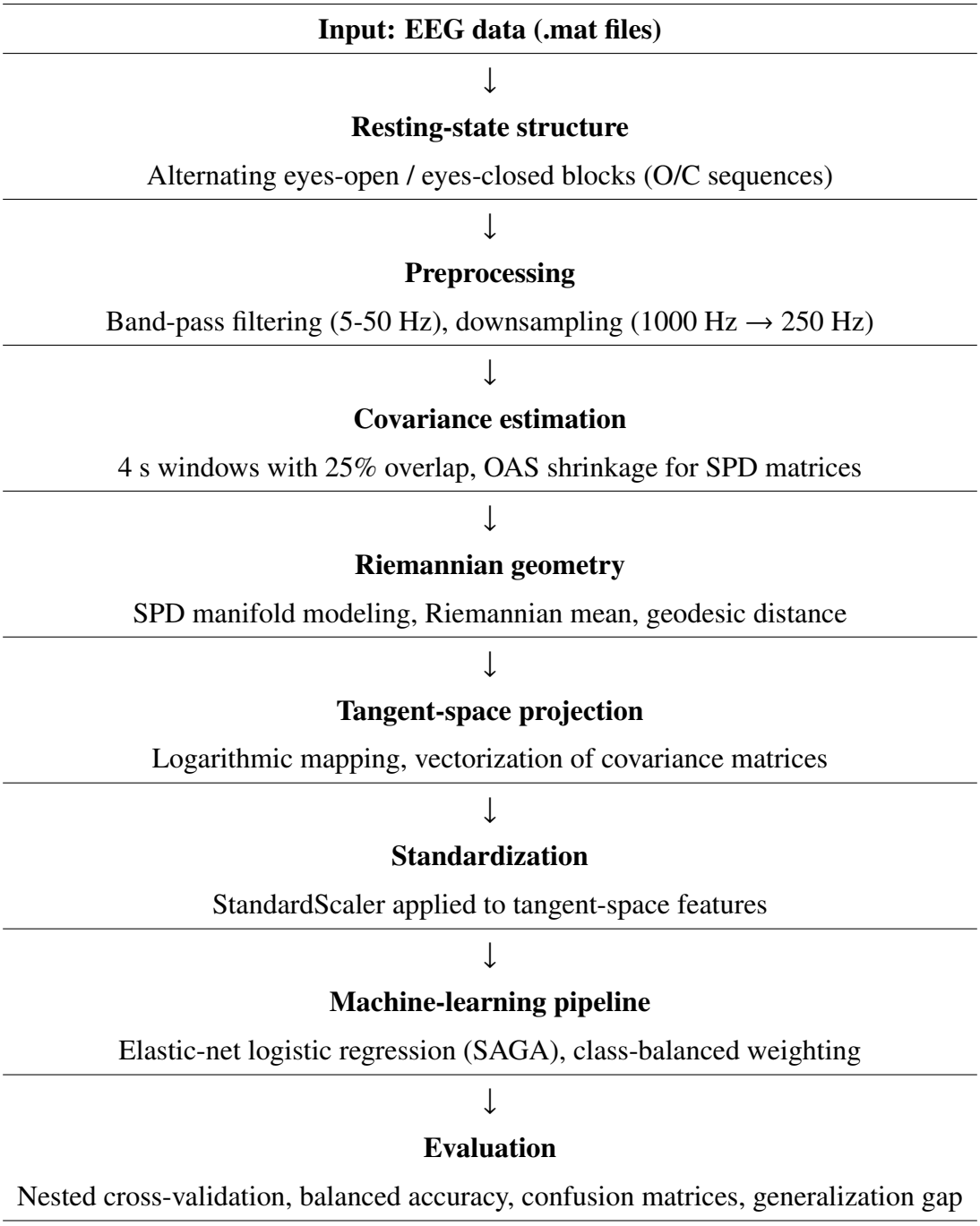


Figure 2.1: Overview of the full preprocessing and classification pipeline. The analysis begins with EEG recordings stored as `.mat` files and proceeds through preprocessing, covariance estimation, Riemannian feature extraction, tangent-space projection, and machine-learning evaluation.

2.1 Feature Extraction

2.1.1 Preprocessing

The primary objective of EEG preprocessing is to reduce noise and artifacts originating from both external (environmental) and internal (biological) sources. In general, noise-reduction strategies fall into two main categories: removing contaminated data segments and attenuating artifacts through signal-processing techniques [18].

Splitting the Data

The resting-state EEG recordings of both AD and WY subjects consisted of alternating eyes-open (O) and eyes-closed (C) periods. During eyes-open segments, participants were instructed to fixate on a central point; during eyes-closed segments, they were asked to close their eyes and relax. Each segment lasted one minute, and transitions were indicated by a tone.

For the AD group, the resting-state session lasted eight minutes and followed one of two predefined sequences: OCCOCOOC or COOCOCCO. For the WY group, the session lasted four minutes, following either OCOC or COCO. To ensure that feature extraction was performed on homogeneous signal segments, the continuous EEG recording was split into contiguous blocks of samples belonging to the same condition (eyes-open or eyes-closed). This prevents mixing of conditions within a single analysis window and preserves the temporal structure of the experimental design.

The splitting procedure was implemented in Python. The logic can be summarized with the following pseudocode:

Algorithm 1: Split EEG Signal into Contiguous Condition Blocks

Input: EEG signal $X \in \mathbb{R}^{C \times T}$, label vector y, target condition l
Output: List of contiguous signal blocks
Initialize empty list B and set $start \leftarrow \emptyset$;
for $i = 1$ **to** T **do**
    if $y_i = l$ ***and*** $start = \emptyset$ **then**
        $start \leftarrow i$;
    else if $y_i \neq l$ ***and*** $start \neq \emptyset$ **then**
        Append segment $X[:, start : i - 1]$ to B;
        $start \leftarrow \emptyset$;
if $start \neq \emptyset$ **then**
    Append segment $X[:, start : T]$ to B;
return B;

Band-pass filtering (5-50 Hz)

Band-pass filtering is a common step in EEG preprocessing used to isolate physiologically meaningful oscillations [47]. EEG signals comprise a mixture of oscillatory components spanning several

frequency ranges, typically grouped into canonical frequency bands, each associated with distinct cognitive or attentional states [22]. As described by Khosla et al., gamma activity (>30 Hz) reflects transient functional integration of neural processes, while beta activity (14-30 Hz) corresponds to alertness and cognitive engagement and may increase under panic conditions. Furthermore, the alpha band (8-13 Hz) reflects relaxed wakefulness with closed eyes; attenuation of alpha power has been linked to anxiety and emotional tension. This makes it particularly relevant for the present thesis. According to the authors, theta activity (4-7.5 Hz) is associated with subconscious processing, relaxation, and meditative states, and increases in theta power often reflect memory-related demands. Delta activity (0.1-3.5 Hz) predominates during deep sleep [22] and is therefore not relevant for the purposes of this study.

The EEG signals in this thesis were band-pass filtered between 5 and 50 Hz to remove slow drifts (<5 Hz) and high-frequency noise (>50 Hz) originating from muscle activity and electrical interference. Although the lower theta range (4-5 Hz) is excluded, the remaining 5-7.5 Hz portion still preserves physiologically meaningful theta activity while avoiding slow-drift contamination. This frequency range preserves the principal oscillatory rhythms relevant to anxiety-related neural processes, including alpha, beta, and low gamma [22].

Filtering was performed using the MNE-Python library [25], which implements zero-phase finite impulse response (FIR) filters designed via the windowed-sinc method. Zero-phase FIR filters are used in EEG research because they preserve peak latencies, avoid phase distortions, and provide a stable and well-controlled frequency response [47].

Mathematical foundation Digital filters can be described mathematically by their transfer function in the z-domain [25]:

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}}, \tag{2.1}$$

where b_k and a_k are the numerator and denominator coefficients. In the time domain, the filter output $y(n)$ is computed from the input $x(n)$ as [25]:

$$y(n) = \sum_{k=0}^{M} b_k x(n-k) - \sum_{k=1}^{N} a_k y(n-k). \tag{2.2}$$

For FIR filters, all denominator coefficients are zero ($a_k = 0$), so the filter reduces to [25]:

$$y(n) = \sum_{k=0}^{M} b_k x(n-k). \tag{2.3}$$

This means that each output value depends only on the previous input values, instead of both the previous input and output values. In the context of EEG and the present work, the digital filter acts on each channel's time series to attenuate frequency components outside the desired range (5-50 Hz). This non-recursive form ensures stability and a linear phase response [25, 47]. In MNE, the filter is applied forward and backward to obtain a zero-phase signal without introducing phase distortions.

Filtering was implemented using the mne.filter.filter_data() function, which applies the same zero-phase FIR design as raw.filter() but operates directly on NumPy arrays [25]. This approach was chosen because the EEG data were stored as array matrices rather than MNE Raw objects.

Downsampling

In signal processing, an important consequence of the sampling theorem is the concept of the Nyquist frequency [41], which states that, in order to reconstruct a band-limited signal without aliasing, the sampling rate must be at least twice the highest frequency component of the signal. Mathematically, this requirement is expressed as [41]:

$$f_s \geq 2B, \tag{2.4}$$

where f_s is the sampling frequency and B is the bandwidth of the signal, i.e., the highest frequency present in the signal. Equivalently, all meaningful frequency components of a properly sampled signal must lie below the Nyquist frequency, defined as [41]:

$$f_{\text{Nyquist}} = \frac{f_s}{2}. \tag{2.5}$$

The process of reducing a sampling rate by an integer factor is referred to as downsampling (also "decimation"). Downsampling reduces the sampling frequency from f_s to $f_s' = f_s/M$, which also reduces the Nyquist frequency. The new Nyquist frequency is therefore:

$$f_{\text{Nyquist,new}} = \frac{f_s'}{2} = \frac{f_s}{2M}. \tag{2.6}$$

To prevent aliasing, the signal must not contain frequencies above this new limit. The decimation procedure in scipy.signal.decimate [46] ensures this by applying a low-pass anti-aliasing filter before subsampling, which keeps every M-th sample. This guarantees that the maximum frequency of the filtered signal satisfies:

$$f_{\max} < \frac{f_s}{2M}, \tag{2.7}$$

so that no aliasing artifacts occur after downsampling.

In the present study, the EEG signals were downsampled from 1000 Hz to 250 Hz ($M = 4$) using scipy.signal.decimate [46]. Since the data were already band-pass filtered to 5-50 Hz, all remaining frequencies lie well below the new Nyquist frequency of 125 Hz. Reducing the sampling frequency decreases data dimensionality and computational load within the 5-50 Hz band of interest.

2.1.2 Windowed Covariance Estimation Using OAS Shrinkage

To obtain stable and discriminative covariance representations from the resting-state EEG, the continuous signal was segmented into short, overlapping windows and a covariance matrix was computed for each window. In Riemannian EEG analysis, covariance matrices are estimated from short EEG epochs and treated as points on the SPD manifold [2, 23]. Because EEG is non-stationary and non-linear [22], computing a single covariance matrix over the entire recording could mix heterogeneous dynamics and degrade the quality of the SPD features.

Window segmentation

For each contiguous block of eyes-open or eyes-closed data, the preprocessed signal $X \in \mathbb{R}^{C \times T}$, where T is the number of samples and C the number of channels, was divided into overlapping windows of fixed duration. In this work, windows of $4\,\mathrm{s}$ with 25% overlap were used. If f_s denotes the sampling rate after downsampling, the window length is:

$$L = \lfloor 4 \cdot f_s \rfloor, \qquad \text{step} = \lfloor L(1 - 0.25) \rfloor.$$

Each window $X_w \in \mathbb{R}^{C \times L}$ was treated as an independent short-time segment from which a spatial covariance matrix was estimated.

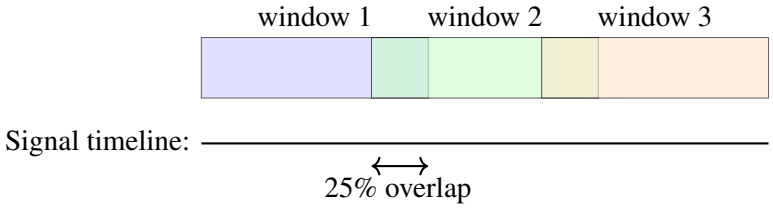


Figure 2.2: Illustration of overlapping window segmentation.

The segmentation was implemented in Python:

Algorithm 2: Segment EEG Signal into Overlapping Windows

Input: Preprocessed EEG signal $X \in \mathbb{R}^{C \times T}$ with C channels and T time samples, window length L, step size S
Output: List of overlapping EEG windows W
Initialize empty list W;
for *start* $\leftarrow 0$ **to** $T - L$ ***step*** S **do**
    stop $\leftarrow$ start + L;
    Append $X[:, \text{start} : \text{stop}]$ to W;
return W;

Covariance estimation

Given a window X_w and the number of samples in a window L, the empirical covariance is [5]

$$\hat{\Sigma} = \frac{1}{L-1} X_w X_w^\top. \tag{2.8}$$

However, it is important that the number of samples L is not small relative to the number of channels C, a known issue in EEG processing that could cause overfitting [31]. Poorly conditioned covariance matrices negatively affect Riemannian distances.

To address this, Chen et al. [10] derived the Oracle Approximating Shrinkage (OAS) estimator, which computes an optimal shrinkage coefficient that minimizes the mean-squared error of the covariance estimate. In this work, spatial covariance matrices were computed using the Covariances estimator from the open-source PyRiemann library [1] where this formula is implemented as:

$$\tilde{\Sigma} = (1 - \rho)\hat{\Sigma} + \rho\, v I_C, \tag{2.9}$$

where I_C is the $C \times C$ identity matrix, v is a scaling factor that preserves the overall power of the signal, and $\rho \in [0, 1]$ is the shrinkage coefficient controlling the trade-off between the empirical covariance and the regularized target.

Scaling factor The scaling factor v ensures that the shrinkage does not alter the global variance of the data [10]. It is defined as the average variance across all channels:

$$v = \frac{1}{C}\mathrm{Tr}(\hat{\Sigma}), \tag{2.10}$$

where $\mathrm{Tr}(\hat{\Sigma})$ denotes the trace of the covariance matrix, i.e., the sum of its diagonal elements. This normalization preserves the overall energy of the EEG signal consistent after shrinkage [10].

Shrinkage coefficient (OAS)

The optimal value of ρ determines how strongly the covariance is pulled toward the identity matrix. Chen et al. [10] derived a closed-form expression for ρ under the assumption of Gaussian-distributed data, known as the *Oracle Approximating Shrinkage* (OAS) estimator. It minimizes the mean-squared error between the true covariance and its estimate, yielding:

$$\rho_{\mathrm{OAS}} = \min\left\{ \frac{(1 - 2/p)\mathrm{Tr}(\hat{\Sigma}^2) + \mathrm{Tr}^2(\hat{\Sigma})}{(n + 1 - 2/p)\left[\mathrm{Tr}(\hat{\Sigma}^2) - \frac{\mathrm{Tr}^2(\hat{\Sigma})}{p}\right]}, 1 \right\}, \tag{2.11}$$

where n is the number of samples in the window ($n = L$) and p is the number of channels ($p = C$). The trace terms $\mathrm{Tr}(\hat{\Sigma})$ and $\mathrm{Tr}(\hat{\Sigma}^2)$ represent the first and second-order moments of the covariance matrix. The OAS estimator automatically adapts the shrinkage intensity to the data dimensionality and sample size [10], making it suitable for covariance estimation from short EEG windows.

Implementation

This step computes the empirical covariance for each window and applies the OAS regularization to obtain a set of symmetric positive definite (SPD) matrices:

$$\{\tilde{\Sigma}_1, \tilde{\Sigma}_2, \ldots, \tilde{\Sigma}_{n_{\mathrm{win}}}\} \subset \mathcal{P}(C). \tag{2.12}$$

These matrices are well-conditioned and suitable for Riemannian processing, ensuring numerical stability and improved discriminative power in subsequent tangent-space classification. In practice, each window was passed to the Covariances estimator from PyRiemann [1]:

```
covs = Covariances(estimator="oas").fit_transform(epochs)
```

where epochs is the stacked array of windows with shape (n_{win}, C, L). The fit_transform() method first fits the estimator to the input data (computing the OAS shrinkage parameters) and then transforms each epoch into its corresponding covariance matrix, returning an array of SPD matrices. The use of OAS follows the recommendations of Olias et al. [31], who demonstrated that Gaussian shrinkage

estimators outperform classical methods such as Ledoit-Wolf when applied to EEG data with limited samples.

Subject-level aggregation

Because each subject yields multiple window-level covariance matrices, a single representative SPD matrix was obtained using the Riemannian (Fréchet) mean:

$$\bar{\Sigma} = \arg\min_{\Sigma \in \mathcal{P}(C)} \sum_{w=1}^{n_{\text{win}}} \delta_R^2(\Sigma, \tilde{\Sigma}_w), \tag{2.13}$$

ensuring that the final subject-level descriptor respects the geometry of the SPD manifold. This is introduced deeply in the next chapters. These preprocessing steps ensure that the extracted covariance features reflect stable neural activity relevant to distinguishing clinical from non-clinical subjects.

2.2 Riemannian Geometry and Tangent Space

This section explains the transformation of the SPD matrices into Euclidean vectors for the binary classification task in the present thesis.

2.2.1 PyRiemann

PyRiemann is a Python library that implements machine-learning methods on the manifold of Symmetric (resp. Hermitian) Positive Definite (SPD) (resp. HPD) matrices [1]. It follows the scikit-learn [35] API, enabling Riemannian geometry to be integrated into standard pipelines for classification, regression, and clustering of EEG, MEG (Magnetoencephalography), and fNIRS data.

In EEG analysis, covariance matrices computed from multichannel signals contain meaningful spatial information, including channel variances, cross-channel correlations, and oscillatory synchrony [2]. Furthermore, these covariance matrices are SPD by construction. According to Barachant et al., instead of treating them as ordinary Euclidean vectors, Riemannian geometry models them on their natural curved manifold, where distances, means and tangent-space projections are defined in a geometrically consistent way. Prior work has shown that Riemannian metrics yield more robust and discriminative representations for EEG classification compared to Euclidean approaches [3, 12].

2.2.2 Riemannian metric

Given a short-time EEG segment $X \in \mathbb{R}^{C \times N_t}$ with C channels and N_t samples, the spatial covariance matrix is estimated using the sample covariance estimator [5]:

$$P = \frac{1}{N_t - 1} X X^\top. \tag{2.14}$$

Since P is symmetric positive definite, it lies on the SPD manifold $\mathcal{P}(C)$. The Riemannian distance between two covariance matrices P_1 and P_2 is defined as [5]:

$$\delta_R(P_1, P_2) = \left(\sum_{c=1}^{C} \log^2 \lambda_c \right)^{1/2}, \tag{2.15}$$

where $\{\lambda_c,\ c = 1, \ldots, C\}$ are the eigenvalues of $P_1^{-1} P_2$ [16]. This distance corresponds to the geodesic curve length between two points on the SPD manifold [36], where the matrix logarithm naturally appears as the inverse of the exponential map, ensuring affine invariance and capturing the intrinsic curvature of the space.

An important property of the affine-invariant Riemannian metric is its *congruence-invariance* [36]. This means that the Riemannian distance between covariance matrices remains unchanged under invertible linear transformations of the data, such as channel-wise scaling or mixing. Consequently, Riemannian approaches are robust to certain amplitude and spatial transformations of EEG covariance matrices and may reduce the need for additional normalization or spatial filtering steps compared with conventional approaches [12].

For comparison, it is important to understand why Euclidean distance between covariance matrices is not always appropriate for such classification tasks. It is defined as the Frobenius norm of the difference of two matrices [11]:

$$d_E(\Sigma_1, \Sigma_2) = \|\Sigma_1 - \Sigma_2\|_F \tag{2.16}$$

This treats covariance matrices as points in a flat space, when they live on a curved manifold [29]. Following Moakher et al., this is not invariant under inversion, meaning that a matrix and its inverse are not the same distance from the identity matrix. This violates the natural geometry of SPD matrices, where a matrix and its inverse should be symmetric around the identity. Furthermore, the Euclidean arithmetic mean is defined as [11]:

$$\bar{\Sigma}_E = \frac{1}{I} \sum_{i=1}^{I} \Sigma_i \tag{2.17}$$

This mean suffers from the *swelling effect* [11], where the determinant of the mean can become larger than the determinants of all individual matrices. Because this thesis relies on covariance-based features for subject-level classification, using a geometrically consistent metric is essential for obtaining stable and meaningful representations. The affine-invariant Riemannian distance and its associated mean address these issues [29].

Given a set of covariance matrices $\{P_m\}_{m=1}^{M}$, their Riemannian (Fréchet) mean is defined as the matrix that minimizes the sum of squared Riemannian distances [5]:

$$\bar{P} = \arg\min_{P} \sum_{m=1}^{M} \delta_R^2(P, P_m). \tag{2.18}$$

This way, when multiple covariance matrices are available - for example, many windows from the same subject - their average will be computed in a way that respects the manifold structure. Although no closed-form solution exists, the mean can be computed efficiently using iterative algorithms [2].

As mentioned, PyRiemann [1] treats covariance matrices as points in the curved manifold. The shortest path between two points P_1 and P_2 on the Riemannian manifold of SPD matrices is defined as [2]:

$$\gamma(t) = P_1^{1/2} \left(P_1^{-1/2} P_2 P_1^{-1/2} \right)^t P_1^{1/2}, \quad t \in [0, 1]. \tag{2.19}$$

Analogously to the matrix logarithm, the power of SPD matrices can be computed using diagonalization:

$$P^t = V D^t V^{-1}, \tag{2.20}$$

where V and D are the eigenvectors and eigenvalues of P, respectively.

All Riemannian computations in this thesis were performed using the open-source PyRiemann library [1] and this Fréchet mean is used as the reference point for tangent-space projection, enabling the use of Euclidean classifiers such as logistic regression.

2.2.3 Tangent Space Projection

Covariance matrices extracted from EEG windows are Symmetric Positive Definite (SPD) and therefore lie on the curved Riemannian manifold $\mathcal{P}(C)$ of SPD matrices [3], which has intrinsic dimension $C(C+1)/2$, corresponding to the number of unique elements in a symmetric $C \times C$ matrix. Because this space is non-Euclidean, standard linear classifiers such as logistic regression cannot operate directly on SPD matrices. Tangent space projection provides a principled way to locally "flatten" the manifold, enabling the use of Euclidean machine-learning methods while preserving the geometric structure of covariance representations [3].

Scalar product

The space $\mathcal{P}(C)$ forms a differentiable Riemannian manifold where at each point $C \in \mathcal{P}(C)$, one can define a tangent space $T_C\mathcal{P}(C)$, which is a Euclidean vector space of symmetric matrices [3]. Thus, the Riemannian metric induces an inner product between tangent vectors $S_1, S_2 \in T_C\mathcal{P}(C)$ (i.e., two symmetric matrices):

$$\langle S_1, S_2 \rangle_C = \mathrm{tr}(S_1 C^{-1} S_2 C^{-1}), \tag{2.21}$$

which defines lengths and angles in the tangent space. This scalar product is the foundation of the Riemannian kernel introduced later.

Logarithmic and exponential maps

The logarithmic map projects a covariance matrix Σ_i from the manifold onto the tangent space at a reference point C_{ref} [3]:

$$S_i = \mathrm{Log}_{C_{\mathrm{ref}}}(\Sigma_i) = C_{\mathrm{ref}}^{1/2} \, \mathrm{logm}\left(C_{\mathrm{ref}}^{-1/2} \Sigma_i C_{\mathrm{ref}}^{-1/2} \right) C_{\mathrm{ref}}^{1/2}. \tag{2.22}$$

The inverse operation, the exponential map, projects a tangent vector S_i back onto the manifold [3]:

$$\Sigma_i = \mathrm{Exp}_{C_{\mathrm{ref}}}(S_i) = C_{\mathrm{ref}}^{1/2}\,\mathrm{expm}\left(C_{\mathrm{ref}}^{-1/2} S_i C_{\mathrm{ref}}^{-1/2}\right) C_{\mathrm{ref}}^{1/2}. \tag{2.23}$$

These two operations are inverses of each other, and enable local linearization and reconstruction of SPD matrices [4].

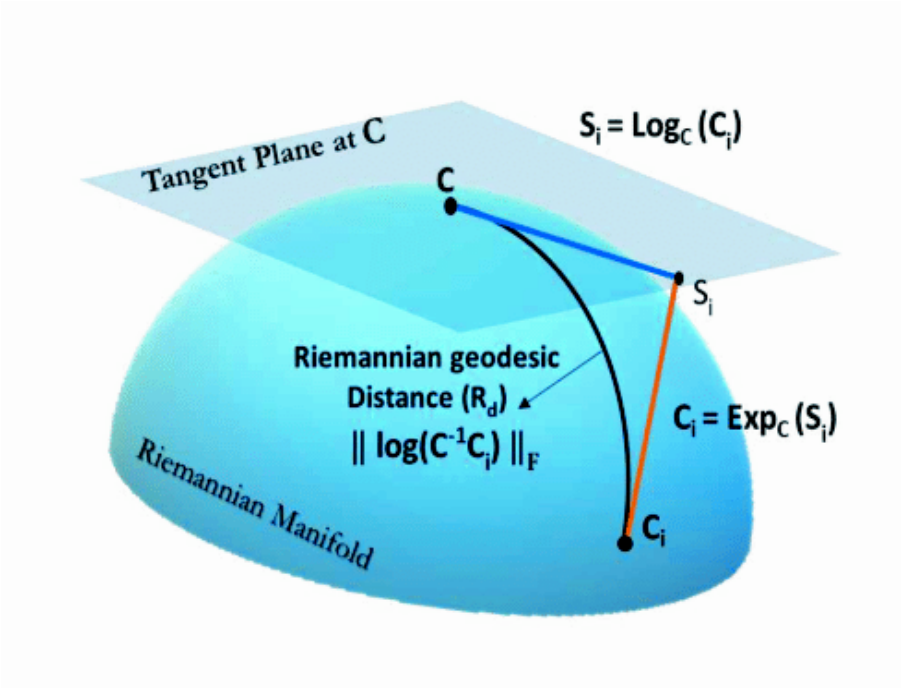


Figure 2.3: Illustration of the Riemannian manifold and tangent-space mapping. Adapted from Singh, A., Lal, S., and Guesgen, H., "Small Sample Motor Imagery Classification Using Regularized Riemannian Features," *IEEE Access*, vol. PP, pp. 1–1, Apr. 2019, doi: 10.1109/ACCESS.2019.2909058.

Riemannian kernel

Using the scalar product defined above, a kernel can be constructed to measure similarity between covariance matrices while respecting the manifold's curvature [4]:

$$k_R(C_i, C_j; C_{\mathrm{ref}}) = \langle \phi(C_i), \phi(C_j) \rangle_{C_{\mathrm{ref}}}, \tag{2.24}$$

where $\phi(C) = \mathrm{Log}_{C_{\mathrm{ref}}}(C)$ is the mapping function. Expanding this expression yields [4]:

$$k_R(C_i, C_j; C_{\mathrm{ref}}) = \mathrm{tr}\left[\mathrm{logm}(C_{\mathrm{ref}}^{-1/2} C_i C_{\mathrm{ref}}^{-1/2})\,\mathrm{logm}(C_{\mathrm{ref}}^{-1/2} C_j C_{\mathrm{ref}}^{-1/2})\right]. \tag{2.25}$$

This kernel defines a valid inner product in the tangent space and satisfies Mercer's theorem, allowing its use in kernel-based classifiers [4]. In this thesis, however, we rely on the tangent-space representation and apply a linear classifier rather than a kernel method.

Symmetric matrix formulation

For computational convenience, the kernel can be reformulated using symmetric matrices [4]:

$$\widetilde{S}_i = C_{\text{ref}}^{-1/2}\text{Log}_{C_{\text{ref}}}(C_i)C_{\text{ref}}^{-1/2} = \text{logm}(C_{\text{ref}}^{-1/2}C_iC_{\text{ref}}^{-1/2}), \tag{2.26}$$

so that

$$k_R(C_i, C_j; C_{\text{ref}}) = \text{tr}(\widetilde{S}_i\widetilde{S}_j) = \text{vect}(\widetilde{S}_i)^\top\text{vect}(\widetilde{S}_j). \tag{2.27}$$

This equivalence allows a kernel-free implementation by transforming covariance matrices into symmetric tangent matrices and applying linear classification on their half-vectorized form [4].

Vectorization and inverse operation

Each symmetric tangent matrix S_i is vectorized using the operator $\text{upper}(\cdot)$, which extracts the upper triangular part and applies a $\sqrt{2}$ weighting to off-diagonal elements to preserve the Frobenius norm [3]:

$$s_i = \text{upper}(S_i) \in \mathbb{R}^{C(C+1)/2}. \tag{2.28}$$

Furthermore, the reciprocal operation, $\text{unvect}(\cdot)$, reconstructs the symmetric matrix from its vectorized form, ensuring reversibility between matrix and vector representations.

Choice of reference matrix

The reference matrix C_{ref} defines the point on the manifold $\mathcal{M}$ where the tangent space is computed [4]. It acts as the origin of the local Euclidean approximation and determines how covariance matrices are projected. Several choices exist:

- **Identity matrix** I_E: Setting the reference matrix to the identity $C_{\text{ref}} = I_E$, where I_E is the $E \times E$ identity matrix, results in the *log-Euclidean kernel*, denoted k_{LE} [4]:

  $$k_{LE}(\text{vect}(C_i), \text{vect}(C_j)) = k_R(\text{vect}(C_i), \text{vect}(C_j); I_E) = \text{tr}[\text{logm}(C_i)\,\text{logm}(C_j)]. \tag{2.29}$$

  This kernel corresponds to centering the tangent space at the identity and is interpreted as a Frobenius inner product.

- **Arithmetic mean**: Another choice for C_{ref} is the average of all covariance matrices, defined as [4]:

  $$\mathfrak{A}(C_1, \ldots, C_P) = \frac{1}{P}\sum_{p=1}^{P} C_p. \tag{2.30}$$

  This mean minimizes Euclidean distance between matrices but does not account for the manifold's curvature [3].

- **Geometric (Fréchet) mean**: The most geometrically consistent choice is the Riemannian (Fréchet) mean, introduced by Moakher [28], defined as:

$$\mathfrak{G}(C_1, \ldots, C_P) = \arg\min_{C \in \mathcal{P}(C)} \sum_{p=1}^{P} \delta_R^2(C, C_p), \tag{2.31}$$

where the Riemannian distance between two SPD matrices is given by:

$$\delta_R(C_1, C_2) = \| \log(C_1^{-1/2} C_2 C_1^{-1/2}) \|_F = \left(\sum_{i=1}^{E} \log^2 \lambda_i \right)^{1/2}, \tag{2.32}$$

and $\{\lambda_i\}$ are the eigenvalues of $C_1^{-1} C_2$. According to [4], the geometric mean provides the best local approximation of the manifold and ensures affine invariance.

Equivalence between kernels If covariance matrices are "whitened" using the reference matrix C_{ref} [4]:

$$\widetilde{C}_p = C_{\text{ref}}^{-1/2} C_p C_{\text{ref}}^{-1/2}, \tag{2.33}$$

then applying the kernel with C_{ref} is equivalent to applying it with the identity matrix:

$$k_R(C_i, C_j; C_{\text{ref}}) = k_R(\widetilde{C}_i, \widetilde{C}_j; I_E). \tag{2.34}$$

This shows that "whitening" the data is mathematically equivalent to shifting the tangent space to the identity [4].

Reference matrix in this work

In this thesis, tangent space projection was implemented using the TangentSpace(metric="riemann") transformer from the PyRiemann library [1]. By default, PyRiemann computes the reference matrix C_{ref} as the Riemannian (geometric) mean of all covariance matrices provided during the fit() step, as defined in Equation 2.31.

This reference point is then used in the subsequent transform() step to project each covariance matrix Σ_i into the tangent space according to:

$$S_i = \text{Log}_{C_{\text{ref}}}(\Sigma_i). \tag{2.35}$$

Calling the fit() method therefore means that the tangent space is centered at the Riemannian mean of the training data.

2.3 Machine-Learning Pipeline

The purpose of this section is to demonstrate how the classifier was trained and evaluated in this thesis.

2.3.1 Feature Standardization Using `StandardScaler`

After projecting the covariance matrices into the tangent space, each trial is represented by a feature vector $s_i \in \mathbb{R}^{C(C+1)/2}$ containing the upper triangular elements of the symmetric tangent matrix, as described in the sections above. Because these features differ in magnitude and variance across dimensions, standardization was applied to ensure numerical stability and balanced contribution of all features to the classifier.

The feature vectors were standardized using the StandardScaler from the scikit-learn library [35]. This transformation performs z-score normalization so that each feature has zero mean and unit variance [17], defined as:

$$z = \frac{x - \mu}{\sigma}, \tag{2.36}$$

where μ and σ denote the mean and standard deviation computed from the training set. The fit() step computes the mean and standard deviation as [35]:

$$\mu_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij}, \tag{2.37}$$

$$\sigma_j = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \mu_j)^2}, \tag{2.38}$$

and the subsequent transform() step applies the normalization using these parameters according to Equation 2.36. In the pipeline implementation, this process is automatically executed through the fit_transform() method during training and the transform() method during testing.

This ensures that all features are centered at zero and have unit variance, preventing features with larger numerical ranges from dominating the optimization process. This is important for linear models with regularization, such as the elastic-net logistic regression used in this work, where the penalty term assumes comparable scales across all input dimensions [33].

2.3.2 Logistic Regression Classifier

After standardizing the feature vectors, the binary classification was performed using a penalized logistic regression model implemented via sklearn.linear_model.LogisticRegression [35]. Logistic regression is a supervised machine learning algorithm and a linear probabilistic classifier that models the posterior probability of class membership [13]. It is particularly suitable for binary classification tasks, such as predicting clinical or subclinical participants in this study, based on resting-state EEG.

For a binary classification problem with feature vector

$$x = (x_1, x_2, \ldots, x_p)^\top, \tag{2.39}$$

logistic regression computes a linear combination of the input features [13]:

$$z = \beta_0 + \sum_{j=1}^{p} \beta_j x_j, \tag{2.40}$$

where β_0 is the intercept and β_j are the model coefficients.

The linear predictor z is transformed into a probability through the logistic (sigmoid) function, which maps any real-valued input into the interval $[0, 1]$ [14]:

$$P(y = 1 \mid x) = \frac{1}{1 + e^{-z}} = \frac{1}{1 + e^{-(\beta_0 + \beta^\top x)}}. \tag{2.41}$$

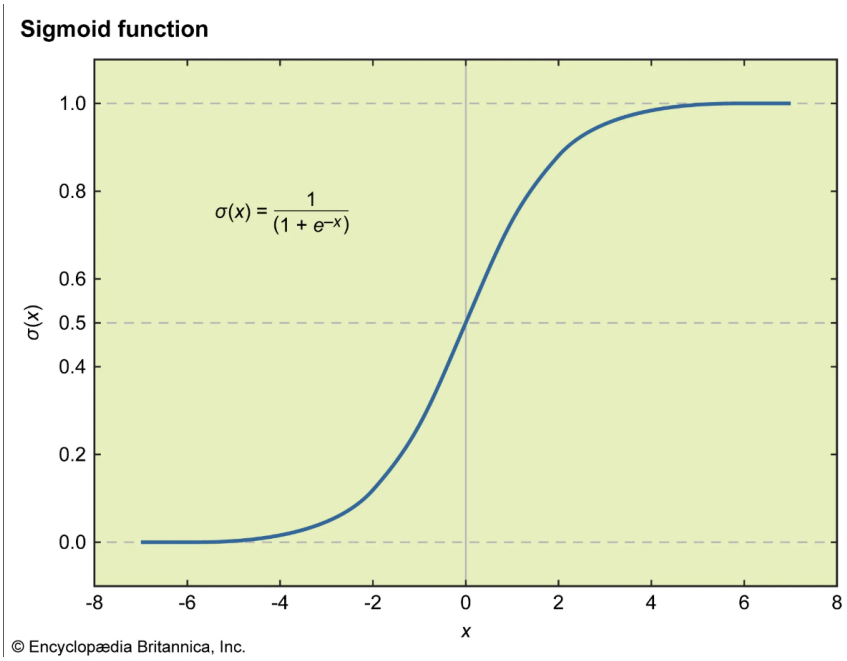


Figure 2.4: Sigmoid function $\sigma(x)$.[1]

This function defines the conditional probability of the positive class $y = 1$ given the features x. The probability of the negative class is $P(y = 0 \mid x) = 1 - P(y = 1 \mid x)$ [14].

According to Cox, a sample is assigned to the positive class if $P(y = 1 \mid x) \geq 0.5$, which is equivalent to the linear decision rule $\beta^\top x + \beta_0 \geq 0$. The sigmoid function's S-shaped curve ensures that the model output can be interpreted as a probability, making logistic regression a natural choice for probabilistic binary classification.

This formulation originates from the *logit model* introduced by David R. Cox in 1958 [13], which defines the log-odds of the event as a linear function of the predictors:

$$\log \frac{P(y = 1 \mid x)}{1 - P(y = 1 \mid x)} = \beta_0 + \beta^\top x. \tag{2.42}$$

[1] Adapted from Baugh, L. S., "Sigmoid function", *Encyclopaedia Britannica*, 17 Nov. 2025. Available at: https://www.britannica.com/science/sigmoid-function (Accessed 16 June 2026).

The logistic function is the inverse of this logit transformation, converting log-odds back into probabilities. This probabilistic interpretation makes logistic regression different from ordinary least squares regression, as it models the likelihood of categorical outcomes rather than continuous values [20].

In modern implementations such as scikit-learn [35], logistic regression is solved by minimizing the negative log-likelihood (cross-entropy) of the Bernoulli-distributed outcomes:

$$\mathcal{L}_{\log}(w, b) = -\sum_{i=1}^{n} \left[y_i \log P(y_i = 1 \mid x_i) + (1 - y_i) \log(1 - P(y_i = 1 \mid x_i)) \right], \tag{2.43}$$

optionally combined with a regularization term to prevent overfitting. The optimization procedure and regularization strategy are described in the following subsections.

Logistic Loss and Regularization

The logistic regression model described above is trained by estimating its parameters $\beta = (\beta_0, \beta_1, \ldots, \beta_p)$ through maximum likelihood estimation. For n training samples, the likelihood of the observed labels is [30]:

$$L(\beta) = \prod_{i=1}^{n} P(y_i \mid x_i), \tag{2.44}$$

and maximizing this likelihood is equivalent to minimizing the negative log-likelihood, also known as the *logistic loss* or *cross-entropy loss* [14]:

$$\mathcal{L}_{\log}(w, b) = -\frac{1}{n} \sum_{i=1}^{n} \left[y_i \log \sigma(z_i) + (1 - y_i) \log(1 - \sigma(z_i)) \right], \quad z_i = w^\top x_i + b. \tag{2.45}$$

Furthermore, this loss, which is convex in (w, b), measures the discrepancy between the predicted probabilities and the true labels. It penalizes incorrect predictions more strongly when the model assigns high confidence to the wrong class, ensuring stable probabilistic outputs [30].

In practice, logistic regression is often combined with a regularization term to control model complexity and improve generalization [30]. The regularized objective minimized by scikit-learn [35] is of the form:

$$\min_{w,b} \; \mathcal{L}_{\log}(w, b) + \lambda \, \Omega(w), \tag{2.46}$$

where $\Omega(w)$ is a penalty on the weights and $\lambda > 0$ is the regularization strength. The user-facing parameter C in scikit-learn [35] is the inverse of the regularization strength:

$$\lambda = \frac{1}{C}. \tag{2.47}$$

where smaller values of C correspond to stronger regularization, whereas larger values allow the model to fit the training data more closely. In this study, the regularization strength was selected through grid search over the values $C \in \{0.001, 0.01, 0.1, 1.0\}$, enabling the model to adapt its complexity to the data.

Elastic-Net Regularization

In this work, Elastic-Net is used as a regularization that combines the ℓ_1 (Lasso) and ℓ_2 (Ridge) penalties, providing a balance between feature selection and coefficient shrinkage [48].

L1 penalty (Lasso) The ℓ_1 penalty is defined as [48]:

$$\Omega_{\ell_1}(w) = \sum_{j=1}^{p} |w_j|. \tag{2.48}$$

L1 regularization promotes sparsity by shrinking some coefficients exactly to zero, effectively performing feature selection [43]. This makes the property useful when only a subset of features contributes meaningfully to the classification task.

L2 penalty (Ridge) The ℓ_2 penalty is defined as [48]:

$$\Omega_{\ell_2}(w) = \sum_{j=1}^{p} w_j^2. \tag{2.49}$$

L2 regularization shrinks coefficients smoothly toward zero but generally retains all features [19], which is considered to improve stability and mitigate collinearity among correlated predictors.

Elastic-Net penalty Elastic-Net combines both penalties into a single convex formulation [48]:

$$\Omega_{\text{EN}}(w) = (1-\alpha)\frac{1}{2}\sum_{j=1}^{p} w_j^2 + \alpha \sum_{j=1}^{p} |w_j|, \tag{2.50}$$

where $\alpha \in [0, 1]$ controls the balance between ℓ_1 and ℓ_2 regularization. In scikit-learn [35], this parameter corresponds to l1_ratio, which implements the mixing

$$\text{l1\_ratio} \cdot \ell_1 + (1 - \text{l1\_ratio}) \cdot \ell_2. \tag{2.51}$$

The limiting cases are:

$$\begin{aligned} \alpha = 0 \quad &\Rightarrow \text{pure Ridge } (\ell_2), \\ \alpha = 1 \quad &\Rightarrow \text{pure Lasso } (\ell_1), \\ 0 < \alpha < 1 \quad &\Rightarrow \text{Elastic-Net (mixed } \ell_1\text{–}\ell_2). \end{aligned}$$

The ℓ_1 component encourages sparsity in the weight vector (feature selection), while the ℓ_2 component stabilizes the solution and mitigates collinearity among features [48]. This combination could be advantageous in high-dimensional settings such as tangent-space EEG features, where many correlated dimensions may be redundant. The parameter l1_ratio was optimized during nested cross-validation over the values

$$\text{l1\_ratio} \in \{0.2, 0.5, 0.8\},$$

corresponding to regularization schemes with a stronger ℓ_2 component, an equal balance of ℓ_1 and ℓ_2, and a stronger ℓ_1 component, respectively.

SAGA Solver

Non-smoothness in the elastic-net objective arises from the ℓ_1 term, whose lack of a derivative at zero induces sparsity [48]. To efficiently solve this composite objective, the solver="saga" option was used. SAGA is a stochastic variance-reduced gradient optimization method specifically designed for large-scale convex problems [15]. According to Defazio et al., it extends stochastic gradient descent (SGD) by reducing the variance of gradient estimates, leading to faster and more stable convergence.

SAGA optimizes finite-sum problems of the form:

$$F(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x) + h(x), \tag{2.52}$$

where $f_i(x)$ is a smooth convex loss for sample i (such as the logistic loss), and $h(x)$ is a possibly non-smooth convex regularizer (such as the elastic-net penalty) [15]. This matches the structure of penalized logistic regression exactly, where the data term corresponds to the per-sample logistic loss and the regularizer corresponds to the elastic-net penalty. The algorithm combines the efficiency of stochastic updates with variance reduction and proximal handling of the non-smooth regularizer [15].

In scikit-learn's implementation of penalized logistic regression, the SAGA solver minimizes the composite objective [35]:

$$\mathcal{J}(w) = \mathcal{L}_{\text{weighted}}(w) + \lambda \left(\alpha \|w\|_1 + \frac{1-\alpha}{2} \|w\|_2^2 \right), \tag{2.53}$$

where $\mathcal{L}_{\text{weighted}}(w)$ is the weighted logistic loss and the second term corresponds to the Elastic-Net regularization. SAGA is also one of the few solvers in scikit-learn that supports ℓ_1 and elastic-net regularization directly through proximal operations, which motivates its use in this study.

Stopping criteria: max_iter and tol

The iterative SAGA optimization is controlled by two parameters: max_iter and tol. The parameter max_iter specifies the maximum number of passes (epochs) over the training data, while tol defines the convergence tolerance, i.e., the minimum relative improvement in the objective function or parameter vector required between successive iterations for the optimization to be considered converged [35].

In this thesis, the following values were used:

```
max_iter = 15000
tol      = 1e-3
```

A relatively large max_iter value of 15 000 was chosen to ensure that the SAGA solver converges reliably across all hyperparameter settings during nested cross-validation. SAGA provides fast convergence for smooth and strongly convex objectives, but its theoretical convergence rate becomes slower in the presence of non-smooth terms such as the ℓ_1 component of the elastic-net penalty [15]. In high-dimensional settings with correlated features, such as EEG tangent-space representations,

the variance of stochastic gradients can further slow convergence. Increasing the iteration budget therefore should help prevent premature termination in difficult optimization regimes and allow the coefficients to stabilize.

The tolerance parameter tol was set to 10^{-3}, which is slightly more relaxed than the default value of 10^{-4} used in scikit-learn's LogisticRegression implementation [35]. This choice provides a practical balance between computational efficiency and optimization accuracy: smaller tolerances (e.g., 10^{-4} or 10^{-5}) yield marginally more precise solutions but substantially increase runtime, whereas larger tolerances (e.g., 10^{-2}) risk stopping before the model has fully converged. The value 10^{-3} was therefore selected as a compromise that ensures stable convergence without excessive computational cost.

If the stopping criterion is not met within max_iter iterations, scikit-learn issues a convergence warning indicating that the solution may not be fully optimized [35]. In practice, no such warnings occurred.

Class weighting

The dataset used in this thesis is imbalanced, with different numbers of clinical and subclinical participants in both AD and WY groups. To mitigate bias toward the majority class, the option class_weight="balanced" was used. In scikit-learn, this automatically rescales the contribution of each class in the loss according to the inverse of its frequency [35]:

$$w_c = \frac{n}{2\,n_c}, \tag{2.54}$$

where n is the total number of samples and n_c is the number of samples in class $c \in \{0, 1\}$. The logistic loss then penalizes misclassification of minority-class samples more strongly.

2.4 Evaluation

This section describes how the classification pipeline is evaluated using nested cross-validation, detailing how model performance, hyperparameter tuning, and generalization ability were assessed.

2.4.1 Nested Cross-Validation and Model Evaluation

Nested cross-validation (CV) was employed to obtain an unbiased estimate of the generalization performance of the tangent-space logistic regression classifier, directly addressing the research question of whether resting-state EEG can discriminate clinical from non-clinical participants. This procedure separates hyperparameter optimization from model evaluation, preventing the optimistic bias that arises when the same data are used both to tune and to assess a model [9]. The nested structure consists of two loops: an outer loop for performance estimation and an inner loop for hyperparameter optimization.

Cross-validation was performed at the subject level. Covariance matrices from all windows belonging to a subject were first aggregated into a single subject-specific covariance matrix using the

Riemannian mean. The resulting subject-level representations were then assigned to training and test folds. Consequently, no information from a test subject was available during model training, preventing information leakage and ensuring that performance estimates reflect generalization to previously unseen subjects. The following pseudocode summarizes the nested cross-validation procedure and provides an overview of the steps explained in the subsequent sections:

Algorithm 3: Nested Cross-Validation for Tangent-Space Logistic Regression

Input: Subject IDs, condition, random states (seeds) $\in \{1, 7, 21, 42, 123\}$
Output: Seed-level outer scores, confusion matrices, generalization gaps
foreach *seed r in seeds* **do**
  Define inner CV $\leftarrow$ StratifiedKFold($k = 3$, shuffle=True, seed = r);
  Define outer CV $\leftarrow$ StratifiedKFold($k = 5$, shuffle=True, seed = r);
  Initialize inner_scores $\leftarrow$ [ ], outer_scores $\leftarrow$ [ ], cm_total $\leftarrow \mathbf{0}_{2\times2}$;
  foreach (D_{train}, D_{test}) *in outer CV* **do**
    Extract $(X_{\text{train}}, y_{\text{train}})$ and $(X_{\text{test}}, y_{\text{test}})$ from subject-level covariances;
    Initialize pipeline: TangentSpace $\rightarrow$ StandardScaler $\rightarrow$ LogisticRegression;
    Search $C \in \{0.001, 0.01, 0.1, 1.0\}$ and ℓ_1-ratio $\in \{0.2, 0.5, 0.8\}$ via GridSearchCV with inner CV;
    Fit best pipeline on $(X_{\text{train}}, y_{\text{train}})$;
    Predict y_{pred} on X_{test};
    Append balanced_accuracy(y_{test}, y_{pred}) to outer_scores;
    Append best inner CV score to inner_scores;
    cm_total $\leftarrow$ cm_total + confusion_matrix(y_{test}, y_{pred});
  $\bar{s}_{\text{inner}} \leftarrow$ mean(inner_scores), $\bar{s}_{\text{outer}} \leftarrow$ mean(outer_scores), $\sigma_{\text{outer}} \leftarrow$ std(outer_scores);
  Gap $\leftarrow \bar{s}_{\text{inner}} - \bar{s}_{\text{outer}}$;
  Store {outer_scores, cm_total, Gap} for seed r;
return mean and std of $\bar{s}_{\text{outer}}$ across seeds, all seed-level results;

Choice of StratifiedKFold

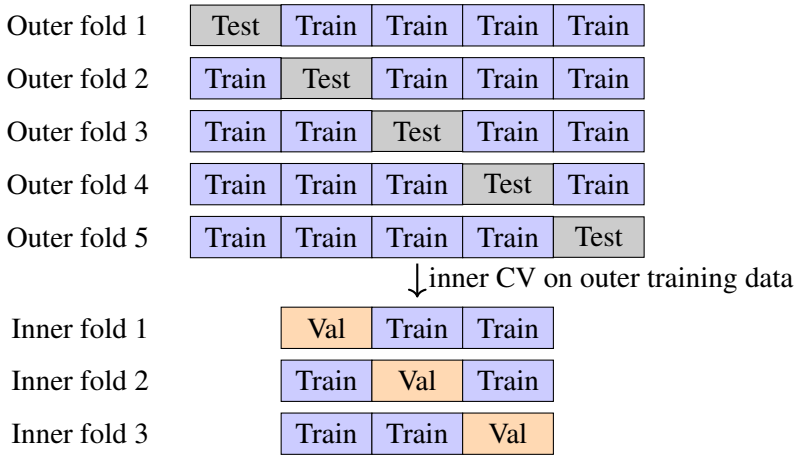
The dataset used in this study is imbalanced, with unequal numbers of clinical and subclinical participants in AD and WY groups. To ensure that each fold contains approximately the same proportion of both classes, stratified sampling was applied. The StratifiedKFold iterator from scikit-learn was chosen because it preserves class ratios in every fold while producing non-overlapping, deterministic partitions [35]. This is important in nested cross-validation, where hyperparameter tuning should be performed using the same outer training and test partitions to ensure a fair comparison between models. In k-fold cross-validation, the dataset is partitioned into k mutually exclusive folds. Each subject appears exactly once in a test fold and k-1 times in the corresponding training folds, providing an exhaustive evaluation in which every subject is tested exactly once [24].

Alternative splitters such as StratifiedShuffleSplit generate random resamples rather than fixed folds [35], which may result in overlapping test sets and prevent every subject from being evaluated exactly once. StratifiedKFold therefore provides a more consistent evaluation framework for subject-level classification.

Outer and Inner Folds

The outer loop consisted of 5 stratified folds, providing five independent test evaluations. In each iteration, four folds were used for training and one for testing. The choice of five outer folds follows common practice in machine-learning research, where 5- or 10-fold cross-validation is recommended as a compromise between bias and variance of the performance estimate while retaining a sufficiently large training set in each fold [24].

The inner loop used 3 stratified folds within the outer training set to tune hyperparameters. Because the inner cross-validation is repeated for every outer training split and every hyperparameter configuration, it accounts for the majority of the computational cost in nested cross-validation [9]. Therefore, three inner folds were chosen to balance reliable hyperparameter selection with computational efficiency.



Inner folds shown for outer fold 1 only; the same structure repeats for all 5 outer folds.

Figure 2.5: Nested cross-validation structure with 5 outer folds and 3 inner folds. The outer test fold (gray) is held out completely during training and hyperparameter tuning. The inner validation fold (orange) operates only on the outer training data to select the best hyperparameters via grid search. After tuning, the model is retrained on the full outer training set and evaluated on the outer test fold.

Shuffle and Random Seeds

The shuffle=True option was enabled in both the inner and outer StratifiedKFold objects to randomize the order of subjects before splitting. This ensures random_state (seed) controls the partitioning reproducibly.

In scikit-learn, many algorithms include a random_state parameter that controls the pseudo-random number generator used for operations such as data shuffling or initialization [35]. Setting this parameter ensures that the same sequence of random operations is reproduced every time the function is executed, providing reproducibility of results. The seed must be an integer in the valid range supported by NumPy's random number generator. This mechanism allows researchers to control

randomness and replicate experiments reliably [35].

In this work, multiple random seeds (1, 7, 21, 42, 123) were used to assess the stability of the nested cross-validation results with respect to different shuffling initializations. Using several seeds allows us to evaluate whether the model's performance is robust to changes in the random partitioning of subjects [39].

Grid Search and Hyperparameter Space

Hyperparameter tuning was performed using GridSearchCV [35], which exhaustively evaluates all combinations of the regularization strength C and the Elastic-Net mixing parameter l1_ratio. The grid was defined as:

$$C \in \{0.001,\ 0.01,\ 0.1,\ 1.0\}, \quad \text{l1\_ratio} \in \{0.2,\ 0.5,\ 0.8\}.$$

Smaller C values correspond to stronger regularization (larger penalty), while larger values allow the model to fit the training data more closely. The parameter l1_ratio controls the balance between ℓ_1 (Lasso) and ℓ_2 (Ridge) regularization: l1_ratio $= 0.2$ emphasizes smoothness, l1_ratio $= 0.8$ encourages sparsity, and l1_ratio $= 0.5$ provides an equal mix (see 2.3.2).

The grid search was parallelized using n_jobs=-1, which utilizes all available CPU cores to accelerate computation. For each hyperparameter configuration, the inner CV computes the mean balanced accuracy across the three folds. The configuration achieving the highest inner-CV score is selected as the optimal set of hyperparameters. After this, the model was retrained on the entire outer training set using these parameters and evaluated on the held-out outer test fold. This ensures that the test data remain completely unseen during both training and tuning.

Balanced Accuracy as the Optimization Metric

Balanced accuracy was used as the scoring metric in both the inner and outer loops because the dataset is imbalanced - the AD group contains predominantly clinical subjects, whereas the WY group contains predominantly non-clinical subjects:

$$\text{fold\_score} = \text{balanced\_accuracy\_score}(y_{\text{test}}, y_{\text{pred}})$$

This metric is defined as the arithmetic mean of sensitivity and specificity and is particularly suitable for imbalanced classification problems [8]. In scikit-learn it is implemented as [35]:

$$\text{Balanced Accuracy} = \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right), \tag{2.55}$$

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. This metric penalizes models that favor the majority class and provides a fairer assessment of performance across clinical and subclinical groups.

Performance Metrics and Aggregation

For each seed, a full 5-fold outer cross-validation was performed. For each outer fold, using scikit-learn, a confusion matrix was computed based on the predicted and true labels [35]:

$$\texttt{cm} = \texttt{confusion\_matrix}(y_{\text{test}}, y_{\text{pred}}, \texttt{labels} = [0, 1]).$$

$$\text{CM} = \begin{bmatrix} TN & FP \\ FN & TP \end{bmatrix}.$$

Within each seed, these matrices were summed in all five outer folds to obtain an aggregated confusion matrix, providing a comprehensive summary of classification performance across all subjects. This aggregated matrix is used to report total true positives, false positives, true negatives, and false negatives for that seed. Because each subject appears exactly once in an outer test fold, summing the confusion matrices across folds yields the same total counts that would be obtained by evaluating all outer-fold predictions together.

For each fold, the best inner-CV balanced accuracy and the outer-fold balanced accuracy were recorded. After all folds were completed, the mean and standard deviation of the outer scores were computed for each seed:

$$\text{Outer CV mean} = \overline{s}_{\text{outer}}, \quad \text{Outer CV std} = \sigma_{\text{outer}}.$$

The *generalization gap* was defined as the difference between the mean inner-CV balanced accuracy and the mean outer-CV balanced accuracy for each seed:

$$\text{Generalization Gap} = \overline{s}_{\text{inner}} - \overline{s}_{\text{outer}}.$$

A small gap indicates good generalization, while a large gap suggests overfitting to the inner validation folds. After all seeds were evaluated, the outer-CV means were averaged across seeds to obtain the final performance estimate, providing a stable summary robust to the choice of random partition.

2.4.2 Permutation Test

To assess whether the classification performance obtained by the nested cross-validation exceeded chance level, a permutation test was conducted for both the eyes-closed and eyes-open conditions in AD and WY groups. Permutation tests are commonly used statistical tools [38].

For each permutation, the class labels were randomly shuffled while keeping the feature matrix fixed, and a 5-fold stratified cross-validation was performed using fixed hyperparameters ($C = 1.0$, $\texttt{l1\_ratio} = 0.5$), yielding a permuted balanced accuracy score. This process was repeated $n = 1000$ times to construct an empirical null distribution of balanced accuracy under the null hypothesis that there is no relationship between the EEG features and the clinical labels. A fixed random seed ($\texttt{random\_state} = 42$) was used both for the stratified fold splitting and for generating the label permutations, ensuring reproducibility of the null distribution.

The observed balanced accuracy from the nested CV was then compared against this null distribution. The p-value was computed as [38]:

$$p = \frac{\#\{\text{perm scores} \geq \text{observed score}\} + 1}{n_{\text{perm}} + 1},$$

where the $+1$ correction ensures the p-value is never zero and accounts for the observed score itself. A significance threshold of $\alpha = 0.05$ was applied, meaning that a result was considered statistically significant if the probability of observing it by chance alone was less than 5%. This would indicate that the model performs above chance, whereas a non-significant result would suggest that the observed performance is consistent with random label assignment.

Note that fixed hyperparameters were used inside the permutation test rather than full nested CV, as running 1000 such iterations would be computationally prohibitive. The following pseudocode demonstrates this approach:

Algorithm 4: Permutation Test for Balanced Accuracy

Input: Subject IDs, condition, $n_{\text{perm}} = 1000$, random_state $= 42$
Output: real_score, perm_scores, p_value
Extract subject-level features (X, y) for the given condition;
Define CV $\leftarrow$ StratifiedKFold($k = 5$, shuffle=True, random_state $= 42$);
Initialize pipeline: TangentSpace $\rightarrow$ StandardScaler $\rightarrow$ LogisticRegression ($C = 1.0$, ℓ_1-ratio $= 0.5$);
Compute real_score as the mean balanced accuracy of CV on (X, y);
Initialize empty list perm_scores;
for $i \leftarrow 1$ **to** n_{perm} **do**
    Randomly permute labels: $y_{\text{perm}} \leftarrow$ permutation of y;
    Compute mean CV balanced accuracy on $(X,\ y_{\text{perm}})$ and append to perm_scores;
Estimate p_value by comparing real_score to the distribution of perm_scores;
return real_score, perm_scores, p_value;

3 Results

This chapter reports the quantitative results obtained from the nested cross-validation procedure applied to the eyes-open and eyes-closed resting-state EEG data. All reported classification results refer to the binary prediction of whether a participant has no mental disorder or at least one mental disorder, based on the resting-state EEG. Participants in the AD group ranged in age from 18 to 63 years, whereas participants in the WY group ranged from 18 to 30 years. Nested cross-validation was performed separately for the AD (mostly clinical patients with at least one diagnosed condition) and WY (mostly subclinical) groups, across 5 random seeds. To reduce computational overhead, all extracted features were stored in a folder and reused rather than recomputed each time.

All reported numerical results correspond to a single full execution of the pipeline. Due to the use of randomized cross-validation splits and stochastic optimization, exact numerical values may vary slightly across runs, even when random seeds are fixed. However, the overall performance patterns and conclusions remain stable.

Across all tasks and conditions, a total of **3600 models** were trained during the nested cross-validation procedure (5 random seeds $\times$ 5 outer folds $\times$ 3 inner folds $\times$ 12 hyperparameter combinations per condition). The dataset comprised **146 AD subjects** and **149 WY subjects**. Each subject contributed multiple covariance windows per condition. The number of windows varied depending on the recording length, typically ranging between **12 and 18 windows per subject** (mean $\approx$ 15). These windows were used to compute subject-level covariance matrices for both eyes-open and eyes-closed conditions.

During cross-validation, each outer fold contained approximately **116-120 training subjects** and **29-30 test subjects**, depending on the task and condition. Table 3.1 summarizes the subject distribution per fold.

Table 3.1: Approximate number of subjects per fold.

Task	**Train Subjects**	**Test Subjects**
AD (open/closed)	116–117	29–30
WY (open/closed)	119–120	29–30

Table 3.2: Across-seed summary of nested CV results.

Task	Condition	Outer Mean	Outer Std	Inner Mean	Gen. Gap
AD	Open	0.499	0.033	0.538	0.039
WY	Open	0.513	0.050	0.558	0.044
AD	Closed	0.469	0.038	0.532	0.063
WY	Closed	0.507	0.028	0.569	0.062

3.0.1 AD - Eyes-Open Condition

Table 3.3 summarizes the across-seed performance for the AD eyes-open condition, reporting the mean and standard deviation of the outer-fold balanced accuracy across the five random seeds, the mean inner-CV accuracy, and the mean generalization gap.

Table 3.3: AD (eyes-open): across-seed summary of nested CV results.

Outer Mean	Outer Std	Inner Mean	Gen. Gap
0.499	0.033	0.538	0.039

The mean outer-CV balanced accuracy across all seeds was 0.499, with a standard deviation of 0.033 across seeds. The mean inner-CV balanced accuracy was 0.538, yielding a mean generalization gap of 0.039.

Table 3.4 reports the same metrics broken down by individual seed.

Table 3.4: AD (eyes-open): per-seed nested CV results.

Seed	Outer Mean	Outer Std	Inner Mean	Gen. Gap
1	0.514	0.081	0.535	0.021
7	0.498	0.047	0.530	0.032
21	0.554	0.102	0.541	-0.013
42	0.464	0.087	0.530	0.066
123	0.466	0.099	0.553	0.087

Outer-CV means ranged from 0.464 (seed 42) to 0.554 (seed 21), reflecting moderate variability across random partitions. The generalization gap was positive for four out of five seeds, with seed 21 yielding a slightly negative gap of -0.013, indicating that for this particular partition the outer performance marginally exceeded the inner-CV estimate. Generalization gaps ranged from -0.013 to 0.087.

Table 3.5 reports the mean balanced accuracy per outer fold, averaged across all five seeds, and is visualized in Figure 3.1.

Table 3.5: AD (eyes-open): mean balanced accuracy per outer fold (averaged across seeds).

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
0.503	0.524	0.403	0.601	0.449

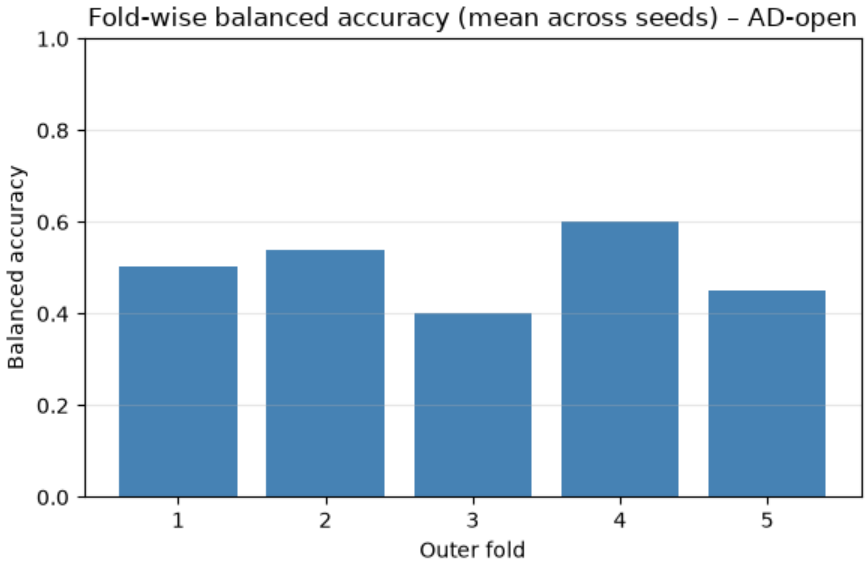


Figure 3.1: AD eyes-open: fold-wise balanced accuracy averaged across five random seeds.

Fold-wise performance was heterogeneous, ranging from 0.403 in fold 3 to 0.601 in fold 4. Table 3.6 reports the best hyperparameters selected by the inner grid search for each seed and outer fold.

Table 3.6: AD (eyes-open): best hyperparameters (C, ℓ_1-ratio) per seed and outer fold.

Seed	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1	(1.0, 0.2)	(1.0, 0.8)	(1.0, 0.8)	(0.01, 0.2)	(0.001, 0.2)
7	(0.1, 0.5)	(0.001, 0.2)	(1.0, 0.8)	(1.0, 0.5)	(1.0, 0.5)
21	(1.0, 0.8)	(0.1, 0.2)	(0.001, 0.2)	(1.0, 0.2)	(0.1, 0.8)
42	(0.001, 0.2)	(0.01, 0.2)	(0.1, 0.2)	(0.001, 0.2)	(1.0, 0.2)
123	(1.0, 0.2)	(0.1, 0.8)	(0.1, 0.5)	(0.1, 0.5)	(1.0, 0.2)

Table 3.7 reports the outer-fold balanced accuracy for each seed and fold individually.

Table 3.7: AD (eyes-open): outer-fold balanced accuracy per seed and fold.

Seed	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1	0.612	0.533	0.370	0.558	0.500
7	0.506	0.500	0.471	0.578	0.435
21	0.447	0.620	0.500	0.721	0.481
42	0.500	0.580	0.326	0.500	0.412
123	0.447	0.471	0.348	0.646	0.419

Individual fold scores ranged from 0.326 (seed 42, fold 3) to 0.721 (seed 21, fold 4).
The aggregated confusion matrix, summed across all five seeds and all five outer folds, is shown in Figure 3.2 and reported numerically in Table 3.8.

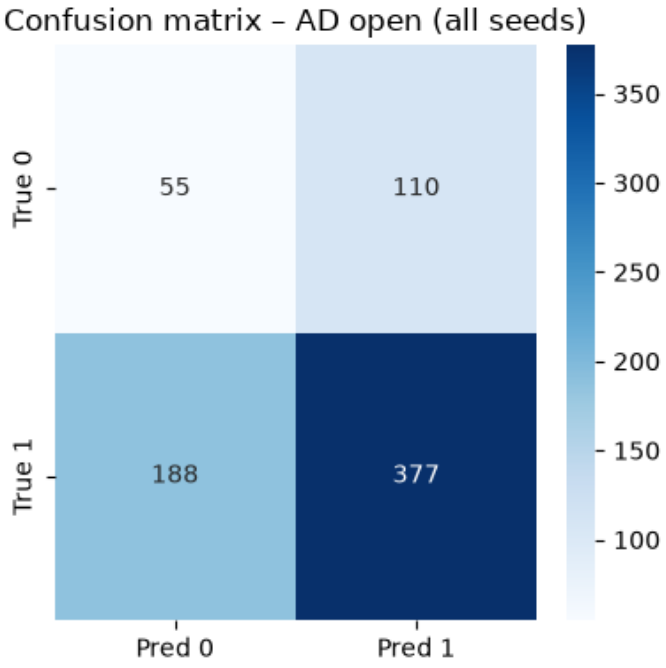


Figure 3.2: AD eyes-open: confusion matrix aggregated across all seeds and outer folds. Rows represent true labels and columns represent predicted labels.

Table 3.8: AD (eyes-open): aggregated confusion matrix per seed and overall total.

Seed	TN	FP	FN	TP
1	18	15	56	57
7	7	26	24	89
21	17	16	48	65
42	3	30	19	94
123	10	23	41	72
Total	55	110	188	377

Across all seeds and folds combined, the model predicted class 1 (clinical) for the majority of the test subjects. Out of 730 total subject-level predictions (146 subjects $\times$ 5 seeds), 377 true positives and 188 false negatives were recorded, while only 55 true negatives and 110 false positives were obtained.

The predicted probability distributions for both classes are shown in Figure 3.3. The x-axis extends beyond the nominal $[0, 1]$ range, reaching approximately -0.2 to 1.2. This occurs because the distributions are estimated using kernel density estimation (KDE) [35], which applies a smooth kernel function at each data point. Near the boundaries of 0 and 1, the kernel extends slightly beyond these limits, producing non-zero density outside the valid probability range, but does not imply that the classifier produced probabilities outside of the valid range $[0, 1]$.

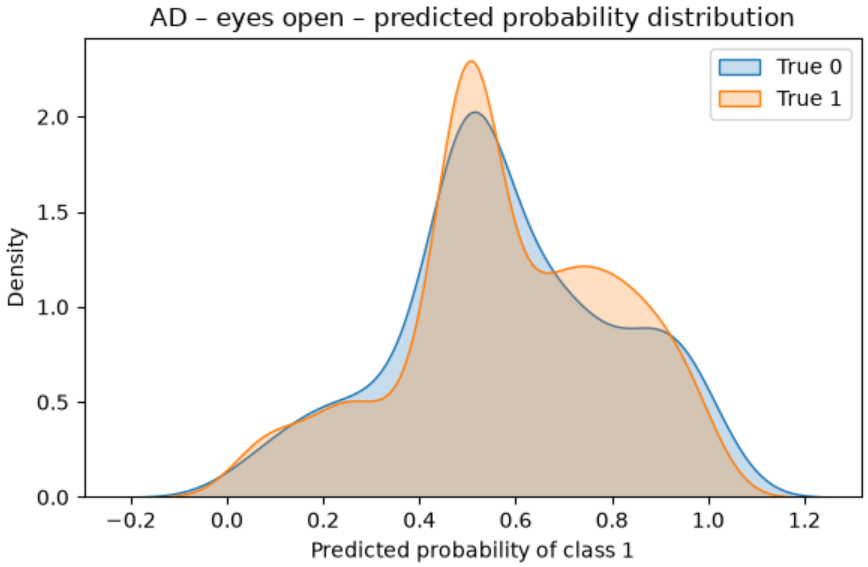


Figure 3.3: AD eyes-open: distribution of predicted probabilities of class 1 for true class 0 (subclinical, blue) and true class 1 (clinical, orange), aggregated across all seeds and outer folds. The x-axis extends slightly beyond $[0, 1]$ due to kernel density estimation boundary effects.

Both distributions are concentrated around 0.5, with substantial overlap between true class 0 and true class 1. The true class 1 distribution shows a slightly heavier tail toward higher probabilities,

while both distributions peak near 0.5, consistent with the near-chance classification performance observed in the outer CV scores. This is further supported by the permutation test, which yielded a real balanced accuracy of 0.51 and a p-value of 0.30, indicating that the observed performance does not significantly exceed chance level ($\alpha = 0.05$).

3.0.2 WY - Eyes-Open Condition

Table 3.9 summarizes the across-seed performance for the WY eyes-open condition, reporting the mean and standard deviation of the outer-fold balanced accuracy across the five random seeds, the mean inner-CV accuracy, and the mean generalization gap.

Table 3.9: WY (eyes-open): across-seed summary of nested CV results.

Outer Mean	Outer Std	Inner Mean	Gen. Gap
0.513	0.050	0.558	0.044

The mean outer-CV balanced accuracy across all seeds was 0.513, with a standard deviation of 0.050 across seeds. The mean inner-CV balanced accuracy was 0.558, yielding a mean generalization gap of 0.044.

Table 3.10 reports the same metrics broken down by individual seed.

Table 3.10: WY (eyes-open): per-seed nested CV results.

Seed	Outer Mean	Outer Std	Inner Mean	Gen. Gap
1	0.508	0.056	0.532	0.024
7	0.480	0.055	0.600	0.120
21	0.523	0.139	0.529	0.006
42	0.602	0.087	0.571	-0.031
123	0.454	0.077	0.556	0.102

Outer-CV means ranged from 0.454 (seed 123) to 0.602 (seed 42), reflecting substantial variability across random partitions - notably larger than that observed in the AD eyes-open condition. The generalization gap was positive for four out of five seeds, with seed 42 yielding a slightly negative gap of -0.031, indicating that for this particular partition the outer performance marginally exceeded the inner-CV estimate. Generalization gaps ranged from -0.031 to 0.120, with seed 7 showing the largest gap of 0.120.

Table 3.11 reports the mean balanced accuracy per outer fold, averaged across all five seeds, and is visualized in Figure 3.4.

Table 3.11: WY (eyes-open): mean balanced accuracy per outer fold (averaged across seeds).

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
0.525	0.469	0.498	0.521	0.552

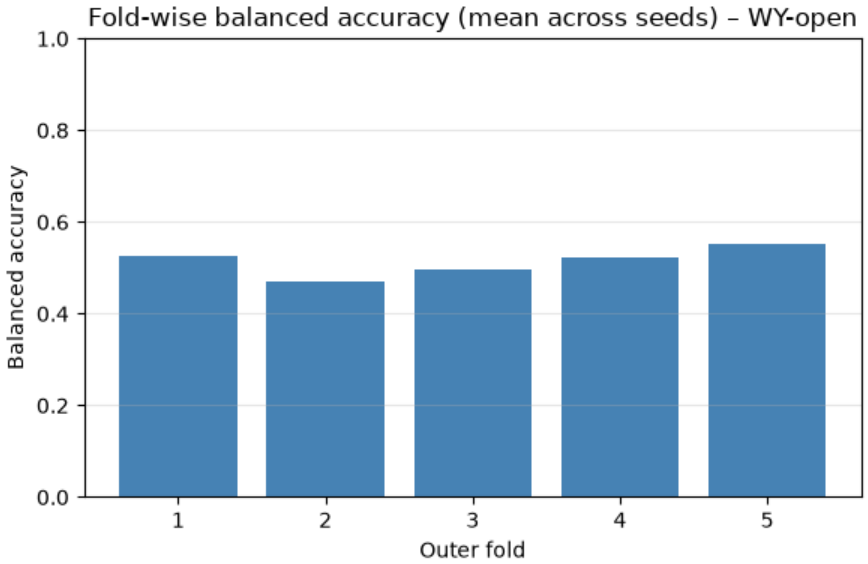


Figure 3.4: WY eyes-open: fold-wise balanced accuracy averaged across five random seeds.

Fold-wise performance was more homogeneous than in the AD eyes-open condition, ranging from 0.469 in fold 2 to 0.552 in fold 5, with all folds remaining close to chance level.

Table 3.12 reports the best hyperparameters selected by the inner grid search for each seed and outer fold.

Table 3.12: WY (eyes-open): best hyperparameters (C, ℓ_1-ratio) per seed and outer fold.

Seed	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1	(0.01, 0.2)	(0.001, 0.2)	(0.001, 0.2)	(0.01, 0.2)	(1.0, 0.2)
7	(0.1, 0.8)	(1.0, 0.2)	(1.0, 0.5)	(0.1, 0.8)	(0.1, 0.2)
21	(1.0, 0.5)	(0.1, 0.5)	(1.0, 0.2)	(0.1, 0.2)	(0.1, 0.5)
42	(1.0, 0.8)	(1.0, 0.5)	(0.1, 0.2)	(1.0, 0.5)	(0.01, 0.2)
123	(1.0, 0.2)	(0.1, 0.8)	(0.1, 0.2)	(0.01, 0.2)	(1.0, 0.2)

Table 3.13 reports the outer-fold balanced accuracy for each seed and fold individually.

Table 3.13: WY (eyes-open): outer-fold balanced accuracy per seed and fold.

Seed	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1	0.463	0.500	0.500	0.615	0.462
7	0.537	0.471	0.404	0.548	0.442
21	0.426	0.462	0.510	0.423	0.795
42	0.667	0.510	0.567	0.529	0.737
123	0.537	0.404	0.510	0.490	0.327

Individual fold scores ranged from 0.327 (seed 123, fold 5) to 0.795 (seed 21, fold 5), indicating substantial within-seed variability. Notably, seed 21 fold 5 and seed 42 fold 5 both produced markedly higher scores of 0.795 and 0.737 respectively, while the remaining folds for those seeds were considerably lower.

The aggregated confusion matrix, summed across all five seeds and all five outer folds, is shown in Figure 3.5 and reported numerically in Table 3.14.

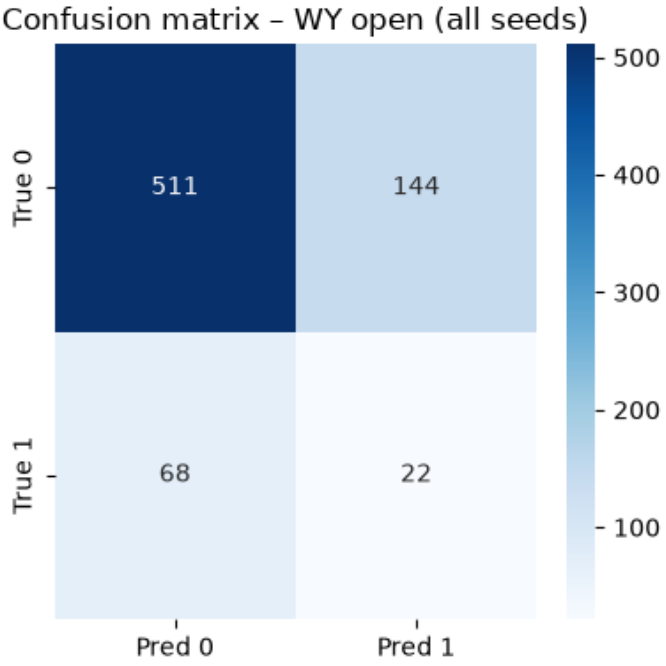


Figure 3.5: WY eyes-open: confusion matrix aggregated across all seeds and outer folds. Rows represent true labels and columns represent predicted labels.

Table 3.14: WY (eyes-open): aggregated confusion matrix per seed and overall total.

Seed	TN	FP	FN	TP
1	85	46	11	7
7	104	27	15	3
21	113	18	15	3
42	112	19	12	6
123	97	34	15	3
Total	511	144	68	22

Across all seeds and folds combined, out of 745 total subject-level predictions (149 subjects $\times$ 5 seeds), the model predicted class 0 (subclinical) for the vast majority of subjects, yielding 511 true negatives and 144 false positives, while producing only 22 true positives and 68 false negatives.

The predicted probability distributions for both classes are shown in Figure 3.6. As in the AD eyes-open condition, the x-axis extends beyond the nominal $[0, 1]$ range due to kernel density estimation boundary effects.

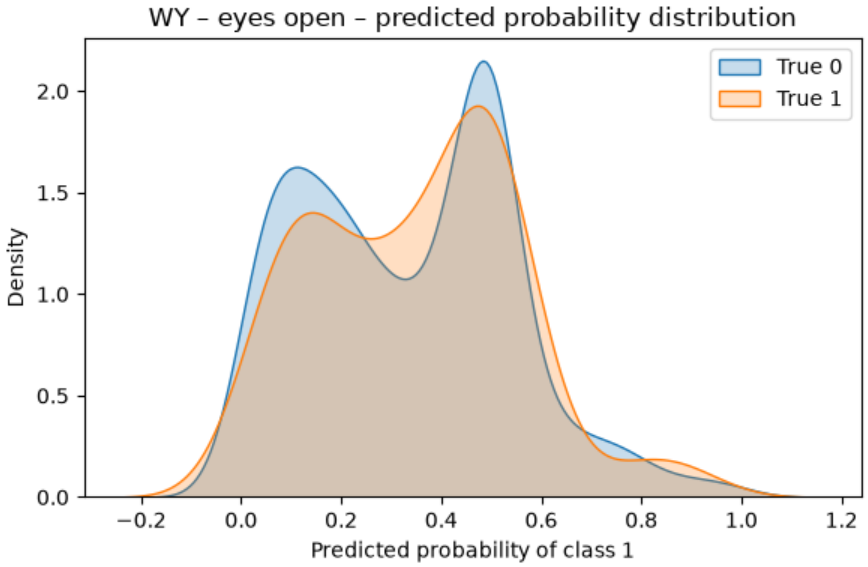


Figure 3.6: WY eyes-open: distribution of predicted probabilities of class 1 for true class 0 (subclinical, blue) and true class 1 (clinical, orange), aggregated across all seeds and outer folds. The x-axis extends slightly beyond $[0, 1]$ due to kernel density estimation boundary effects.

Both distributions show a bimodal shape, with one peak near 0.1 and a second peak near 0.5. The true class 0 distribution has a more pronounced peak near 0.1, while the true class 1 distribution is shifted slightly toward higher probabilities with a stronger second peak near 0.5. Despite this slight separation, the two distributions overlap substantially across the entire probability range. The

permutation test yielded a real balanced accuracy of 0.54 and a p-value of 0.55, indicating that the observed performance does not significantly exceed chance level ($\alpha = 0.05$).

3.0.3 AD - Eyes-Closed Condition

Table 3.15 summarizes the across-seed performance for the AD eyes-closed condition, reporting the mean and standard deviation of the outer-fold balanced accuracy across the five random seeds, the mean inner-CV accuracy, and the mean generalization gap.

Table 3.15: AD (eyes-closed): across-seed summary of nested CV results.

Outer Mean	Outer Std	Inner Mean	Gen. Gap
0.469	0.038	0.532	0.063

The mean outer-CV balanced accuracy across all seeds was 0.469, with a standard deviation of 0.038 across seeds. The mean inner-CV balanced accuracy was 0.532, yielding a mean generalization gap of 0.063.

Table 3.16 reports the same metrics broken down by individual seed.

Table 3.16: AD (eyes-closed): per-seed nested CV results.

Seed	Outer Mean	Outer Std	Inner Mean	Gen. Gap
1	0.424	0.071	0.533	0.109
7	0.513	0.058	0.535	0.022
21	0.510	0.045	0.541	0.030
42	0.429	0.088	0.515	0.086
123	0.468	0.038	0.537	0.070

Outer-CV means ranged from 0.424 (seed 1) to 0.513 (seed 7). The generalization gap was positive for all five seeds, ranging from 0.022 to 0.109, with seed 1 showing the largest gap of 0.109.

Table 3.17 reports the mean balanced accuracy per outer fold, averaged across all five seeds, and is visualized in Figure 3.7.

Table 3.17: AD (eyes-closed): mean balanced accuracy per outer fold (averaged across seeds).

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
0.437	0.430	0.491	0.498	0.487

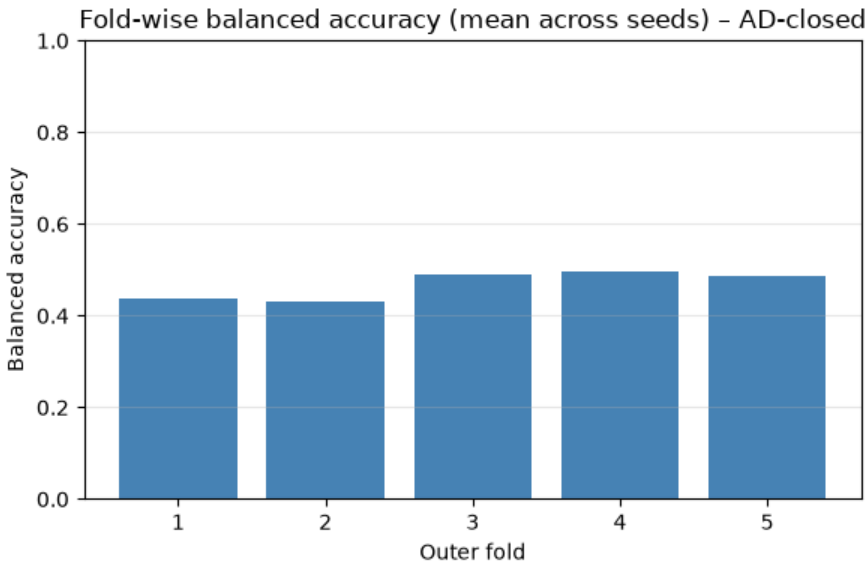


Figure 3.7: AD eyes-closed: fold-wise balanced accuracy averaged across five random seeds.

Fold-wise performance ranged from 0.430 in fold 2 to 0.498 in fold 4, with all folds remaining below chance level.

Table 3.18 reports the best hyperparameters selected by the inner grid search for each seed and outer fold.

Table 3.18: AD (eyes-closed): best hyperparameters (C, ℓ_1-ratio) per seed and outer fold.

Seed	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1	(0.001, 0.2)	(0.1, 0.8)	(0.1, 0.8)	(1.0, 0.2)	(0.001, 0.2)
7	(0.1, 0.5)	(0.001, 0.2)	(0.1, 0.8)	(1.0, 0.2)	(0.01, 0.2)
21	(0.001, 0.2)	(1.0, 0.2)	(0.1, 0.5)	(0.1, 0.8)	(0.1, 0.8)
42	(0.1, 0.8)	(0.01, 0.2)	(0.1, 0.8)	(0.001, 0.2)	(1.0, 0.5)
123	(1.0, 0.2)	(0.1, 0.2)	(0.001, 0.2)	(0.1, 0.2)	(0.1, 0.8)

Table 3.19 reports the outer-fold balanced accuracy for each seed and fold individually.

Table 3.19: AD (eyes-closed): outer-fold balanced accuracy per seed and fold.

Seed	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1	0.500	0.366	0.431	0.321	0.500
7	0.484	0.500	0.493	0.627	0.461
21	0.500	0.449	0.536	0.581	0.484
42	0.304	0.341	0.493	0.500	0.506
123	0.398	0.496	0.500	0.461	0.484

Individual fold scores ranged from 0.304 (seed 42, fold 1) to 0.627 (seed 7, fold 4), indicating substantial within-seed variability across folds.

The aggregated confusion matrix, summed across all five seeds and all five outer folds, is shown in Figure 3.8 and reported numerically in Table 3.20.

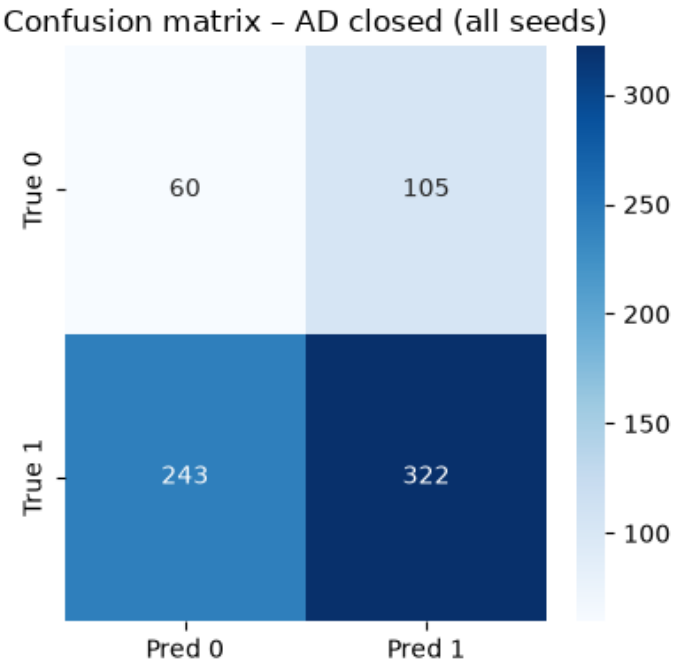


Figure 3.8: AD eyes-closed: confusion matrix aggregated across all seeds and outer folds. Rows represent true labels and columns represent predicted labels.

Table 3.20: AD (eyes-closed): aggregated confusion matrix per seed and overall total.

Seed	TN	FP	FN	TP
1	10	23	50	63
7	15	18	50	63
21	17	16	55	58
42	6	27	38	75
123	12	21	50	63
Total	60	105	243	322

Across all seeds and folds combined, out of 730 total subject-level predictions (146 subjects × 5 seeds), the model predicted class 1 (clinical) for the majority of subjects, yielding 322 true positives and 243 false negatives, while producing only 60 true negatives and 105 false positives.

The predicted probability distributions for both classes are shown in Figure 3.9. As in the previous conditions, the x-axis extends beyond the nominal $[0, 1]$ range due to kernel density estimation boundary effects.

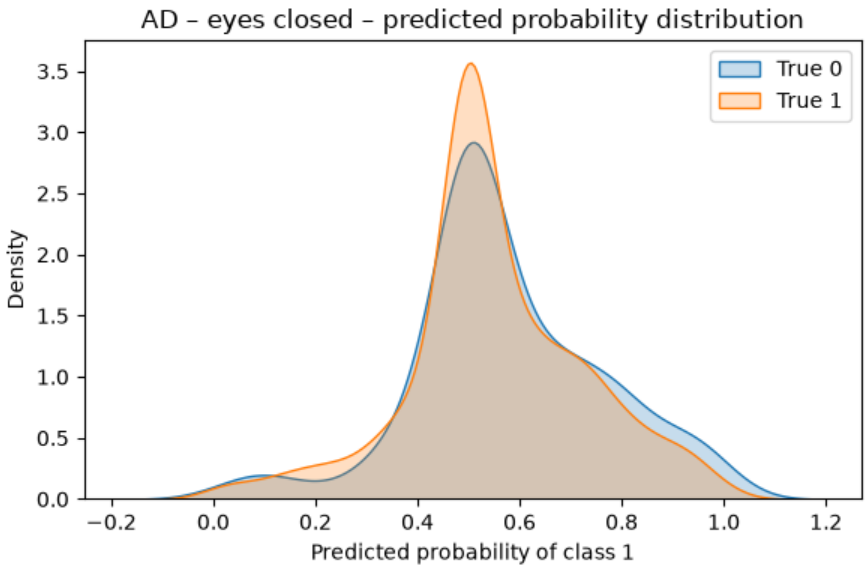


Figure 3.9: AD eyes-closed: distribution of predicted probabilities of class 1 for true class 0 (subclinical, blue) and true class 1 (clinical, orange), aggregated across all seeds and outer folds. The x-axis extends slightly beyond $[0, 1]$ due to kernel density estimation boundary effects.

Both distributions show a sharp peak near 0.5, with the true class 1 distribution peaking slightly higher and showing a heavier right tail compared to true class 0. The two distributions overlap substantially across the entire probability range. The permutation test yielded a real balanced accuracy of 0.46 and a p-value of 0.75, indicating that the observed performance does not significantly exceed chance level ($\alpha = 0.05$).

3.0.4 WY - Eyes-Closed Condition

Table 3.21 summarizes the across-seed performance for the WY eyes-closed condition, reporting the mean and standard deviation of the outer-fold balanced accuracy across the five random seeds, the mean inner-CV accuracy, and the mean generalization gap.

Table 3.21: WY (eyes-closed): across-seed summary of nested CV results.

Outer Mean	Outer Std	Inner Mean	Gen. Gap
0.507	0.028	0.569	0.062

The mean outer-CV balanced accuracy across all seeds was 0.507, with a standard deviation of 0.028 across seeds. The mean inner-CV balanced accuracy was 0.569, yielding a mean generalization gap

of 0.062.

Table 3.22 reports the same metrics broken down by individual seed.

Table 3.22: WY (eyes-closed): per-seed nested CV results.

Seed	Outer Mean	Outer Std	Inner Mean	Gen. Gap
1	0.485	0.019	0.547	0.063
7	0.469	0.080	0.568	0.100
21	0.550	0.097	0.555	0.005
42	0.512	0.093	0.587	0.075
123	0.519	0.074	0.587	0.069

Outer-CV means ranged from 0.469 (seed 7) to 0.550 (seed 21). The generalization gap was positive for all five seeds, ranging from 0.005 to 0.100, with seed 7 showing the largest gap of 0.100.

Table 3.23 reports the mean balanced accuracy per outer fold, averaged across all five seeds, and is visualized in Figure 3.10.

Table 3.23: WY (eyes-closed): mean balanced accuracy per outer fold (averaged across seeds).

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
0.541	0.492	0.494	0.532	0.475

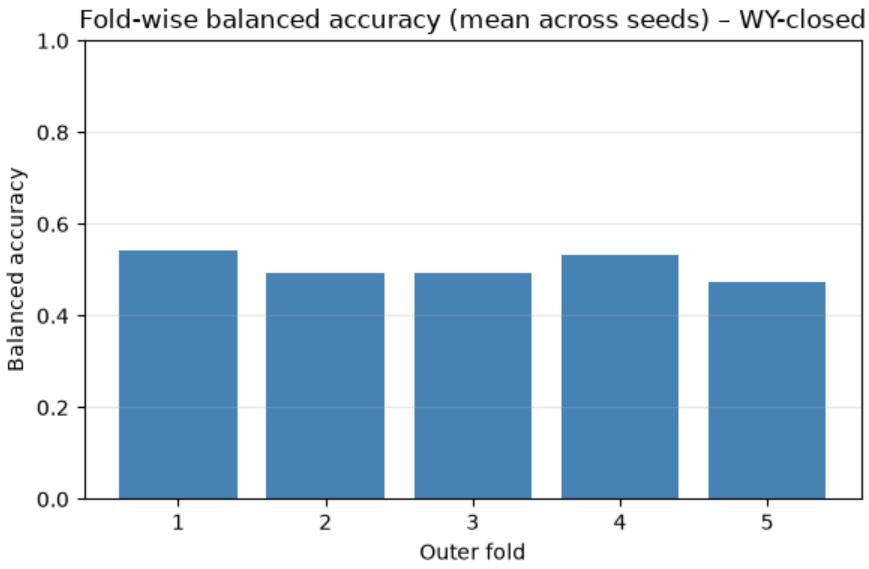


Figure 3.10: WY eyes-closed: fold-wise balanced accuracy averaged across five random seeds.

Fold-wise performance ranged from 0.475 in fold 5 to 0.541 in fold 1, with all folds remaining close to chance level.

Table 3.24 reports the best hyperparameters selected by the inner grid search for each seed and outer fold.

Table 3.24: WY (eyes-closed): best hyperparameters (C, ℓ_1-ratio) per seed and outer fold.

Seed	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1	(1.0, 0.2)	(0.001, 0.2)	(0.001, 0.2)	(0.1, 0.8)	(0.1, 0.2)
7	(1.0, 0.5)	(0.01, 0.2)	(0.1, 0.8)	(0.01, 0.2)	(0.1, 0.2)
21	(0.1, 0.5)	(1.0, 0.8)	(0.01, 0.2)	(1.0, 0.8)	(1.0, 0.2)
42	(0.1, 0.2)	(1.0, 0.5)	(0.01, 0.2)	(1.0, 0.8)	(1.0, 0.2)
123	(1.0, 0.2)	(1.0, 0.2)	(0.01, 0.2)	(1.0, 0.8)	(1.0, 0.2)

Table 3.25 reports the outer-fold balanced accuracy for each seed and fold individually.

Table 3.25: WY (eyes-closed): outer-fold balanced accuracy per seed and fold.

Seed	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1	0.500	0.500	0.500	0.462	0.462
7	0.593	0.356	0.510	0.423	0.462
21	0.426	0.548	0.471	0.606	0.699
42	0.630	0.548	0.452	0.567	0.365
123	0.556	0.510	0.538	0.606	0.385

Individual fold scores ranged from 0.356 (seed 7, fold 2) to 0.699 (seed 21, fold 5), indicating substantial within-seed variability across folds.

The aggregated confusion matrix, summed across all five seeds and all five outer folds, is shown in Figure 3.11 and reported numerically in Table 3.26.

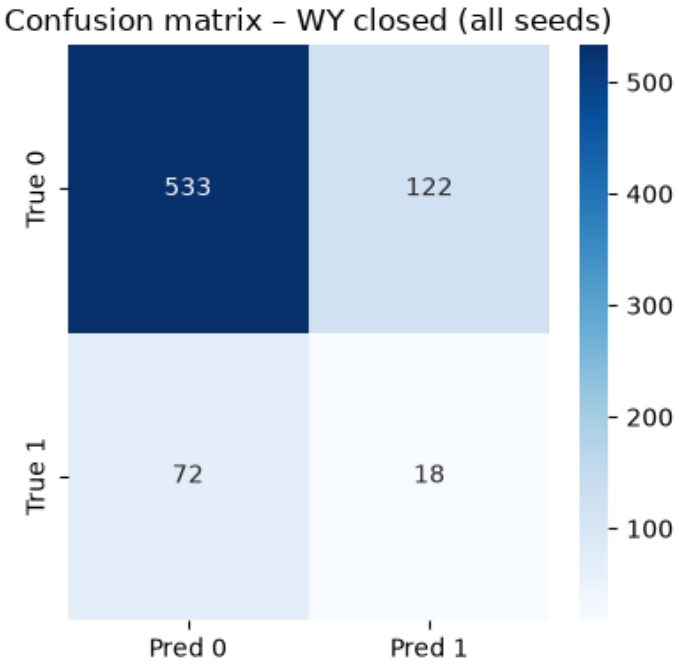


Figure 3.11: WY eyes-closed: confusion matrix aggregated across all seeds and outer folds. Rows represent true labels and columns represent predicted labels.

Table 3.26: WY (eyes-closed): aggregated confusion matrix per seed and overall total.

Seed	**TN**	**FP**	**FN**	**TP**
1	118	13	17	1
7	101	30	15	3
21	107	24	13	5
42	106	25	14	4
123	101	30	13	5
Total	533	122	72	18

Across all seeds and folds combined, out of 745 total subject-level predictions (149 subjects $\times$ 5 seeds), the model predicted class 0 (subclinical) for the vast majority of subjects, yielding 533 true negatives and 122 false positives, while producing only 18 true positives and 72 false negatives.

The predicted probability distributions for both classes are shown in Figure 3.12. As in the previous conditions, the x-axis extends beyond the nominal $[0, 1]$ range due to kernel density estimation boundary effects.

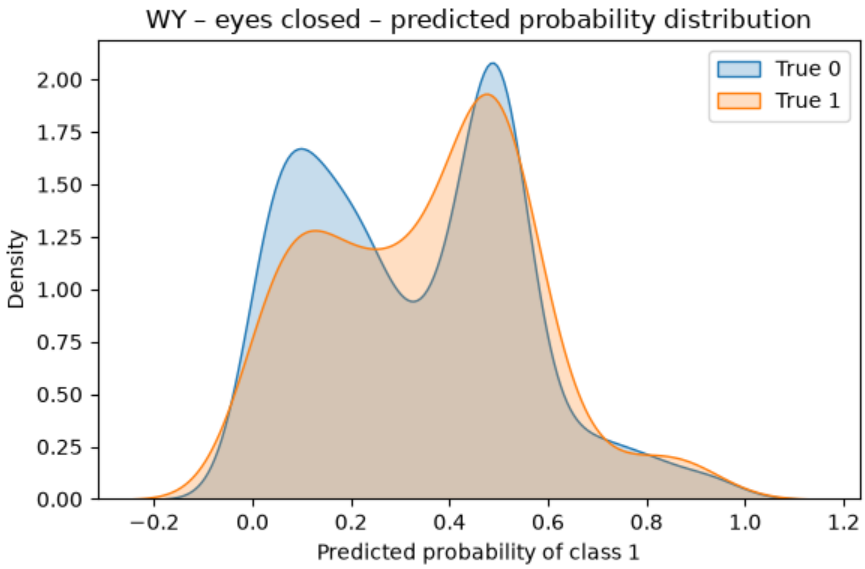


Figure 3.12: WY eyes-closed: distribution of predicted probabilities of class 1 for true class 0 (subclinical, blue) and true class 1 (clinical, orange), aggregated across all seeds and outer folds. The x-axis extends slightly beyond $[0, 1]$ due to kernel density estimation boundary effects.

Both distributions show a bimodal shape with peaks near 0.1 and 0.5, closely resembling the pattern observed in the WY eyes-open condition. The true class 0 distribution has a more pronounced first peak near 0.1, while the true class 1 distribution is shifted slightly toward higher probabilities with a stronger second peak near 0.5. The two distributions overlap substantially. The permutation test yielded a real balanced accuracy of 0.53 and a p-value of 0.30, indicating that the observed performance does not significantly exceed chance level ($\alpha = 0.05$).

4 Discussion

4.1 Overall Performance Interpretation

The primary finding of this thesis is that the covariance-based Riemannian classification pipeline did not achieve reliable predictive performance for either of the two investigated classification tasks. Mean outer-fold balanced accuracy values ranged from 0.469 (AD eyes-closed) to 0.513 (WY eyes-open) across all four conditions, indicating that the classifier performed at chance level throughout. All four permutation tests yielded p-values well above the significance threshold of $\alpha = 0.05$ (ranging from $p = 0.30$ to $p = 0.75$), confirming that none of the observed performances exceeded what would be expected under random label assignment.

This result suggests that the covariance structure of resting-state EEG, as captured by 4-second windowed OAS-regularized covariance matrices and aggregated via the Riemannian mean, does not carry sufficient discriminative information to reliably separate participants with at least one diagnosed mental disorder from those without one. Covariance matrices represent second-order statistical relationships between EEG channels, capturing inter-electrode synchrony and spatial relationships between channels [12]. While these representations have proven highly effective in paradigms with strong and reproducible task-related neural activity, the classification task investigated here may present a fundamentally different challenge: the signal differences between the clinical and subclinical groups may be too weak relative to the large within-class variability inherent to resting-state EEG [21].

An important characteristic of the present classification problem that may have contributed to the near-chance results is the heterogeneity of the clinical group. The positive class (clinical) comprised participants diagnosed with a broad range of mental disorders, including generalized anxiety disorder, social anxiety disorder, obsessive-compulsive disorder, specific phobia, panic disorder, PTSD, depressive disorders, and several others. This diagnostic heterogeneity means that the clinical group likely encompasses substantially different neural profiles, making it unlikely that a single decision boundary in tangent space can reliably separate all clinical subjects from subclinical ones. As noted by Uyanik et al. [45], anxiety detection from EEG signals is considerably more difficult than detection of neurological disorders such as epilepsy, precisely because there are no obvious amplitude or frequency differences between anxious and non-anxious resting-state EEG. This observation aligns directly with the near-chance performance observed across all conditions in the present work.

Another factor that may have contributed to the near-chance performance is age-related variability in resting-state EEG. Barry et al. [7] demonstrated that resting-state EEG differs substantially between young and older healthy adults, with age-related reductions in delta, theta, and alpha power and increases in beta activity. Importantly, the AD group exhibited a substantially broader age range

(18-63 years) than the WY group (18-30 years), which likely amplified within-class variability in the mostly clinical group and further reduced the separability of the two classes in the covariance feature space.

Furthermore, both the predicted probability distributions and the confusion matrices provide consistent evidence of limited discriminability. In all four conditions, the predicted probability distributions of the two classes overlapped substantially, with both distributions peaking near 0.5. This indicates that the classifier assigned similar probabilities to both classes, reflecting uncertainty rather than meaningful class separation. The confusion matrices further revealed systematic prediction biases: in the AD group, the model showed a strong tendency to predict the clinical class for most subjects regardless of true label, while in the WY group, the model predominantly predicted the subclinical class. These opposing biases suggest that the classifiers largely reflected the class imbalance of each group - the AD group contained predominantly clinical participants, whereas the WY group contained predominantly subclinical ones - rather than learning generalizable class-specific covariance patterns.

The variability of results across random seeds further underscores the instability of the observed performance. Within individual conditions, outer-CV means varied substantially depending on the random partition of subjects. For example, in the WY eyes-open condition, seed 42 yielded an outer mean of 0.602 while seed 123 yielded 0.454 - a difference of approximately 0.15 balanced accuracy points for the same classification problem. Similarly, in the AD eyes-open condition, seed 21 produced an outer mean of 0.554 compared to 0.464 for seed 42. Such sensitivity to the choice of random partition indicates that the observed performance levels were substantially driven by which particular subjects happened to be assigned to training versus test folds, rather than by a consistently learnable signal in the covariance features. The large within-seed standard deviations (up to 0.139 for individual seeds) and the standard deviations across seeds - ranging from 0.028 (WY eyes-closed) to 0.050 (WY eyes-open) - quantify this partition-induced variability and confirm that no condition produced consistently above-chance results regardless of how subjects were split.

4.2 Condition-Specific Observations

4.2.1 Eyes-Open versus Eyes-Closed

A minor performance difference was observed between the eyes-open and eyes-closed conditions. For the AD group, the eyes-open condition yielded a slightly higher mean outer balanced accuracy (0.499) compared to the eyes-closed condition (0.469). A similar trend was observed for the WY group, where the eyes-open condition (0.513) marginally outperformed the eyes-closed condition (0.507). However, given the small magnitude of these differences and the near-chance level of all four conditions, caution is warranted in interpreting this pattern. Furthermore, across both cohorts, the generalization gaps were slightly larger in the eyes-closed conditions (AD=0.063, WY=0.062) than in the eyes-open conditions (AD=0.039, WY=0.044).

One possible explanation for the slightly higher eyes-open performance with smaller generalization gap is that the dominant alpha rhythm, which is strongly suppressed upon eye opening [6, 7], contributes disproportionately to the estimated covariance structure during eyes-closed recordings. In the

eyes-closed condition, the alpha rhythm may dominate the covariance matrices, potentially reducing the relative influence of more subtle disorder-related differences and causing the inner-CV optimization to capture noise or subject-specific fluctuations. According to Barry et al., in contrast, eyes-open recordings involve ongoing visual processing and attentional engagement, producing a broader range of neural dynamics that may generate somewhat more variable covariance patterns. However, given that neither condition exceeded chance level significantly, this interpretation remains exploratory.

4.2.2 AD versus WY Groups

The two cohorts showed slightly different performance patterns and confusion matrix structures. The AD group, which consists predominantly of clinical participants, exhibited a consistent bias toward predicting the clinical class, resulting in high true positive counts but low true negative counts across all seeds. Conversely, the WY group, which consists predominantly of subclinical participants, showed the opposite bias, with the model largely predicting the subclinical class and achieving high true negative counts but very low true positive counts.

These opposing biases are consistent with the class distributions of the two cohorts and suggest that despite balanced accuracy optimization and class weighting, the models were unable to learn a reliable decision boundary that generalizes across subjects. Instead, the classifiers appear to have defaulted toward a majority-class prediction strategy, because balanced class weights equalize the training loss but don't guarantee balanced predictions when there is no discriminative signal.

Beyond the confusion matrix patterns, the summary metrics also revealed some differences between the two cohorts. The WY group achieved slightly higher mean outer balanced accuracy in both conditions (0.513 eyes-open, 0.507 eyes-closed) compared to the AD group (0.499 eyes-open, 0.469 eyes-closed). The WY group also showed consistently higher inner-CV means (0.558 and 0.569) compared to AD (0.538 and 0.532), suggesting that the inner loop found slightly more structure in the WY covariance features during hyperparameter tuning, even though this did not translate into reliably above-chance outer performance in either group. Notably, the generalization gaps were comparable between the two groups, ranging from 0.039 to 0.063 across all conditions.

One possible explanation for the slightly higher WY performance is the composition of the two cohorts. The WY group consists predominantly of subclinical participants, who by definition share a more homogeneous profile - healthy individuals varying in anxiety dimension scores but without diagnosed disorders. The AD group, by contrast, contains a mix of clinical participants spanning more than fifteen different diagnoses alongside a smaller number of healthy controls. This greater diagnostic heterogeneity within the AD group likely produces more variable covariance structures across subjects, making it harder for the classifier to find a consistent decision boundary. However, since neither group exceeded chance level significantly, these differences should be interpreted with caution.

4.3 Generalization Gap

The generalization gap - defined as the difference between the mean inner-CV balanced accuracy and the mean outer-CV balanced accuracy - was consistently positive across all conditions and the majority of seeds, indicating that inner-CV performance systematically overestimated outer performance. Mean generalization gaps ranged from 0.039 (AD eyes-open) to 0.063 (AD eyes-closed and WY eyes-closed).

In a small number of seeds, the generalization gap was slightly negative - for example, seed 21 in the AD eyes-open condition (-0.013) and seed 42 in the WY eyes-open condition (-0.031). A negative gap indicates that the outer test performance marginally exceeded the inner-CV estimate for that particular partition. Rather than indicating superior generalization, such values likely reflect the high variance of performance estimates across folds when the underlying signal is weak, consistent with near-chance classification performance throughout.

These gaps are modest in absolute terms, but this should not be taken as strong evidence that nested cross-validation effectively limited overfitting, given that performance was near chance. However, the fact that all gaps were positive indicates a systematic tendency for the inner loop to slightly overestimate model performance. This is a known property of nested cross-validation with small datasets: even with proper separation of tuning and evaluation, the inner CV operates on a smaller training set, and the best-scoring hyperparameter configuration may capitalize on random variation in the inner folds, leading to a small upward bias in the inner score relative to the true generalization performance [9].

A closer examination of the selected hyperparameters reveals some patterns across conditions. The ℓ_1-ratio of 0.2 was the most frequently selected value in all four conditions, indicating a systematic preference for predominantly ℓ_2 regularization over sparsity-inducing ℓ_1 penalization. To understand why, it is important to consider the structure of the tangent-space features. EEG signals recorded from neighboring electrodes overlap substantially due to volume conduction [12], which means that the resulting covariance matrices contain many highly correlated entries. When projected into tangent space, this likely produces feature vectors where many dimensions carry redundant and overlapping information. According to Zou et al., ℓ_1 regularization promotes sparsity by zeroing out coefficients, which works well when a small number of features are strongly and independently informative. However, when features are highly correlated and class-specific signal is weak and distributed across many dimensions - as expected here given the near-chance performance - ℓ_1 penalization could become unstable: no consistent sparse subset of features reliably discriminates between classes across different data partitions. ℓ_2 regularization handles this more gracefully by shrinking all correlated coefficients jointly rather than attempting to select a sparse subset [48].

Regarding the regularization strength C, a value of 1.0 (weak regularization) was most frequently selected in three of the four conditions (AD eyes-open, WY eyes-open, WY eyes-closed), with the most common combination overall being ($C = 1.0$, ℓ_1-ratio $= 0.2$). This suggests that the models attempted to exploit weak but broadly distributed covariance differences rather than applying strong shrinkage. The AD eyes-closed condition was an exception: smaller values of C (0.001 and 0.1) were more frequently selected, indicating that stronger regularization was required to prevent

overfitting in this condition. This is consistent with the lowest outer balanced accuracy observed across all conditions (0.469), suggesting that the AD eyes-closed covariance features contained the least structured class-specific information of the four settings.

Despite these mild tendencies, it is important to note that no hyperparameter configuration converged to a stable and consistent optimum across all folds and seeds. While ℓ_1-ratio $= 0.2$ and $C = 1.0$ appeared most frequently, substantial variation remained - different folds within the same seed, and the same fold across different seeds, often selected entirely different configurations. This residual instability indicates that the observed preferences reflect broad structural properties of the feature space (high inter-dimensional correlation, weak signal) rather than the discovery of a genuinely optimal regularization strategy. Had the tangent-space features contained strong and consistent class-specific information, a more clearly dominant hyperparameter configuration would be expected to emerge reliably across partitions.

4.4 Predicted Probability Distributions

The predicted probability distributions across the four conditions reveal two structurally distinct failure patterns. In both AD conditions, the predicted probability distributions of the subclinical and clinical classes are both sharply concentrated near 0.5, with substantial overlap and minimal separation between them. This unimodal, uncertainty-centered pattern reflects a failure mode in which the classifier is genuinely unable to assign meaningful probability estimates to individual subjects, effectively treating most subjects as equally likely to belong to either class. The near-identical distribution shapes for true class 0 and true class 1 in the AD conditions indicate that the tangent-space covariance features carry no consistent directional information that allows the model to distinguish clinical from subclinical participants.

The WY conditions exhibit a qualitatively different pattern. Both eyes-open and eyes-closed WY distributions are bimodal, with distinct peaks near 0.1 and 0.5. The subclinical majority (true class 0) shows a more pronounced first peak near 0.1, indicating that the model assigned a large subset of WY subjects very low clinical probability with moderate confidence. The clinical minority (true class 1) is concentrated near the second peak at 0.5, reflecting persistent uncertainty rather than confident clinical detection. This contrast represents a mechanistically different failure mode from the AD case: the AD distributions indicate uniform uncertainty across all subjects, whereas the WY distributions indicate asymmetric and partial discrimination - the model can identify a subset of clearly non-clinical subjects with some confidence, but fails to reliably detect the few clinical individuals in the cohort.

This pattern is consistent with the extreme class imbalance of the WY group, in which most subjects are healthy. Even with balanced class weighting equalizing the contribution of each class to the training loss, the classifier may learn covariance signatures that are primarily informative in the negative direction - features that distinguish the highly homogeneous WY subclinical majority - because these subjects dominate the feature space. The result is a model that assigns low clinical probability to a substantial portion of the easy negative cases while remaining uncertain about the clinical minority,

producing the bimodal shape. Importantly, this bimodal structure appears in both WY conditions with high consistency, suggesting it reflects a structural property of the cohort's covariance feature distribution.

4.5 Comparison with Existing Literature

The near-chance performance observed in this thesis contrasts with many published EEG classification studies, but is consistent with the broader pattern of results for resting-state EEG classification of psychiatric conditions.

Barachant et al. demonstrated that covariance-based Riemannian methods can achieve excellent performance in motor imagery BCI applications, reporting mean classification accuracies of up to 86% [4]. Similarly, the review by Congedo et al. [12] summarizes numerous studies in which Riemannian geometry achieved state-of-the-art performance across diverse BCI paradigms. However, a critical distinction applies: these studies investigated paradigms specifically designed to evoke strong and reproducible neural responses, such as motor imagery, P300, and steady-state visual evoked potentials, where class-specific neural activity produces substantial and consistent differences in covariance structure across trials and subjects.

In contrast, resting-state EEG lacks this external event structure. Resting-state dynamics reflect spontaneous neural fluctuations driven by ongoing network activity, which vary substantially within and between subjects over time [21]. Furthermore, this variability makes it considerably more challenging to extract stable, subject-level covariance representations that reliably distinguish clinical from non-clinical groups.

Even within the psychiatric EEG classification literature, where resting-state features are used, results are mixed. Park et al. [34] applied elastic-net classifiers to resting-state quantitative EEG from 945 subjects across six major psychiatric disorder categories. They reported classification accuracies of up to 91% for anxiety disorders using whole-band power spectral density, but only after IQ adjustment and with disorder-specific groupings. Crucially, their study used disorder-specific groups with homogeneous diagnoses, whereas the present work classified across a highly heterogeneous clinical group spanning more than fifteen different diagnoses. Such diagnostic heterogeneity likely reduces the consistency of neural signatures across clinical subjects, making binary classification more challenging.

Kim et al. [23] applied a Riemannian geometry-based classifier (FgMDM) to resting-state EEG for PTSD classification, and noted that previous studies on machine-learning-based diagnosis of PTSD with resting-state EEG reported poor accuracies as low as 60%, before their Riemannian approach improved performance to 75%. This observation supports the view that resting-state covariance-based classification of psychiatric conditions is inherently challenging, even when the clinical group is relatively homogeneous (PTSD only). The difficulty is compounded in the present work by the broad diagnostic heterogeneity of the clinical group.

Consistent with these observations, Uyanik et al. [45] conducted a systematic review of EEG-based

classification across multiple neurological and psychiatric disorders, finding that anxiety detection consistently yielded the lowest classification accuracy among all investigated conditions. The authors attribute this to the absence of obvious amplitude or frequency differences in anxiety-related EEG, in contrast to conditions such as epilepsy where pathological EEG patterns are more clearly distinguishable. This directly contextualizes the near-chance performance observed in the present work, which targeted a heterogeneous anxiety and psychiatric disorder group.

Together, these findings suggest that the near-chance performance described in this thesis is not solely attributable to the choice of pipeline, but may reflect fundamental properties of the classification problem: the combination of resting-state EEG, heterogeneous clinical labels - many of which anxiety disorders - and subject-level classification likely represents a particularly challenging setting for any machine-learning approach.

4.6 Limitations

Several limitations should be considered when interpreting the results of this thesis.

First, the clinical group comprised participants diagnosed with a wide range of mental disorders, including anxiety disorders, depressive disorders, OCD, PTSD, somatic disorders, and others. This diagnostic heterogeneity means that the positive class does not correspond to a homogeneous neural profile, and different clinical participants may exhibit very different EEG signatures. The binary label ”at least one diagnosed disorder” (clinical) aggregates these diverse profiles into a single class, potentially reducing the ability to detect disorder-specific neural patterns that a more targeted classification approach might detect.

Second, the analysis relied exclusively on covariance-based features derived from resting-state EEG. While covariance matrices capture important spatial relationships between channels, they do not explicitly represent spectral information or temporal dynamics [27]. Potentially informative biomarkers, such as frequency-band power, alpha asymmetry, or other functional connectivity measures, are not represented in the feature space of the present pipeline. Combining multiple feature types may improve classification performance.

Third, the available sample comprised 146 AD subjects and 149 WY subjects. Although this is reasonable for EEG research, it remains modest for subject-level machine learning with high inter-subject variability. After nested cross-validation splits, individual models were effectively trained on approximately 116-120 subjects, which may limit the stability of the learned decision boundaries, particularly given the heterogeneity of the clinical group.

Fourth, only a linear classifier was applied after tangent-space projection. While logistic regression offers interpretability and computational efficiency, it may be unable to capture nonlinear relationships between covariance features and clinical labels that a more flexible classifier could detect.

Finally, model evaluation was performed using nested cross-validation on a single dataset collected within a single research project. Independent external validation on data from different sites, acqui-

sition systems, or populations would be required to assess the robustness and generalizability of the findings.

4.7 Contributions and Future Work

Despite the modest classification performance, this thesis makes several contributions to the field of EEG-based mental health classification.

First, it provides a systematic and reproducible evaluation of a complete covariance-based Riemannian pipeline for subject-level resting-state EEG classification of clinical or subclinical participants. The pipeline - comprising OAS-regularized covariance estimation, Riemannian mean aggregation, tangent-space projection, elastic-net logistic regression, and nested cross-validation with multiple random seeds and permutation test - represents a principled and methodologically rigorous approach that can serve as a baseline for future work.

Second, the results suggest that resting-state EEG covariance features alone may be insufficient for reliable binary classification of heterogeneous clinical groups. This finding has practical implications for future study design. Researchers may need to consider more targeted and diagnostically homogeneous clinical groupings, richer feature representations like combining covariance with spectral features for example, or the use of task-evoked EEG paradigms - such as the flanker, picture-viewing, or fear-conditioning tasks - may provide more discriminative neural signals for classification than the resting-state recordings analyzed here.

5 Conclusion

This thesis investigated whether resting-state EEG contains discriminative information that allows the classification of participants as either subclinical or having at least one diagnosed mental disorder. To this end, a Riemannian geometry-based machine-learning pipeline was implemented and evaluated, combining OAS-regularized covariance estimation, Riemannian mean aggregation, tangent-space projection, elastic-net logistic regression, and nested cross-validation with multiple random seeds. The pipeline was applied to resting-state EEG recordings from two cohorts - the AD group (predominantly clinical, $n = 146$) and the WY group (predominantly subclinical, $n = 149$) - across both eyes-open and eyes-closed conditions.

The primary finding is that the pipeline did not achieve reliable classification performance in any of the four investigated conditions. Mean outer-fold balanced accuracy ranged from 0.469 to 0.513 across all conditions, remaining close to chance level throughout. Permutation tests confirmed that none of the observed performances exceeded what would be expected under random label assignment. Generalization gaps were modest and positive across conditions, indicating that nested cross-validation effectively limited overfitting, but that the learned models captured only weakly discriminative structure in the tangent-space features. Confusion matrices revealed systematic prediction biases reflecting the class imbalance of each cohort rather than genuine class separation, and predicted probability distributions showed substantial overlap between clinical and subclinical subjects in all conditions.

These findings have several implications for future research. First, disorder-specific classification - separating participants with a single well-defined diagnosis from subclinical controls - may yield more consistent neural signatures and higher classification performance. Second, enriching the feature representation by combining covariance-based features with spectral power, functional connectivity, or nonlinear complexity measures may provide additional discriminative information. Finally, task-evoked EEG paradigms available in the same dataset may elicit disorder-specific neural responses that are more amenable to classification than resting-state recordings.

In summary, this thesis evaluates a Riemannian EEG classification pipeline and investigates whether covariance-based resting-state features can separate clinical and subclinical participants. While the results do not support the use of resting-state EEG covariance features alone for reliable binary classification of heterogeneous clinical groups, they contribute to the understanding of this challenging problem and offer a reproducible baseline for future research.

Used Tools

Various digital tools were used in a supporting role for the present work. The generic AI tools ChatGPT by OpenAI, Microsoft Copilot by Microsoft, and Claude by Anthropic were used for grammar, spelling checks, linguistic refinement, and scientific phrasing. They also supported the creation of all illustrations that were not taken from other cited sources, as well as the tables and figures from the author's own analysis results. In addition, they were also used for programming-related assistance, particularly for Python and LaTeX for pseudocode, where individual suggested code fragments were independently checked, adapted, tested, and verified. All content developed with the help of these tools was independently reviewed, critically evaluated, and is the full responsibility of the author.

List of Figures

List of Tables

Bibliography

[1] Alexandre Barachant, Quentin Barthélemy, Jean-Rémi King, Alexandre Gramfort, Sylvain Chevallier, Pedro L. C. Rodrigues, Emanuele Olivetti, Vladislav Goncharenko, Gabriel Wagner vom Berg, Ghiles Reguig, Arthur Lebeurrier, Erik Bjäreholt, Maria Sayu Yamamoto, Pierre Clisson, Marie-Constance Corsi, Igor Carrara, Apolline Mellot, Bruna Junqueira Lopes, Brent Gaisford, Ammar Mian, Anton Andreev, and Gregoire Cattan. pyriemann, 2026.

[2] Alexandre Barachant, Stephane Bonnet, Marco Congedo, and Christian Jutten. Riemannian geometry applied to BCI classification. In *Conference on Latent Variable Analysis (LVA/ICA), St Malo*, 2010.

[3] Alexandre Barachant, Stephane Bonnet, Marco Congedo, and Christian Jutten. Multiclass brain–computer interface classification by riemannian geometry. In *2012 IEEE Transactions on Biomedical Engineering*, pages 920–928. IEEE, 2012.

[4] Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Classification of covariance matrices using a Riemannian-based kernel for BCI applications. *Neurocomputing*, 112:172–178, July 2013.

[5] Alexandre Barachant, Stephane Bonnet, Marco Congedo, and Christian Jutten. A brain-switch using riemannian geometry. In *BCI 2011 - 5th International Brain-Computer Interface Conference*, pages 64–67, Sep 2011.

[6] Robert J. Barry, Adam R. Clarke, Stuart J. Johnstone, Christopher A. Magee, and Jacqueline A. Rushby. EEG differences between eyes-closed and eyes-open resting conditions. *Clinical Neurophysiology*, 118(12):2765–2773, 2007.

[7] Robert J. Barry and Frances M. De Blasio. EEG differences between eyes-closed and eyes-open resting remain in healthy ageing. *Biological Psychology*, 129:293–304, 2017.

[8] Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M. Buhmann. The balanced accuracy and its posterior distribution. In *2010 20th International Conference on Pattern Recognition*, pages 3121–3124, 2010.

[9] Gavin C Cawley and Nicola LC Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *The Journal of Machine Learning Research*, 11:2079–2107, 2010.

[10] Yilun Chen, Ami Wiesel, Yonina C. Eldar, and Alfred O. Hero. Shrinkage algorithms for MMSE covariance estimation. *IEEE Transactions on Signal Processing*, 58(10):5016–5029, October 2010.

[11] S. Chevallier, E.K. Kalunga, and Q. Barthélemy. Review of riemannian distances and divergences. In *SSVEP-based BCI. Neuroinform 19*, pages 93–106, 2021.

[12] Marco Congedo, Alexandre Barachant, and Rajendra Bhatia. Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review. *Brain-Computer Interfaces*, 4(3):155–174, 2017.

[13] D. R. Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–232, 07 1958.

[14] David Roxbee Cox. *Analysis of binary data*. Routledge, 2018.

[15] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives, 2014.

[16] W. Förstner and B. Moonen. A metric for covariance matrices. In *Technical report, Dept. of Geodesy and Geoinformatics, Stuttgart Universität*, 1999.

[17] Keinosuke Fukunaga. *Introduction to statistical pattern recognition*. Elsevier, 2013.

[18] Joachim Gross, Sylvain Baillet, Gareth R. Barnes, Richard N. Henson, Arjan Hillebrand, Ole Jensen, Karim Jerbi, Vladimir Litvak, Burkhard Maess, Robert Oostenveld, Lauri Parkkonen, Jason R. Taylor, Virginie van Wassenhove, Michael Wibral, and Jan-Mathijs Schoffelen. Good practice for conducting and reporting MEG research. *NeuroImage*, 65:349–363, 2013.

[19] Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[20] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*. John Wiley & Sons, 2013.

[21] Arjun Khanna, Alvaro Pascual-Leone, Christoph M. Michel, and Faranak Farzan. Microstates in resting-state EEG: Current status and future directions. *Neuroscience Biobehavioral Reviews*, 49:105–113, 2015.

[22] Ashima Khosla, Padmavati Khandnor, and Trilok Chand. A comparative analysis of signal processing and classification methods for different applications based on EEG signals. *Biocybernetics and Biomedical Engineering*, 40(2):649–690, 2020.

[23] Yong-Wook Kim, Sungkean Kim, Miseon Shim, Min Jin Jin, Hyeonjin Jeon, Seung-Hwan Lee, and Chang-Hwan Im. Riemannian classifier enhances the accuracy of machine-learning-based diagnosis of ptsd using resting EEG. *Progress in Neuro-Psychopharmacology and Biological Psychiatry*, 102:109960, 2020.

[24] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.

[25] Eric Larson. Mne-python, April 2026.

[26] H. Lin. Detecting stress based on social interactions in social networks. 29(9):1820–1833, 1 Sept. 2017.

[27] Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy, and Florian Yger. A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, 15(3):031005, 2018.

[28] Maher Moakher. A differential geometric approach to the geometric mean of symmetric positive-definite matrices. *SIAM Journal on Matrix Analysis and Applications*, 26(3):735–747, 2005.

[29] Maher Moakher. A riemannain framework for the averaging, smoothing and interpolation of some matrix-valued data. In *Processing of the 17th International Symposium on Mathematical Theory of Networks and Systems*, pages 1720–1729, 2006.

[30] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[31] Javier Olias, Rubén Martín-Clemente, M. Auxiliadora Sarmiento-Vega, and Sergio Cruces. EEG signal processing in mi-bci applications with improved covariance matrix estimators. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 27(5):895–904, 2019.

[32] World Health Organization. Mental disorders. 13 August 2025.

[33] Dilber Uzun Ozsahin, Mubarak Taiwo Mustapha, Auwalu Saleh Mubarak, Zubaida Said Ameen, and Berna Uzun. Impact of feature scaling on machine learning models for the diagnosis of diabetes. In *2022 International Conference on Artificial Intelligence in Everything (AIE)*, pages 87–94, 2022.

[34] Su Mi Park, Boram Jeong, Da Young Oh, Chi-Hyun Choi, Hee Yeon Jung, Jun-Young Lee, Donghwan Lee, and Jung-Seok Choi. Identification of major psychiatric disorders from resting-state electroencephalography using a machine learning approach. *Frontiers in Psychiatry*, Volume 12 - 2021, 2021.

[35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[36] X. Pennec, P. Fillard, and N. Ayache. A Riemannian Framework for Tensor Computing. In *Int J Comput Vision 66*, pages 41–66, 2006.

[37] G. Pfurtscheller and F.H. Lopes da Silva. Event-related EEG/MEG synchronization and desynchronization: basic principles. *Clinical Neurophysiology*, 110(11):1842–1857, 1999.

[38] Belinda Phipson and Gordon K Smyth. Permutation p-values should never be zero: Calculating exact p-values when permutations are randomly drawn. *Statistical Applications in Genetics and Molecular Biology*, 9(1), October 2010.

[39] Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché Buc, Emily Fox, and Hugo Larochelle. Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program). *Journal of machine learning research*, 22(164):1–20, 2021.

[40] Herbert Ramoser, Johannes Muller-Gerking, and Gert Pfurtscheller. Optimal spatial filtering of single trial EEG during imagined hand movement. *IEEE transactions on rehabilitation engineering*, 8(4):441–446, 2000.

[41] Claude E Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, 1949.

[42] C. Su, Z. Xu, and J. Pathak. Deep learning in mental health outcome research: a scoping review. 2020.

[43] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 01 1996.

[44] Ryota Tomioka, Kazuyuki Aihara, and Klaus-Robert Müller. Logistic regression for single trial EEG classification. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.

[45] Hakan Uyanik, Abdulkadir Sengur, Massimo Salvi, Ru-San Tan, Jen Hong Tan, and U. Rajendra Acharya. Automated detection of neurological and mental health disorders using eeg signals and artificial intelligence: A systematic review. *WIREs Data Mining and Knowledge Discovery*, 15(1):e70002, 2025. e70002 DMKD-00820.R1.

[46] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[47] Andreas Widmann, Erich Schröger, and Burkhard Maess. Digital filter design for electrophysiological data – a practical approach. *Journal of Neuroscience Methods*, 250:34–46, 2015. Cutting-edge EEG Methods.

[48] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 04 2005.