



HAL
open science

Transfert d'apprentissage intra et inter sujets en interfaces cerveau-machine non-invasives

Alexandre Bleuzé

► **To cite this version:**

Alexandre Bleuzé. Transfert d'apprentissage intra et inter sujets en interfaces cerveau-machine non-invasives. Bio-informatique [q-bio.QM]. Université Grenoble Alpes [2020-..], 2023. Français. NNT : 2023GRALS057 . tel-04591006

HAL Id: tel-04591006

<https://theses.hal.science/tel-04591006v1>

Submitted on 28 May 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : ISCE - Ingénierie pour la Santé la Cognition et l'Environnement

Spécialité : BIS - Biotechnologie, instrumentation, signal et imagerie pour la biologie, la médecine et l'environnement

Unité de recherche : Grenoble Images Parole Signal Automatique

Transfert d'apprentissage intra et inter sujets en interfaces cerveau-machine non-invasives

Intra- and Inter-Subject Transfer Learning for Non-Invasive Brain-Computer Interface

Présentée par :

Alexandre Bleuzé

Direction de thèse :

Marco CONGEDO

DIRECTEUR DE RECHERCHE, CNRS DELEGATION ALPES

Directeur de thèse

Jérémie MATTOU

CHARGE DE RECHERCHE, INSERM DELEGATION AUVERGNE-RHONE-ALPES

Co-encadrant de thèse

Rapporteurs :

SAID MOUSSAOUI

PROFESSEUR DES UNIVERSITES, ECOLE CENTRALE NANTES

SYLVAIN CHEVALIER

PROFESSEUR DES UNIVERSITES, UNIVERSITE PARIS-SACLAY

Thèse soutenue publiquement le **8 décembre 2023**, devant le jury composé de :

OLIVIER MICHEL,

PROFESSEUR DES UNIVERSITES, GRENOBLE INP

Président

MARCO CONGEDO,

DIRECTEUR DE RECHERCHE, CNRS DELEGATION ALPES

Directeur de thèse

SAID MOUSSAOUI,

PROFESSEUR DES UNIVERSITES, ECOLE CENTRALE NANTES

Rapporteur

SYLVAIN CHEVALIER,

PROFESSEUR DES UNIVERSITES, UNIVERSITE PARIS-SACLAY

Rapporteur

MARIE-CONSTANCE CORSI,

CHARGE DE RECHERCHE, INRIA CENTRE DE PARIS

Examinatrice

TETIANA AKSENOVA,

DIRECTRICE DE RECHERCHE, CEA CENTRE DE GRENOBLE

Examinatrice

Invités :

JEREMIE MATTOU

CHARGE DE RECHERCHE, INSERM DELEGATION AUVERGNE-RHONE-ALPES



Remerciements

Cher lecteur ou lectrice, si tu lis ces lignes c'est que tu t'apprêtes certainement à lire au moins une partie de cette thèse, j'espère qu'elle te plaira. Cependant toute thèse est un travail sur le long terme qui n'est réalisable qu'avec l'aide et le soutien d'autres personnes.

Pour cela, c'est avec beaucoup d'émotion que j'aimerais tout d'abord remercier les membres de mon jury Marie-Constance Corsi, Tetiana Aksenova, Olivier Michel d'avoir pris le temps de venir assister à cette thèse et de m'avoir fait un grand nombre de retours constructifs. Plus précisément je voudrais remercier tout particulièrement Saïd Moussaoui et Sylvain Chevalier pour avoir accepté de prendre le rôle de rapporteur pour cette thèse et d'y avoir porté une attention toute particulière.

Je remercie aussi mes superviseurs Marco Congedo et Jérémie Mattout pour m'avoir guidé durant ces 3 années. Je suis heureux d'avoir eu deux superviseurs qui se soient aussi bien complétés et je conseillerais à n'importe qui d'avoir toujours un second superviseur pour ces raisons. Merci de m'avoir autant appris autant sur le plan scientifique que humain.

Ensuite je souhaiterais remercier Aurélien Van Langenhove pour m'avoir fait découvrir le domaine passionnant des interfaces cerveau-ordinateur. Saïd et toi m'avez appris toutes les bases du domaine. Tu retrouveras d'ailleurs peut-être quelques uns de tes conseils dans toute la partie introductive de la thèse.

Je voudrais maintenant remercier chaleureusement toutes les personnes que j'ai pu rencontrer au cours de mes 3 années de thèse au GIPSA-lab, que ce soit mes co-bureau Davoud, Juliette, Fatih, Hung ou simplement des personnes avec qui j'ai pu partager des moments qui ont pu me motiver au cours de ces 3 années comme Clémentine, Charles, Léa, Clément, Roza, Yigit, Olivier et j'en oublie beaucoup. Je voudrais également remercier tout particulièrement Arthur pour avoir été mon plus gros soutien et ami pendant cette dernière année de thèse qui aurait pu être particulièrement difficile comme c'est le cas pour beaucoup de gens. Et comment terminer ce paragraphe sans remercier toutes les personnes des relations humaines du GIPSA et également de l'école doctorale EDISCE pour leur aide et leur disponibilité notamment sur mes débuts au GIPSA avec l'aide de Virginie Faure ou encore sur la fin de la thèse auprès de l'école doctorale avec l'aide d'Orélie Garzena.

Et le travail de thèse n'ayant pas lieu qu'au labo, je voudrais aussi remercier tous mes amis en dehors, Roman, Vincent, Albane, Tomass avec une mention

spéciale pour Iris avec qui les journées bibliothèque tout l'été m'ont permis d'écrire cette thèse bien plus rapidement que ce que j'aurais cru possible. J'aimerais aussi remercier Inès d'avoir été là pour me soutenir sur la fin de cette thèse mais également pour me rendre aussi heureux depuis maintenant quelques mois et beaucoup d'autres à venir.

Je voudrais aussi remercier ma famille qui m'ont toujours aidé et soutenu en plus de me pousser à me dépasser et à faire de mon mieux pour arriver là où j'en suis aujourd'hui.

Finalement je voudrais te remercier toi lecteur ou lectrice pour l'intérêt que tu portes à ce travail en espérant que ces "quelques pages" t'aient fait découvrir ou apprendre des choses sur ce domaine captivant qu'est celui des interfaces cerveau-ordinateur.

Table des matières

Remerciements	2
1 Introduction	10
1.1 Électroencéphalogramme	11
1.2 Interface Cerveau-Ordinateur (ICO)	11
1.3 Pré-traitement	15
1.3.1 Le filtrage fréquentiel	15
1.3.2 Le filtrage spatial	16
1.4 Matrice de covariance et géométrie riemannienne	21
1.4.1 Matrice de covariance en MI	22
1.4.2 Matrice de covariance en ERP	23
1.4.3 Matrice de covariance en SSVEP	24
1.4.4 Un point sur la classification dans la variété riemannienne	27
1.4.5 De la matrice de covariance au vecteur de caractéristiques	28
1.4.6 Classification dans l'espace tangent	32
1.5 Apprentissage par transfert	35
1.5.1 La variabilité en ICO	35
1.5.2 Les pistes pour résoudre cette variabilité	37
1.5.3 État de l'art des méthodes d'apprentissage par transfert pour les ICO riemanniennes	40
2 Apprentissage par transfert en ICO	45
2.1 Le Tangent Space Alignment (TSA) origine et première version .	46
2.1.1 État de l'art dans le TL	46
2.1.2 Première version du TSA	47
2.1.3 Données et algorithmes	49
2.1.4 Résultats	51
2.1.5 Premières conclusions et discussion	54
2.2 Étude des améliorations apportées au TSA	55
2.2.1 Les défauts de la première version	55
2.2.2 Améliorer la matrice des produits croisés \mathbf{K}	58
2.2.3 Utiliser un assemblage d'estimations de moyennes varié et robuste	59
2.2.4 TSA améliorée par PCA (TSAxPCA)	60
2.2.5 TSA améliorée par CSP (TSAxCSP)	62

2.2.6	TSA améliorée à l'aide de valeurs propres (TSAxEIG) . . .	66
2.2.7	TSA améliorée à l'aide de bootstraps de moyennes (TSAx- BOOTSRAP)	66
2.3	Étude du TSAxPCA élargie à plusieurs paradigme et sur un grand nombre de données	67
2.3.1	Description des données	67
2.3.2	Résultats obtenus	69
2.3.3	Discussion des résultats obtenus avec la TSAxPCA	78
3	Apprentissage par transfert inter-configurations	82
3.1	Généralisation inter-configurations du TSAxPCA	82
3.2	La source arbitraire	84
3.3	Résultats intra- et inter-configurations de la source arbitraire . . .	88
3.3.1	Résultats intra-configuration	90
3.3.2	Résultats inter-configurations	94
3.3.3	Discussion	101
4	Conclusions et perspectives	102
4.1	Le TSA et l'apprentissage par transfert	102
4.2	La source arbitraire et le transfert d'information entre bases de données hétérogènes	103
4.3	Perspectives futures	104
A	Résultats supplémentaires	106
A.1	Résultats détaillés du Chapitre 2	107
A.2	Résultats détaillés du Chapitre 3	148

Abbreviations

CSP	Common Spatial Pattern
cVEP	code-modulated visual evoked potentials
EEG	électroencéphalogramme
ERP	event related potentials
ICA	Independent Component Analysis
ICO	interface cerveau-ordinateur
LDA	Linear Discriminant Analysis
LR	Logistic Regression
MDRM	Minimum Distance to Riemannian Mean
MI	motor imagery
PCA	Principal Components Analysis
SDP	symétrique définie positive
SSVEP	steady-state visual evoked potentials
SVC	Support Vector Classifier
SVM	Support Vector Machine
TL	transfer learning
TSLDA	Tangent Space Linear Discriminant Analysis

Contexte

Une interface cerveau-ordinateur (ICO) désigne un système de liaison directe entre un cerveau et un ordinateur, permettant à un individu d'effectuer des tâches sans passer par l'action des nerfs périphériques et des muscles. Le terme Interface Cerveau-Ordinateur ou Brain-Computer Interface (BCI) en anglais a été premièrement introduit par Jacques Vidal en 1973 dans son article intitulé "Toward Direct Brain-Computer Communication" [84]. Cette dénomination a ensuite été reprise et précisée dans différents articles, notamment "Brain-computer interfaces for communication and control" de J. R. Wolpaw et al. [89]. Les ICO sont devenues de plus en plus intéressantes dans le secteur de la santé en raison de leur potentiel d'aide aux patients ([37], [78]). Elles ont été utilisées pour aider certaines personnes à récupérer leurs fonctions motrices après un accident vasculaire cérébral ou une lésion de la moelle épinière ou encore pour aider les personnes atteintes de maladies telles que la sclérose latérale amyotrophique ou la sclérose en plaques, qui ont perdu la capacité de contrôler leurs membres ou de communiquer [69]. Les ICO peuvent également être utilisés pour surveiller l'activité cérébrale des patients atteints d'épilepsie [3] ou de troubles du sommeil. Un autre facteur qui ajoute à l'attrait des ICO est leur potentiel d'augmentation des capacités des personnes en bonne santé dans des domaines tels que les jeux vidéo, par exemple ([54], [46]). Des compétitions ont même vu le jour, comme le Cybathlon, pour promouvoir les progrès des applications ICO ([65]).

La recherche en interface cerveau-ordinateur est en pleine essor pour beaucoup de raisons, notamment l'amélioration du matériel permettant leur mise en place (principalement les électroencéphalogrammes) ainsi que les avancées technologiques prodigieuses qui ont eu lieu en terme d'algorithmes d'apprentissage machine depuis le début du 21ème siècle. De nombreux articles sont publiés chaque jours et les avancées du domaine sont de plus en plus médiatisées. Ces systèmes sont pourtant loin d'avoir les caractéristiques requises pour être utilisés par le grand public. Le matériel requis reste souvent cher pour avoir un signal de qualité et son utilisation dans le cadre d'une ICO demande une calibration importante à chaque début de session. Cependant, de plus en plus de données sont désormais disponibles en libre accès sur internet. Ces données pourraient rendre possible la création de modèles ayant vu les données de nombreuses personnes, permettant à la fois d'augmenter les performances des systèmes tout en réduisant leur temps de calibration. Cela aurait également pour conséquence de permettre l'utilisation de matériel moins performant pour des performances équivalentes, rendant les ICO plus abordables. Le problème principal aujourd'hui est que les données disponibles en accès libre sont très hétérogènes, que ce soit en terme de qualité, de paradigme ou même simplement de matériel. Pour ces raisons, il est très difficile d'adapter toutes ces données pour en retirer des caractéristiques communes permettant la classification des signaux.

Dans le but de retirer des caractéristiques commune à différentes bases de données, les auteurs de ([90]) et ([58]) ont récemment publié de nouveaux ensembles de données pour les paradigmes de potentiels évoqués P300 via de la

représentation visuelle séquentielle rapide et en imagerie motrice respectivement, où la précision et les caractéristiques ont été profondément étudiées pour assurer le meilleur apprentissage par transfert possible, que ce soit entre les sujets, entre les ensembles de données ou même entre les paradigmes. En outre, les auteurs de ([43]) ont développé une bibliothèque Python open source qui vise à faciliter l'utilisation des ensembles de données ICO pour n'importe quel paradigme et à faciliter la comparaison des méthodes avec n'importe quelle métrique pour chaque ensemble de données disponible. Cette bibliothèque est encore en développement actif et permet déjà d'utiliser et de comparer plus de 30 jeux de données dans trois paradigmes bien connus étant les potentiels évoqués P300, l'imagerie motrice et les potentiels évoqués visuels stables. Ces trois paradigmes seront décrits plus en détails en partie 1.2.

Objectifs de la thèse

Cette thèse a pour objectif de proposer des méthodes permettant d'adapter et d'utiliser les données disponibles en accès libre afin de créer des modèles d'apprentissage automatique très robustes car entraînés sur les données de nombreux sujets. Pour ce faire, on s'intéresse à la géométrie riemannienne dont l'utilisation dans le cadre des interfaces cerveau-ordinateur a été mis en avant par les travaux d'Alexandre Barachant [6]. Mes travaux s'inscrivent également dans la suite de ceux de Paulo Zanini et Perdo Luiz Rodriguez ayant également développée plusieurs algorithmes permettant d'adapter les données des sujets les uns aux autres [101], [71], parfois dans des configurations expérimentales très différentes [70].

Notations

Tout au long de cette thèse, le maximum a été fait pour garder des notations claires et simples à comprendre. Les conventions suivantes ont été utilisées pour désigner les objets mathématiques : les matrices seront représentées par des caractères gras majuscules (par exemple \mathbf{A}), les vecteurs seront désignés par des caractères gras minuscules (par exemple \mathbf{a}), les scalaires et les indices seront représentés par des caractères italiques minuscules (par exemple a), les constantes seront représentées par des caractères italiques majuscules (par exemple A), et les ensembles d'objets mathématiques seront généralement représentés par des caractères majuscules stylisés (par exemple \mathcal{A} ou \mathcal{S}_{++}^N dénotant l'ensemble des matrices symétriques définies positives de taille (N, N)), sauf cas particulier tels que \mathbb{R} ou \mathbb{C} étant respectivement l'ensemble des réels et l'ensemble des complexes. La trace d'une matrice sera indiquée par la fonction $tr(\cdot)$, sa transposée par $(\cdot)^T$, la norme 2 ou la norme de Frobenius par $\|\cdot\|$, les distances seront usuellement représentées par un $\delta(\cdot, \cdot)$, notamment la distance riemannienne $\delta_R(\cdot, \cdot)$, le produit de Hadamard par \circ , la fonction logarithme par \log et son analogue matriciel par $\text{Log}(\cdot)$, et de même l'exponentielle par \exp et son analogue matriciel par $\text{Exp}(\cdot)$. De manière générale les opérateurs mathématiques seront en

caractères normaux (par exemple $A(\cdot)$ ou $a(\cdot)$). Enfin, \mathbf{I}_N désignera la matrice identité en dimension N .

Chapitre 1

Introduction

Table des matières

1.1	Électroencéphalogramme	11
1.2	Interface Cerveau-Ordinateur (ICO)	11
1.3	Pré-traitement	15
1.3.1	Le filtrage fréquentiel	15
1.3.2	Le filtrage spatial	16
1.4	Matrice de covariance et géométrie riemannienne	21
1.4.1	Matrice de covariance en MI	22
1.4.2	Matrice de covariance en ERP	23
1.4.3	Matrice de covariance en SSVEP	24
1.4.4	Un point sur la classification dans la variété riemannienne	27
1.4.5	De la matrice de covariance au vecteur de caractéristiques	28
1.4.6	Classification dans l'espace tangent	32
1.5	Apprentissage par transfert	35
1.5.1	La variabilité en ICO	35
1.5.2	Les pistes pour résoudre cette variabilité	37
1.5.3	État de l'art des méthodes d'apprentissage par transfert pour les ICO riemanniennes	40

1.1 Électroencéphalogramme

Lorsque le cerveau fonctionne, les neurones s'activent et génèrent un signal électrique dans le cerveau en agissant comme des dipôles indépendants. Il génèrent alors un champ électrique et magnétique dans le cerveau, dépendant entre autre de l'activation des neurones et de la cohérence de ces activations dans au sein d'une même zone dans le cortex. Ces champs sont observables à l'aide de différent outils, tels que l'électroencéphalogramme (EEG) sous forme d'ondes à des fréquences et des localisations caractéristiques de l'activité.

L'invention de l'EEG est attribuée à Hans Berger, psychiatre allemand qui enregistra un signal d'activité cérébrale humaine le premier en 1924.[105] L'EEG est un système permettant d'enregistrer les signaux électrophysiologiques provenant de son utilisateur. Ce système est constitué principalement d'un bonnet d'électrodes relié à un amplificateur qui envoie ensuite les données numérisées à un ordinateur. Pour chaque électrode, la différence entre le voltage de celle-ci et celui d'une électrode donnée (appelée référence) en général commune à tout le système, permet de déduire l'activité électrique du scalp au niveau de l'électrode. Celle-ci peut provenir d'un grand nombre de sources, qu'elle soient musculaires, cérébrales, oculaires, cardiaques, ou encore matérielles. Dans le cadre d'une interface cerveau-ordinateur, seules certaines sources cérébrales sont intéressantes et celles-ci ont tendance à avoir un *rapport signal sur bruit* très faible, c'est à dire que l'amplitude des signaux intéressants est faible comparée à celle des signaux qui ne le sont pas. Cela pose des problèmes lors du traitement du signal qui a lieu par la suite et demande notamment un filtrage précis qui sera traité en 1.3.

Les positions et nombre d'électrodes pour un bonnet EEG peuvent grandement différer d'une configuration à l'autre. Elles vont cependant dans la grande majorité des cas suivre des normes connues pour le placement, permettant un traitement des données normalisé. [19]

1.2 Interface Cerveau-Ordinateur (ICO)

Une Interface Cerveau-Ordinateur est un système informatisé permettant de traiter les signaux électrophysiologiques provenant du cerveau de l'utilisateur afin de les transformer en commande pour un ordinateur ou une machine [89]. La façon dont sont traités les signaux sera présenté dans la suite, notamment en parties 1.3 et 1.4. Ensuite, une fois la commande définie, celle-ci est envoyée vers l'ordinateur pour être traitée. Cette commande peut être interprétée de différentes manières et donner lieu ou non à un feedback destiné à l'utilisateur. Les ICO ne donnant pas de feedback sont généralement liées à des séances de calibration permettant à la machine d'apprendre un modèle pour comprendre ensuite les données envoyées par le sujet. Ce type d'ICO sont dites en "boucle ouverte" car comme on peut le voir en figure 1.1, la boucle n'est pas fermée sans le feedback envoyé vers l'utilisateur permettant potentiellement de corriger son comportement. On oppose à ces ICO en boucle ouverte les ICO en "boucle

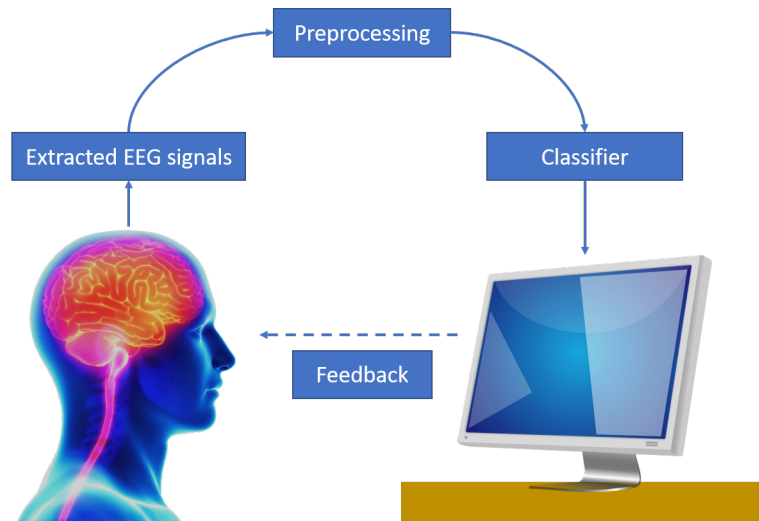


FIGURE 1.1 – Fonctionnement d’une ICO. Les signaux sont recoltés lorsque l’utilisateur exécute la tâche permettant le contrôle. Ces signaux sont ensuite pré-traités et traités afin d’être finalement transformés en commande exécutée par l’ordinateur. En boucle fermée, l’utilisateur reçoit également un feedback provenant de l’ordinateur ou de la machine contrôlée.

fermée". Ces ICO sont logiquement celles pour lesquelles un feedback est donné à l’utilisateur. Ce feedback peut prendre plusieurs formes, allant de la simple image fixe affichée sur un écran au feedback du contrôle moteur d’un système tel qu’un fauteuil roulant ou une prothèse robotique. Il est également aussi possible d’afficher des jauges de contrôle en temps réel à l’écran ou encore d’exécuter des actions dans un jeu vidéo ou même de forcer l’activation de certains muscles qui ne sont que peu ou pas activés suite par exemple à un AVC.

Ces interfaces peuvent être plus ou moins invasives selon leur utilisation. Les ICO invasives sont constituées d’une grille d’électrodes implantée directement dans le cortex, proche d’une population de neurones que l’on souhaite utiliser pour créer des commandes [13]. En contrepartie de leur forte invasivité, ces interfaces peuvent récupérer l’activité de la population de neurones avec une grande précision, pouvant aller jusqu’au contrôle de mouvements fins d’un bras robotique par exemple. Ces ICO sont aujourd’hui très peu utilisées et seulement dans un cadre médicale très stricte [83]. Il existe notamment certaines entreprises privées ayant pour but à moyen/long terme d’ouvrir ce type d’interface au grand public, mais en sont pour l’instant aux test sur les animaux et peinent à faire accepter les tests sur les humains [32]. Les ICO semi-invasives utilisent toujours une grille d’électrode mais cette fois-ci sur la dure-mère, une paroi rigide sous la boîte crânienne [20]. Ce type de méthode permet de ne pas subir la réduction de signal due à l’os qui compose la boîte crânienne tout en ayant des risques très limités pour le cerveau. On obtient alors des résultats assez proches de ceux ob-

tenus avec les ICO beaucoup plus invasives mais avec une intervention médicale moins importante, bien que loin d'être anodine. L'utilisation la plus connue en France est le projet ICO développé par Clinatex et ayant pour but d'aider les personnes atteintes d'un handicap moteur lourd (notamment tétraplégiques) à se déplacer et réaliser des mouvements via le contrôle par ICO d'un exosquelette [10]. Finalement les ICO les plus utilisées aujourd'hui sont non-invasives. Pour celles-ci il suffit d'utiliser un bonnet EEG afin de récupérer les signaux provenant du cerveau de l'utilisateur. Ces ICO ont l'avantage d'être les plus abordables en plus d'être sans risque pour l'utilisateur. En revanche leur résolution spatiale est mauvaise, de même que leur rapport signal/bruit permettant pour le moment des commandes très limitées. Ce type d'ICO étant les plus répandues, cette thèse sera entièrement liée à son utilisation. Ainsi les problèmes de résolution spatiale ainsi que ceux du rapport signal/bruit font partie intégrante de la recherche.

Dans le cadre des ICO non-invasives, les signaux électriques enregistrés par l'EEG sont liés à une activité cérébrale généralement localisée autour d'un groupe d'électrode. Il est ainsi possible en connaissant le groupe d'électrodes lié à une activité cérébrale donnée, de créer des commandes réagissant à ce pattern. Afin de générer des signaux facilement reconnaissables, il est possible de suivre des séries d'instructions précises. Ces séries d'instructions sont appelées paradigmes, les plus connus en interface cerveau-ordinateur (ICO) sont : l'imagerie motrice ou motor imagery (MI) en anglais, les potentiels évoqués ou event related potentials (ERP), les potentiels évoqués visuels stables ou steady-state visual evoked potentials (SSVEP) et plus récemment les potentiels évoqués visuels modulés par code ou code-modulated visual evoked potentials (cVEP).

Dans le cas de la MI, l'utilisateur doit imaginer le mouvement d'une partie du corps. Cette imagerie motrice va créer chez l'utilisateur une réponse localisée dans la partie du cortex sensorimoteur liée à ce membre et autour des fréquences mu (de 8 à 13Hz) et beta (de 13 à 30Hz) [66] [60]. Les plus utilisés sont les mains, les pieds, la langue car ils sont situés à des positions éloignées les uns des autres sur le cortex sensorimoteur, permettant ainsi de faciliter la classification des commandes. La classe "repos" est également souvent utilisée. Elle correspond à un état où le sujet n'imagine aucun mouvement, que ce soit pour créer une classe "non commande" où rien ne se passe pour l'ICO ou une autre classe à part entière. Ce paradigme peut permettre par exemple de contrôler un fauteuil roulant en associant chacune des commandes à une direction. Une séance type d'ICO en MI est constitué d'images représentant la classe à imaginer qui s'affichent successivement à l'écran avec généralement une petite pause entre chaque (notamment pour laisser le sujet cligner des yeux ou autre afin de ne pas parasiter les signaux lors de l'exécution) comme représenté en figure 1.2.

Les ERP fonctionnent très différemment. Ce paradigme est lié à la réaction du cerveau à certains événements. Le but est de créer un événement, généralement visuel ou auditif dont la réaction au niveau du cerveau se caractérisera par un pic de tension dans l'activité électrique du cerveau appelé "potentiel évoqué". Le cas le plus utilisé en ERP dans le cadre des ICO est le P300. Lors d'une ICO en P300 plusieurs motifs sont affichés à l'écran, chacun correspondant à une commande. Lors de la sélection, il est demandé à l'utilisateur de maintenir

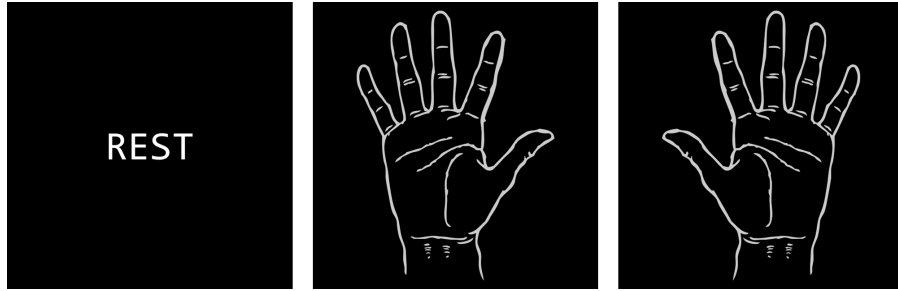


FIGURE 1.2 – Déroulement classique d’une séance de calibration pour l’utilisation d’une ICO en MI. L’écran va afficher successivement les différents états d’imagerie mentale demandés au sujet (ici repos, main droite et main gauche) afin de créer une base de donnée permettant par la suite de les reconnaître lorsqu’ils seront imaginés sans instruction dans le but de commander l’ICO.

sont regard sur le motif correspondant à la commande qu’il souhaite effectuer. Les motifs vont alors flasher à tour de rôle et un potentiel évoqué va apparaître dans la région occipitale du cerveau environs 300 millisecondes après le flash, permettant ainsi de savoir quelle commande l’utilisateur souhaite effectuer. Ce type d’interface cerveau ordinateur est beaucoup utilisé pour épeler des mots afin de communiquer, comme présenté en Figure 1.3. Dans ce cas, une grille contenant un alphabet est affichée à l’écran et les lettres vont flasher, permettant alors de sélectionner une lettre parmi un grand nombre de possibilités.

Le fonctionnement des SSVEP est relativement proche de celui des ERP avec un nombre généralement plus faible de motifs dont chacun clignote à une fréquence différente. Ainsi lorsque l’utilisateur regarde un motif précis, des potentiels apparaissent à la même fréquence que celle à laquelle le motif clignote, créant ainsi une réponse électrophysiologique dont la fréquence suffit à déterminer la commande souhaitée. Leur utilisation se rapproche quant à elle plutôt du paradigme de MI car il se fait dans le cadre d’un signal stable avec un nombre relativement faible de commandes. Ce paradigme présente cependant quelques défauts. En effet ceux-ci reposent sur l’utilisation de flash lumineux clignotants qui peuvent être visuellement fatiguants voire même présenter un risque de crise épileptique.

Les cVEP sont un type de paradigme inspiré des SSVEP. L’idée est la suivante : si les flash des motifs se font non pas en suivant une fréquence constante mais en suivant un paterne unique il est possible d’obtenir une réponse unique du cerveau pour chaque motif, un peu comme un code-barre. Ainsi, là où les réponses générées par les SSVEP peuvent être confondues si les fréquences sont trop proches, les réponses des cVEP sont unique et donnent donc généralement de meilleurs résultats. Par ailleurs, leur détection peut se faire relativement rapidement et avec des stimulations beaucoup moins agressifs que ceux des SSVEP.



FIGURE 1.3 – Déroulement classique de l’utilisation d’un épeleur P300. Successivement chaque ligne puis chaque colonne est mise en surbrillance. Il est dit à l’utilisateur de se concentrer sur le flash lumineux lié à la ligne ou colonne où se situe la lettre qu’il souhaite écrire afin de déclencher un potentiel évoqué 300 ms après le flash. La ligne et la colonne sont ensuite déterminées afin de trouver la lettre choisie et de l’écrire

1.3 Pré-traitement

Les signaux récupérés par l’EEG sont représentés sous forme de tableau numérique à deux dimensions étant respectivement le nombre de canaux (représentant chacun une électrode) et le nombre d’échantillons temporels collectés à la fréquence d’échantillonnage f_s . Lors de l’enregistrement des signaux provenant de l’EEG, la totalité de la session va être enregistrée d’un bloc puis stockée.

1.3.1 Le filtrage fréquentiel

Suite à cela, un filtrage fréquentiel est effectué. Celui-ci peut prendre plusieurs formes :

- Premièrement, en Europe les lignes électriques génèrent un champ électrique à 50 Hz. Ce champ génère du bruit dans les signaux EEG et doit donc être filtré afin de ne pas poser de problème lors du traitement de signal. Un filtre coupe-bande à 50 Hz est donc appliqué sur les séries temporelles de l’EEG.
- Ensuite, comme expliqué en 1.2, chaque paradigme possède une ou plusieurs bandes de fréquences préférentielles qui vont être mises en avant en utilisant un ou plusieurs filtres passe-bande.

Une fois les filtres fréquentiels appliqués, la session est découpée pour correspondre aux commandes envoyées au système. Chacune d’entre elles est représentée par une série temporelle multi-canaux $\mathbf{X}_{i \in [1, N_t]}$ de taille (N_c, N_s) avec N_c le nombre de canaux et N_s le nombre d’échantillons (samples en anglais) de la série temporelle. Cette série temporelle est usuellement appelée un *essai* (ou *trials* en anglais) ; on note N_t le nombre d’essais. L’ensemble des essais est noté par un tableau multidimensionnel \mathbf{X} de taille (N_t, N_c, N_s) . On lui associe

généralement un vecteur de labels \mathbf{y} dont les coefficients y_i correspondent à la classe de l'essai X_i , c'est à dire la tâche mentale exécutée pendant cet essai.

1.3.2 Le filtrage spatial

Lors du traitement des signaux récoltés plus tôt, certains canaux peuvent avoir une plus grande importance que d'autres dans la prise de décision. En effet, il arrive que certains canaux soient trop bruités pour que l'information qu'ils apportent aide, parfois tellement qu'au contraire les utiliser contaminerait l'information apportée par les autres canaux. De la même façon, une électrode située proche du cortex sensori-moteur va généralement donner des informations bien plus pertinentes qu'une électrode frontale dans le cadre d'imagerie motrice. Il en va de même pour tous les autres paradigmes ; chacun possède son set d'électrodes préférentiel permettant d'extraire des informations pertinentes. Cette étape de sélection d'électrodes pertinentes est appelée le filtrage spatial. Elle peut aller du simple choix "à la main" de jeu d'électrodes considérées comme pertinentes à l'utilisation d'algorithmes supervisés permettant d'extraire les canaux maximisant la séparation des classes à différentier de façon automatique. L'une des méthodes les plus connues est l'Analyse en Composantes Principales ou Principal Components Analysis (PCA) [1] qui permet d'identifier les vecteurs principaux de variabilité dans un espace de caractéristiques comme montré en figure 1.4. Pour cela considérons un espace de caractéristiques. Le but est d'extraire ces vecteurs appelés composantes principales, ou a minima de les trier de la composante la plus importante à la plus faible. Ces composantes sont extraites de telle sorte à créer une base orthonormée de l'espace de base mais sur laquelle on a effectué une rotation pour correspondre à l'ordre d'importance des composantes principales. Cette méthode s'avère très utile, notamment lorsqu'il s'agit de réduire la taille d'un espace de caractéristiques avec un grand nombre de dimension, ou encore de comprendre quelles sont les caractéristiques pertinentes pour l'étude d'un phénomène. Dans le cas des ICO cela permet de déduire par exemple certains ensembles de canaux qui varient plus fort que d'autres lors de l'exécution des tâches mentales, cela peut aussi permettre d'observer les canaux variant de manière trop importante et qui peuvent être synonymes de bruit ou d'autres problématiques liées à l'EEG en lui même ou au sujet (décollement d'électrode, mauvaise impédance, contraction musculaire visible dans l'EEG, etc...).

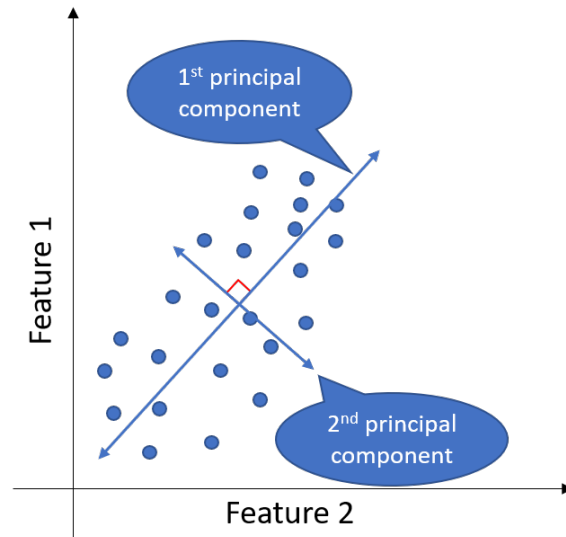


FIGURE 1.4 – Représentation schématique du fonctionnement d’une PCA.

Une autre méthode de filtrage spatiale utilisée en ICO est l’Analyse en Composante Indépendante ou Independent Component Analysis (ICA) [23]. Cette méthode est généralement utilisée pour décomposer un signal réel en plusieurs composantes indépendantes. La façon classique d’expliquer l’utilité de cette méthode est de considérer qu’elle se place dans le domaine de la séparation de source. C’est à dire que son but est de séparer des signaux provenant de sources différentes grâce à des capteurs qui vont capter tous les signaux de façon plus ou moins forte. L’exemple le plus connu est de se placer dans une salle avec plusieurs personnes qui parlent et de dire qu’on a autant de micros à des endroits différents que de personnes qui parlent. Ces micros vont capter des sons venant de chacune des personnes mais de façon un peu différente ; soit décalés dans le temps, soit simplement modulés avec une certaine intensité liée à leur éloignement au micro. Dans le cadre de l’ICA, seule la différence d’intensité est prise en compte mais cette différence permet d’identifier de façon précise ce que dit chacune des personnes et de séparer leur discours, pour peu qu’il y ait au moins autant de micro que de personnes. Son utilisation principale en ICO a pour but d’identifier la part du signal EEG liée à des signaux non cérébraux (notamment les mouvements oculaires) afin de pouvoir l’éliminer pour améliorer la qualité du signal. [2]

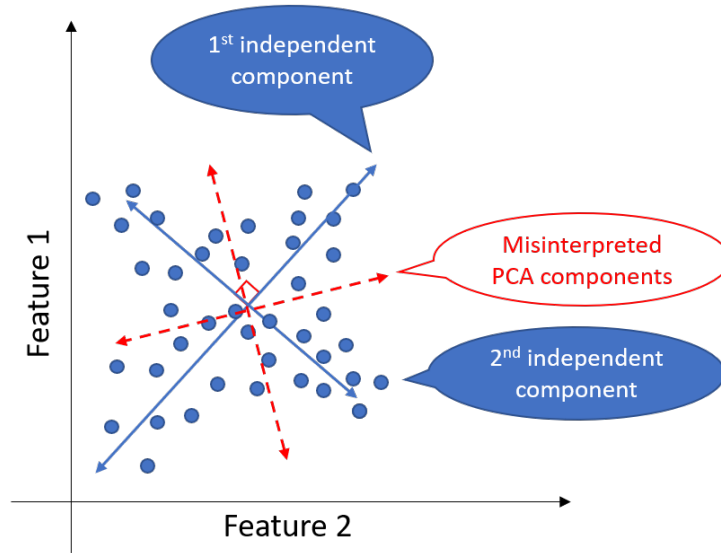


FIGURE 1.5 – Représentation schématique du fonctionnement d’une ICA. À la différence de la PCA, l’ICA s’intéresse surtout à récupérer des composantes indépendantes plutôt que des composante maximisant la variance.

Par ailleurs, en ICO la méthode qui donne généralement les meilleurs résultats lors de la recherche de structures spatiales dans les signaux est généralement le Common Spatial Pattern (CSP) [47], [14], [35]. Celui-ci sélectionne les électrodes maximisant le rapport entre la variance d’une classe A et celle d’une classe B puis sélectionnant ensuite d’autres électrodes qui minimisent ce rapport (c’est à dire maximisent l’inverse de ce rapport). Expliqué grossièrement, cela revient à chercher les électrodes sur lesquelles "la classe A a un fort impact par rapport à B" puis celles où "la classe B a un fort impact par rapport à A" pour faciliter la classification à venir. Pour rentrer dans les détails, considérons deux fenêtres de signal \mathbf{X}_A et \mathbf{X}_B représentatifs des classes A et B ainsi qu’une pondération \mathbf{w} . On cherche à déterminer notre pondération \mathbf{w} de telle sorte à maximiser le ratio de variance entre les deux, c’est à dire

$$\mathbf{w} = \arg \max_{\mathbf{w}} \left(\frac{\|\mathbf{w}\mathbf{X}_A\|^2}{\|\mathbf{w}\mathbf{X}_B\|^2} \right) \quad (1.1)$$

$$= \arg \max_{\mathbf{w}} \left(\frac{\mathbf{w}\mathbf{X}_A\mathbf{X}_A^T\mathbf{w}^T}{\mathbf{w}\mathbf{X}_B\mathbf{X}_B^T\mathbf{w}^T} \right). \quad (1.2)$$

Qui est la même chose que

$$\mathbf{w} = \arg \max_{\mathbf{w}} \left(\frac{\mathbf{w}\mathbf{C}_A\mathbf{w}^T}{\mathbf{w}\mathbf{C}_B\mathbf{w}^T} \right) \quad (1.3)$$

avec \mathbf{C}_A et \mathbf{C}_B les matrices de covariance respectives des classes A et B. Dans les fait, comme obtenir les fenêtres représentatives des classes A et B peut-être complexe, on préfère l'écriture en équation 1.3 car elle permet de considérer comme matrices de covariance représentatives de chacune des classes leur matrices de covariance moyennes calculées de la manière suivante :

$$\mathbf{C}_k = \frac{1}{N_s} \mathbf{X}_k \mathbf{X}_k^T, \quad k \in \{A, B\}. \quad (1.4)$$

Dans le cas général où nous ne possédons pas de fenêtre de signal représentatif des classes A et B, la matrice de covariance correspondante à chacun des classes est calculée en faisant la moyenne des matrices de covariances sur chacun des essais de la classe, donnant alors :

$$\mathbf{C}_k = \frac{1}{N_k} \sum_{y_i=k} \frac{1}{N_s} \mathbf{X}_i \mathbf{X}_i^T, \quad k \in \{A, B\}, \quad (1.5)$$

avec N_k le nombre d'essais présents dans la classe $k \in \{A, B\}$ et y_i le label de la classe correspondant à l'essai i .

Pour résoudre le problème donné en équation 1.3, en reprenant les articles [47] et [68], on considère la matrice de covariance composite

$$\mathbf{C}_{AB} = \frac{1}{2}(\mathbf{C}_A + \mathbf{C}_B) \quad (1.6)$$

que l'on décompose en vecteurs propres tel que

$$\mathbf{C}_{AB} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T \quad (1.7)$$

avec \mathbf{U} la matrice de vecteurs propres et $\mathbf{\Lambda}$ le vecteur de valeurs propres de la matrice de covariance composite. On peut alors calculer la transformation du blanchiment

$$\mathbf{P} = \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{U}^T \quad (1.8)$$

qui égalise les variances de l'espace propre représenté par \mathbf{U} . Cette transformation peut être alors simplement appliquée aux matrices de covariances \mathbf{C}_A et \mathbf{C}_B de la manière suivante

$$\mathbf{S}_k = \mathbf{P} \mathbf{C}_k \mathbf{P}^T, \quad k \in \{A, B\}. \quad (1.9)$$

On obtient alors que les matrices partagent les même vecteurs propres et que la somme des valeurs propres correspondante sera toujours 1 [34]. C'est à dire qu'en considérant une matrice de vecteurs propres \mathbf{Q} de \mathbf{S}_A , on obtient que

$$\mathbf{S}_A = \mathbf{Q} \mathbf{\Lambda}_A \mathbf{Q}^T \quad (1.10)$$

mais également

$$\mathbf{S}_B = \mathbf{Q} \mathbf{\Lambda}_B \mathbf{Q}^T \quad (1.11)$$

avec $\mathbf{\Lambda}_A + \mathbf{\Lambda}_B = \mathbf{I}_{N_c}$.

Ce résultat est particulièrement intéressant car cela signifie que si l'on considère les valeurs de Λ_A rangées dans l'ordre croissant, les premiers vecteurs propres de \mathbf{Q} maximisent la variance au niveau de la classe A et, étant donné la contrainte imposée aux valeurs propres, minimise la variance au niveau de la classe B. À l'inverse, les derniers vecteurs seront ceux qui minimiseront la variance de A et maximiseront celle de B. On obtient alors une matrice de projection \mathbf{W} telle que

$$\mathbf{W} = \mathbf{Q}^T \mathbf{P}. \quad (1.12)$$

Le premier vecteur de \mathbf{W} sera ainsi le \mathbf{w} recherché dans l'équation 1.3 tandis que le dernier correspondra à la maximisation du ratio inverse, nous permettant d'obtenir les pondérations d'électrodes recherchées.

En prenant un petit exemple 2D, permettant de visualiser ces équations géométriquement, cela revient à calculer les variances pour obtenir un résultat proche de la figure 1.6 où on remarque que la variance 1 pour les deux classes ($V_{1,A}$ et $V_{1,B}$) maximise le rapport

$$r = \frac{V_{i,A}}{V_{i,B}},$$

tandis que la variance 2 des deux classes minimise ce rapport (et donc maximise sont inverse). Évidemment il s'agit ici d'un exemple 2D pour une méthode de filtrage spatial donc il est difficile de voir un grand apport à cette sélection mais dans les faits, lorsque beaucoup de dimension sont en jeu, pondérer les canaux à l'aide de ce rapport de variance permet de grandement améliorer la précision d'un système.

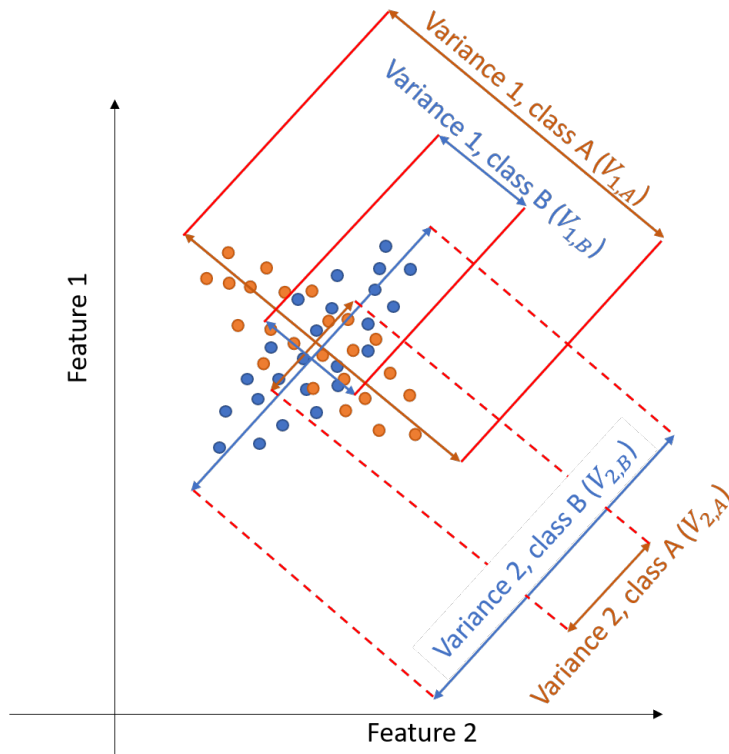


FIGURE 1.6 – Représentation schématique du fonctionnement d'un CSP.

1.4 Matrice de covariance et géométrie riemannienne

Dans le cadre de la géométrie Riemannienne, une matrice de covariance est ensuite calculée pour chaque essai. De part les caractéristiques recherchées, la méthode de calcul des matrices de covariance varie d'un paradigme à l'autre. Ces méthodes sont donc décrites ci-après. C'est là qu'intervient le concept de géométrie Riemannienne pour duquel l'intérêt de la communauté scientifique en ICO ne fait que croître depuis quelques années [98]. Ce type de géométrie prend place car une matrice de covariance n'est pas un objet mathématique classique. Ce type de matrice appartient au domaine des matrices symétriques définies positives (SDP) (ou semi-définies positives si on considère que les signaux peuvent être redondants et donner des matrices à rang non plein). Les matrices SDP possèdent des caractéristiques particulières ne leur permettant pas d'être considérées comme un espace euclidien. En effet, rien que la soustraction de deux matrices SDP, ou la multiplication d'une matrice SDP par un scalaire réel peut donner une matrice non SDP. Le domaine des matrices SDP est en fait une

variété riemannienne [12]. Cette variété, même si elle empêche beaucoup de calculs classiques qu'on retrouve dans un espace euclidien a tout de même certaines propriétés très pratiques. Notamment on peut y calculer la distance entre deux points de telle sorte à ce que tous les points le long du chemin le plus court reliant deux points reste dans la variété. Le nom de la courbe de longueur minimale qui respecte cette règle est appelée la géodésique. L'exemple le plus connu pour expliquer ce concept est de prendre la planète Terre. Et sur cette planète, il existe plusieurs façon d'estimer la distance entre deux villes. On pourrait simplement prendre la distance euclidienne qui les sépare, la ligne droite qui passe au travers de la planète pour estimer cette distance, cependant on voit bien que pour estimer le trajet, même à vol d'oiseau d'une ville à l'autre, cette "ligne droite" n'est pas la meilleure distance. L'autre façon d'estimer cette distance est de suivre la courbe de la planète pour donner une estimation du trajet "à vol d'oiseau" le plus court qui sépare ces deux villes. Ce trajet là est exactement ce qu'on appelle la géodésique dans le cadre de la variété que représente la surface de la planète Terre. De la même manière on calcule la distance entre deux matrices de covariance en suivant la géodésique le long de la variété riemannienne des matrices SDP. Cette distance a permis de donner lieu à différents algorithmes qui vont être étudiés plus loin dans cette introduction.

1.4.1 Matrice de covariance en MI

Le cas de l'imagerie motrice est le plus simples de ceux qui vont être expliqués dans ce manuscrit. On considère les données \mathbf{X} filtrées et découpées en plusieurs essais. Dans le cadre de la géométrie riemannienne, chaque essai \mathbf{X}_i de taille (N_c, N_s) est alors centré puis transformé en matrice de covariance \mathbf{C}_i . La façon la plus simple est de passer par l'équation

$$\mathbf{C}_i = \frac{1}{N_s} \mathbf{X}_i \mathbf{X}_i^T. \quad (1.13)$$

Il s'agit de l'estimateur du maximum de vraisemblance de la matrice de covariance. Il est aussi très fréquent de diviser par $N_s - 1$ plutôt que N_s afin d'obtenir un estimateur non biaisé de la matrice de covariance. Cependant il existe de nombreuses autres estimations de la matrice de covariance, notamment l'estimateur Oracle Approximating Shrinkage (OAS) [21] qui a été très largement utilisée dans les résultats présentés dans cette thèse car très performant. Mathématiquement celui-ci est défini de la manière suivante

$$\hat{\mathbf{C}}_i = (1 - \rho) \mathbf{C}_i + \rho \frac{\text{tr}(\mathbf{C}_i)}{N_c} \mathbf{I}_{N_c} \quad (1.14)$$

avec ρ le coefficient de rétrécissement dont le calcul est détaillé dans l'article [21]. Ces estimateurs sont généralement plutôt des façons de corriger la matrice de covariance des échantillons calculée plus tôt. Ils permettent notamment d'assurer certaines propriétés recherchées pour les matrices de covariance, comme par exemple la définie-positivité. En effet, il arrive régulièrement que des données réelles donnent lieu à des matrices de covariance qui ne sont pas définie-positives,

c'est à dire que certaines valeurs propres de la matrices sont nulles voire parfois même négatives (bien que les valeurs négatives vont généralement être très proches de 0 et liées à des approximations de calcul machine). Cela peut arriver notamment lorsque le gel placé au niveau des électrodes de l'EEG se propage d'une électrode à l'autre et les lie électriquement, ce qui donne lieu à deux canaux qui ont quasiment voire totalement les mêmes signaux, donnant lieu à une valeur propre nulle dans la matrice de covariance (car deux lignes/colonnes ont les mêmes valeurs). La non positivité stricte d'une matrice pose généralement un vrai problème dans la suite des calculs en géométrie riemannienne car elle fait perdre certaines propriétés utiles qui vont parfois générer des erreurs lors de l'exécution du code qui souhaite les utiliser, donnant lieu à de longues journées de travail perdues à essayer de comprendre des erreurs dont il est difficile de trouver la source. Le pré-traitement appliqué aux signaux EEG dans le cadre de la MI est résumé en figure 1.7.

1.4.2 Matrice de covariance en ERP

Pour les ERP, le calcul de matrices de covariance classiques effectué pour les MI est sous-optimal de par les caractéristiques recherchées. En effet, en ERP on souhaite trouver le potentiel évoqué qui est très caractéristique sur le plan temporel mais la matrice de covariance calculée ici donne seulement des informations spatiales. Il faut donc trouver un moyen que les informations spatiales puissent nous donner des informations sur la présence ou non du potentiel évoqué sur le plan temporel. Pour cela la technique qui a été imaginée a été d'estimer le potentiel évoqué caractéristique. Le potentiel évoqué caractéristique représente la réponse attendue lorsque l'événement choisi a lieu (par exemple le flash lumineux de la colonne correspondante à la lettre choisie dans le cadre d'un épeleur P300). La façon la plus simple d'obtenir cette réponse est le calcul de la moyenne des différents essais dans lesquels cette réponse est censée avoir eu lieu. Cela permet d'obtenir une réponse caractéristique relativement propre et peu bruitée [9]. On considère \mathbf{y} le vecteur des labels des classes de l'ensemble des essais \mathbf{X} et y_i le label numéro i de \mathbf{y} correspondant donc à l'essai i de \mathbf{X} . On notera également $+$ la classe "target", à savoir la classe cible qui correspond à l'événement donnant lieu à une réponse caractéristique et $-$ la classe "non-target" qui correspond à tous les essais dans lesquels l'événement n'a pas lieu. Le calcul de ce potentiel évoqué $\bar{\mathbf{X}}^+$ se fait en moyennant les essais appartenant à la classe "target" tel que

$$\bar{\mathbf{X}}^+ = \frac{1}{N_+} \sum_{i, y_i \in +} \mathbf{X}_i, \quad (1.15)$$

avec N_+ le nombre d'essais dans la classe "target". Cette réponse caractéristique $\bar{\mathbf{X}}^+$ est ensuite ajouté à chaque essai (quelle que soit sa classe) comme schématisé dans la figure 1.7 où le potentiel évoqué caractéristique est en vert clair tandis que les essais sont en bleu foncé. On notera \mathbf{X}_i^+ l'essai \mathbf{X}_i augmenté avec le potentiel évoqué caractéristique $\bar{\mathbf{X}}^+$ via la concaténation

$$\mathbf{X}_i^+ = \begin{bmatrix} \mathbf{X}_i \\ \overline{\mathbf{X}}^+ \end{bmatrix}. \quad (1.16)$$

Une fois ce trial augmenté calculé (et parfois même avant), un filtrage spatial est calculé de telle sorte à réduire la dimension de l'essai mais également du potentiel évoqué caractéristique tout en améliorant leur pertinence dans le cadre de la classification à venir. La matrice de covariance \mathbf{C}_i est ensuite calculée de façon analogue à celle des matrices de covariance en MI, à savoir

$$\mathbf{C}_i = \frac{1}{N_s} \mathbf{X}_i^+ \mathbf{X}_i^{+T}. \quad (1.17)$$

Cette nouvelle matrice de covariance augmentée possède des caractéristiques particulièrement utiles pour reconnaître le potentiel évoqué caractéristique, notamment dans la covariance entre l'essai de base \mathbf{X}_i et le potentiel évoqué caractéristique $\overline{\mathbf{X}}^+$ qui apparaît en turquoise dans la figure 1.7 [9]. En effet, lorsque l'essai \mathbf{X}_i possède des signaux se rapprochant fort de la réponse caractéristique $\overline{\mathbf{X}}^+$, la covariance entre les deux va alors être plus élevée que dans un essai classique, ce qui permet de dire qu'on a potentiellement affaire à un essai "target". Il arrive même lorsque la matrice de covariance n'est pas trop retravaillée de ne prendre que cette partie turquoise pour effectuer la classification car les informations les plus déterminantes pour la classification vont généralement s'y trouver. Dans le cadre de cette thèse, la matrice de covariance est ensuite assez retravaillée donc on fait le choix de la garder complète même s'il est évident que certaines informations sont inutiles, notamment la partie vert clair de la matrice de covariance qui est la covariance entre la réponse caractéristique et elle-même, ce qui donne une matrice qui n'a aucune variation d'un essai à l'autre, la rendant inutile au possible voire noyant un peu les informations utiles dans d'autres nombres qui restent constants.

1.4.3 Matrice de covariance en SSVEP

En SSVEP la matrice de covariance n'est pas non plus simple comme ce qu'on peut retrouver en MI. Pour comprendre comment on arrive au résultat final, il faut reprendre ce qui est recherché lors de l'estimation de la classe en SSVEP. En SSVEP on cherche à estimer des classes de fréquence dans les signaux EEG reçus. En effet, comme expliqué plus tôt, la fréquence de clignotement à l'écran de la forme représentant la commande souhaitée doit se retrouver ensuite dans l'EEG. Cela signifie qu'on a autant de fréquences d'intérêt que de classes possibles. Ainsi arrive la mise en place de ce qu'on appelle un banc de filtre ou filter bank en anglais. Ce banc de filtre est constituée de plusieurs filtres fréquentiels qui vont être appliqués de façon indépendante à une copie de chacun des essais. Dans les faits, en considérant un essai \mathbf{X}_i et des filtres à respectivement, fréquence f_1 jusque $f_{N_{class}}$ Hz, avec N_{class} le nombre de classes différentes à classifier, on peut obtenir N_{class} nouveaux essais $\mathbf{X}_i^{f_1}$ jusque $\mathbf{X}_i^{f_{N_{class}}}$ filtrés chacun indépendamment autour de sa fréquence de référence. Ces essais filtrés

sont ensuite concaténés un peu à la manière de l’essai augmenté en ERP, donnant un nouveau type d’essai augmenté $\mathbf{X}_i^{[f_1, \dots, f_{N_{class}}]}$ tel que

$$\mathbf{X}_i^{[f_1, \dots, f_{N_{class}}]} = \begin{bmatrix} \mathbf{X}_i^{f_1} \\ \mathbf{X}_i^{f_2} \\ \dots \\ \mathbf{X}_i^{f_{N_{class}}} \end{bmatrix}. \quad (1.18)$$

Cet essai augmenté est bien évidemment passé par un filtrage spatial permettant d’en réduire la dimension (avant ou après la concaténation, mais préférentiellement avant et appliqué à tous les essais filtrés de la même manière) va ensuite permettre de calculer directement la matrice de covariance \mathbf{C}_i toujours via l’estimateur de maximum de vraisemblance, telle que

$$\mathbf{C}_i = \frac{1}{N_s} \mathbf{X}_i^{[f_1, \dots, f_{N_{class}}]} \mathbf{X}_i^{[f_1, \dots, f_{N_{class}}]T}. \quad (1.19)$$

Il est cependant important de noter que le meilleur estimateur permettant de corriger la matrice de covariance peut varier selon le paradigme ou l’application utilisés. Généralement la correction appliquée est précisée avec la méthode dans les articles. Dans le cadre de cette thèse, l’estimateur OAS a été systématiquement utilisé dans les calculs, sauf mention contraire. La matrice \mathbf{C}_i maintenant définie, on obtient une matrice par bloc comme en figure 1.7 dont l’intérêt principal se trouve au niveau des blocs représentant la covariance entre un essai filtré à une fréquence donnée et lui même. En effet, lorsqu’une fréquence doit se trouver dans les données, l’auto-covariance liée à cette fréquence se démarquera par rapport aux autres par une intensité forte, ce qui permettra de reconnaître la classe. Dans les faits il existe deux grandes façon de traiter la matrice \mathbf{C}_i . La première, considérée comme sous-optimale, est de garder la matrice complète, même si cela implique un grand nombre de données peu intéressantes. En effet, seuls les N_{class} blocs diagonaux présentent un réel intérêt pour la classification tandis que les $N_{class} * (N_{class} - 1)$ blocs non-diagonaux ne font potentiellement que noyer l’information des autres. L’autre méthode, plus utilisée bien que complexe à mettre en place est d’extraire les matrices d’auto-covariance de chaque fréquence et les traiter indépendamment, rajoutant une nouvelle dimension dans le traitement des données. Cela permet d’obtenir généralement de meilleurs résultats car les fréquences vont chacune être traitées de façon indépendante et de réduire grandement la taille des données traitées et plus précisément de la diviser par N_{class} . La seconde méthode, bien que très attrayante n’a pas ou très peu été utilisée au cours de cette thèse car elle rajoutait beaucoup de complexité dans la création d’un code permettant de tourner de la même façon sur les trois paradigmes présentés ici, notamment dû à l’ajout d’une dimension complète dans les données qui n’a lieu que pour un seul paradigme. Il est donc important de noter que les résultats obtenus sur les bases de données en SSVEP peuvent être potentiellement encore améliorés notamment par l’utilisation de cette méthode de découpage des matrices de covariance.

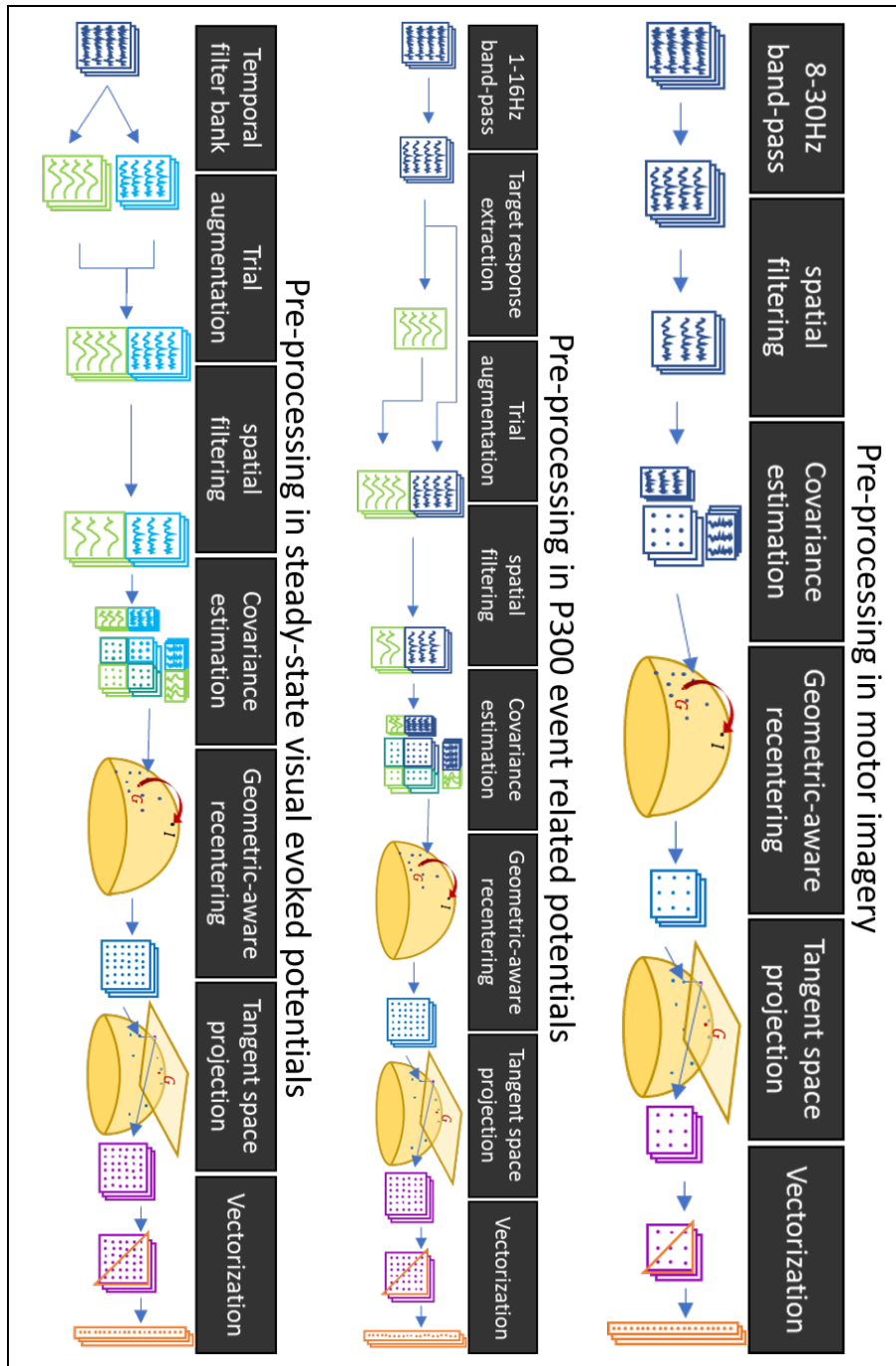


FIGURE 1.7 – Résumé du pré-traitement des données EEG pour les paradigmes MI (haut), P300 (milieu) et SSVEP (bas).

1.4.4 Un point sur la classification dans la variété riemannienne

Une fois la (ou les) matrice de covariance calculée, on se situe comme dit précédemment dans la variété riemannienne des matrices SDP que l'on dénotera $\mathcal{S}_{++}^{N_c}$ en considérant que les matrices seront toutes de taille (N_c, N_c) pour simplifier. L'utilisation de cette variété a été introduite dans le cadre des interfaces cerveau ordinateur pour la première fois par Alexandre Barachant en 2011 au cours de sa thèse [5]. Celle-ci découle de l'utilisation qui en a été faite en détection radar notamment. Après la mise au point d'algorithmes, tels que la minimum distance to Riemannian mean qui sera expliquée plus bas, Alexandre Barachant a participé à plusieurs compétitions de classification de signaux neuronaux sur internet qu'il a remporté à l'aide de ces méthodes face à des algorithmes golden standard du moment, prouvant la puissance de l'utilisation de cette géométrie. Par la suite ses travaux ont été beaucoup repris et la géométrie riemannienne est devenue un domaine à part entière dans la recherche en ICO. D'autres avancées ont d'ailleurs été faites, que ce soit sur le plan de la classification avec notamment la projection des matrices dans l'espace tangent permettant l'utilisation d'algorithmes de classification fonctionnant dans un espace euclidien ou encore dans le domaine de l'apprentissage par transfert [101], [71] qui sera développé plus tard dans cette thèse car se rapportant de très près au travail de recherche qui sera présenté dans les pages à venir.

Comme on vient de le laisser entendre, cette matrice SDP peut désormais être utilisée de nombreuses manières différentes. La première qui a été utilisée est la classification dans la variété. Sur la variété des matrices SDP la géodésique est unique et existe toujours. La distance entre deux points est donc toujours définie et se traduit par la distance de Riemann δ_R définie par

$$\delta_R(\mathbf{C}_1, \mathbf{C}_2) = \|\text{Log}(\mathbf{C}_1^{-\frac{1}{2}} \mathbf{C}_2 \mathbf{C}_1^{-\frac{1}{2}})\| = \sqrt{\sum_{i=1}^{N_c} \log^2(\lambda_i)} \quad (1.20)$$

où les λ_i sont les valeurs propres réelles et strictement positives de $\mathbf{C}_1^{-\frac{1}{2}} \mathbf{C}_2 \mathbf{C}_1^{-\frac{1}{2}}$. De cette distance découle la moyenne riemannienne qui se définit comme le barycentre riemannien d'un ensemble de matrices de covariance. Pour expliquer plus précisément, considérons un ensemble $\mathcal{C} = \{\mathbf{C}_1, \mathbf{C}_2, \dots, \mathbf{C}_{N_t}\}$ de N_t matrices appartenant à $\mathcal{S}_{++}^{N_c}$. Le barycentre riemannien $\mathbf{R}_{\mathcal{C}}$ de ces matrices minimise la somme du carré des distances à lui. Ce qui donne

$$\mathbf{R}_{\mathcal{C}} = \arg \min_{\mathbf{C} \in \mathcal{S}_{++}^{N_c}} \left(\sum_{i=1}^{N_t} \delta_R^2(\mathbf{C}, \mathbf{C}_i) \right) \quad (1.21)$$

Comme il n'existe pas de forme réduite de cette expression, on utilise soit une valeur approchée de ce barycentre, soit on le calcule de manière itérative [53]. Cette définition du barycentre permet plusieurs opérations en ICO, que ce soit le recentrage dont nous parlerons plus tard ou encore le développement de

l'algorithme de Minimum Distance to Riemannian Mean (MDRM) [7] permettant la classification dans la variété de Riemann et mis en place par Alexandre Barachant dans le cadre des ICO en 2011. Cet algorithme bien que très efficace est en fait très simple. Il s'agit simplement de l'extension riemannienne de l'algorithme de Minimum Distance to Mean. La méthode est la suivante, on considère N_t de points repartis en N_{class} classes. Les points d'une même classe ont généralement tendance à être plus proches les uns des autres, comme schématisé en figure 1.8a. Cela permet de considérer leur barycentre comme relativement représentatif de l'ensemble. Ainsi lorsqu'un nouveau point arrive (figure 1.8b) et doit être classifié, il suffit d'estimer la distance de ce point à chacun des barycentres (figure 1.8c) pour finalement prendre une décision en considérant que la classe la plus probable est la classe dont le barycentre est le plus proche (1.8d), d'où le nom Minimum Distance to Mean. Dans certains cas, il est possible suite à la classification du nouveau point d'estimer à nouveau le barycentre de la classe possédant désormais un nouveau point. Cela peut permettre d'avoir une estimation de plus en plus précise du barycentre mais pose cependant le problème de potentiellement décaler le barycentre en cas d'erreur de classification. Cette méthode très simple a le bon goût de simplement nécessiter une distance ainsi que de fait une estimation du barycentre de chacune des classes pour fonctionner correctement. Ainsi dans le cadre de la variété riemannienne des matrices SDP, cette méthode peut être appliquée et tenir compte de la géométrie particulière de l'espace en considérant la distance Riemannienne ainsi que les barycentres riemanniens de chacune des classes donnant donc le MDRM. Pour se le représenter de façon géométrique il est tout à fait correcte de reprendre la représentation de la planète Terre et des villes dont je parlais plus tôt en l'adaptant un peu et en considérant par exemple des villes qu'on souhaite regrouper en plusieurs groupes proches géographiquement pour des raisons de logistique. Cette méthode est vraisemblablement la première à avoir tiré parti de la géométrie riemannienne dans le cadre des ICO.

1.4.5 De la matrice de covariance au vecteur de caractéristiques

L'algorithme de MDRM, bien que toujours performant, a été par la suite dépassé par de nouvelles méthodes dites "dans l'espace tangent". Ces méthodes ont pour but de tirer parti de l'apport de la diversité des informations liées aux matrices de covariance mais en réduisant les contraintes liées à la représentation dans la variété riemannienne, ce qui permettrait d'étendre facilement le nombre d'algorithmes de classifications utilisables dans ce cadre. Pour cela l'idée est de réussir à passer dans un espace euclidien sans perdre trop de la géométrie de la variété. Et en reprenant l'exemple de la planète Terre, quel est le moyen le plus connu de se représenter le monde sans avoir à balader un globe terrestre avec soi? Le planisphère! Et qu'est-ce qu'un planisphère si ce n'est la projection d'une sphère sur un espace en 2 dimensions? Dans le cadres des ICO, la variété riemannienne des matrices SDP de taille N_c est un "cône" en dimension $\mathbb{R}^{\frac{1}{2}N_c(N_c+1)}$. En effet, les matrices étant symétriques, elles sont totalement

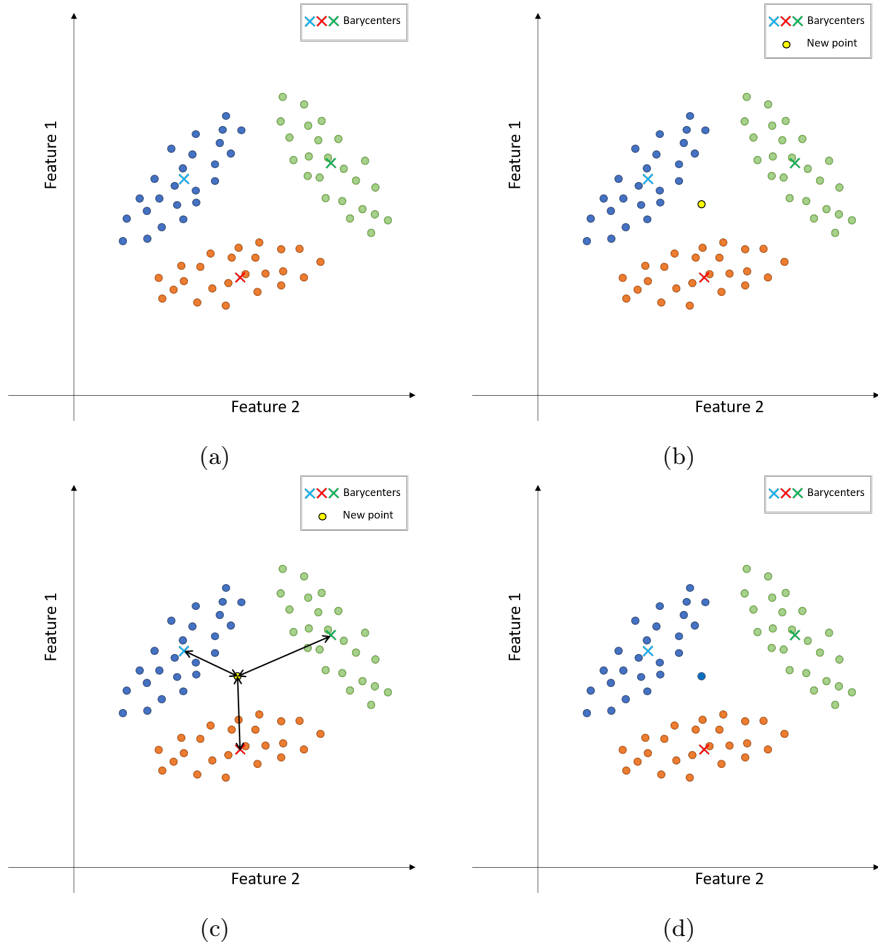


FIGURE 1.8 – Représentation schématique du fonctionnement de l’algorithme de Minimum Distance to Mean (MDM)

définies par leur triangle supérieur droit (ou inférieur gauche), ce qui correspond à $\frac{N_c(N_c+1)}{2}$ termes. Par ailleurs la positivité implique une autre contrainte étant que le déterminant est strictement positif. Pour expliquer pourquoi cette contrainte résulte en un cône, on prend généralement l'exemple des matrices SDP de taille $(2, 2)$ et donc de la forme

$$\mathbf{A} = \begin{pmatrix} a & c \\ c & b \end{pmatrix}$$

Ce type de matrice appartient donc bien à \mathbb{R}^3 car possédant 3 termes indépendants. Ajouter la positivité rajoute la contrainte $ab - c^2 > 0$. Avec les changements de variables $u = \frac{1}{2}(a + b)$ et $v = \frac{1}{2}(a - b)$, on obtient

$$u^2 + v^2 > c^2$$

qui est l'équation d'un cône en 3 dimensions.

En revenant à la forme générale, ce cône est ensuite projeté dans $\mathbb{R}^{\frac{1}{2}N_c(N_c+1)}$ de telle sorte à rapprocher d'une géométrie euclidienne. Pour cela on utilise l'opérateur logarithme matriciel et plus précisément on projette le cône sur l'espace tangent à un point comme schématisé en figure 1.9. La définition de cet espace tangent permet un grand nombre de choses en géométrie riemannienne si bien que la définition même de la distance définie en équation 1.20 sur la variété découle de l'introduction d'une métrique dans l'espace tangent. On utilise généralement comme point de projection le barycentre riemannien \mathbf{R}_C introduit plus haut permettant d'avoir des distances au point de projection minimisées de telle sorte à réduire les erreurs d'estimation liées à de trop grandes distances entre les points. Le choix du barycentre riemannien possède également d'autres avantages notamment dans le cadre de l'apprentissage par transfert en ICO mais cela sera détaillé plus tard. Considérons donc notre ensemble $\mathcal{C} = \{\mathbf{C}_1, \mathbf{C}_2, \dots, \mathbf{C}_{N_i}\}$ de N_i matrices appartenant à $\mathcal{S}_{++}^{N_c}$ ainsi que leur barycentre riemannien \mathbf{R}_C défini plus haut. Lorsque le calcul du barycentre riemannien pose des problèmes (notamment en terme de temps de calcul lié à l'optimisation de l'obtention du minimum), il est également très commun d'utiliser la moyenne Log-euclidienne définie par $\mathbf{R}_{LE} = \text{Exp}\left(\frac{1}{N_i} \sum_i \text{Log}(\mathbf{C}_i)\right)$ ([33]). La projection \mathbf{M}_i d'une matrice \mathbf{C}_i au point \mathbf{R}_C s'écrit

$$\mathbf{M}_i = \mathbf{R}_C^{\frac{1}{2}} \text{Log}\left(\mathbf{R}_C^{-\frac{1}{2}} \mathbf{C}_i \mathbf{R}_C^{-\frac{1}{2}}\right) \mathbf{R}_C^{\frac{1}{2}}. \quad (1.22)$$

La même transformation étant appliquée à toutes les matrices $\mathbf{C}_i \in \mathcal{C}$, cela correspond à un recentrage "riemannien" de l'ensemble telle sorte à positionner le barycentre \mathbf{R}_C à l'identité \mathbf{I}_{N_c} . Les matrices recentrées $\mathbf{R}_C^{-\frac{1}{2}} \mathbf{C}_i \mathbf{R}_C^{-\frac{1}{2}}$ que l'on notera \mathbf{C}'_i sont ensuite envoyées vers l'espace tangent via l'opérateur logarithme matriciel $\text{Log}(\cdot)$. Et finalement les matrices sont renvoyées à une position "proche" de leur toute première position \mathbf{C}_i via la transformation inverse du recentrage tel que $\mathbf{M}_i = \mathbf{R}_C^{\frac{1}{2}} \text{Log}(\mathbf{C}'_i) \mathbf{R}_C^{\frac{1}{2}}$. Cette dernière étape est souvent passée lors du passage à l'espace tangent pour les mêmes raisons d'apprentissage par

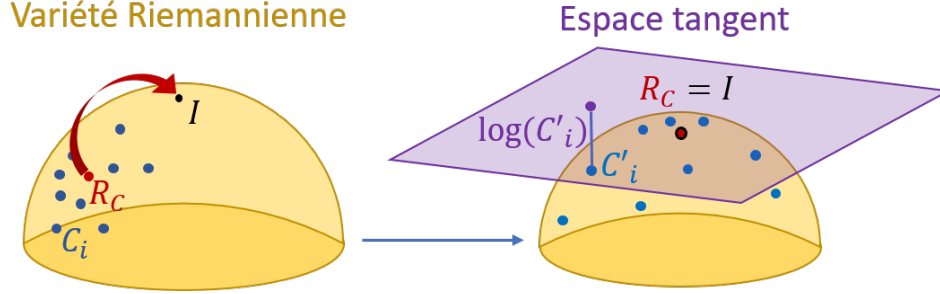


FIGURE 1.9 – Schématisation du recentrage et de la projection dans l'espace tangent à R_C . Les formes utilisées ne sont pas réellement celles que prennent les espaces afin de simplifier la compréhension.

transfert dont j'ai parlé un peu plus tôt. On gardera tout de même la notation \mathbf{M}_i pour noter les matrices projetées, qu'elles soient recentrées ou non. Par la suite, on va préférer avoir des matrices dont la moyenne cette fois-ci arithmétique se situe à $\mathbf{0}_{N_c}$ la matrice nulle carrée de taille N_c . On notera d'ailleurs que suite à l'utilisation de l'opérateur logarithme matriciel $\text{Log}(\cdot)$, le point autour duquel les matrices sont centrées (au sens du barycentre riemannien) n'est plus \mathbf{I}_{N_c} . On observe cependant dans les faits que lorsque les matrices sont centrées (au sens du barycentre riemannien) sur \mathbf{I}_{N_c} avant la projection, elles sont centrées (au sens de la moyenne arithmétique) sur $\mathbf{0}_{N_c}$. En effet, la moyenne riemannienne agit comme une moyenne géométrique sur les matrices et que l'opérateur $\text{Log}(\cdot)$ a donc pour effet que des matrices centrées au 1 matriciel qu'est \mathbf{I}_{N_c} au sens de cette moyenne géométrique sont centrées vers le 0 matriciel qu'est $\mathbf{0}_{N_c}$ au sens arithmétique.

Une fois les matrices envoyées vers l'espace tangent, elles sont désormais traitées comme des matrices symétriques classiques ([63]). À partir de cette étape, on se place dans un espace muni de la norme euclidienne, ce qui permet l'utilisation d'un grand nombre d'algorithmes d'apprentissage automatique. Le fait de traiter avec des matrices symétriques signifie qu'une partie conséquente des données est en double, à savoir les termes non-diagonaux. Pour des raisons calculatoires il est donc usuel de vectoriser ces matrices afin d'obtenir un vecteur de taille $\frac{N_c(N_c+1)}{2}$. Avant de simplement prendre les $N_c(N_c+1)$ termes indépendants, il est préférable de multiplier les termes non-diagonaux par $\sqrt{2}$ de telle sorte à conserver les normes et les distances par rapport aux matrices symétriques. On obtient donc en considérant \mathbf{z}_i le vecteur tangent correspondant à la matrice tangente \mathbf{M}_i

$$\mathbf{z}_i = \text{triu}(\mathbf{M}_i \circ \mathbf{A}),$$

où $\text{triu}(\cdot)$ est l'opérateur qui vectorise la partie triangulaire supérieure d'une matrice, et \mathbf{A} est une matrice ayant la même forme que \mathbf{M}_i , remplie de 1 sur la diagonale et de $\sqrt{2}$ sur la partie hors diagonale. Comme les matrices ont été

préalablement recentrées, les vecteurs résultants sont également recentrés, ce qui signifie que le vecteur tangent moyen est le vecteur nul. Ces vecteurs tangent sont désormais ce qui se rapproche le plus de vecteur de caractéristiques classiques qu'on peut retrouver dans toutes les méthodes d'apprentissage machine usuelles. Cela permet d'utiliser la grande majorité des méthodes connues et qui fonctionnent très bien dans un espace euclidien classique. À un détail près, il est généralement nécessaire d'observer une certaine gausiennité dans les données afin de pouvoir assurer une utilisation correcte de certaines méthodes. Cependant, les transformations qui ont été effectuées jusque là sur les signaux rendent ces vecteurs analogues au logarithme des puissances des signaux, or ce type de vecteurs en interface cerveau ordinateurs (générés suite à un CSP par exemple) ont tendance à se répartir de façon gaussienne dans l'espace de caractéristiques (ou à minima comme des sommes de gausiennes). Cette observation permet donc d'appliquer les méthodes d'apprentissages automatique dans un cadre adapté.

1.4.6 Classification dans l'espace tangent

Trois méthodes sont très utilisées pour classifier des vecteurs tangents, à savoir l'Analyse Discriminante Linéaire ([7]), la Machine à Vecteur de Support ([94]) et la Régression Logistique ([80]). Parmi ces méthodes, la plus connue est probablement l'Analyse Discriminante Linéaire ou Linear Discriminant Analysis (LDA) [41]. La LDA était déjà utilisée sur les vecteurs caractéristiques correspondant au logarithme des puissance des signaux, notamment suite à un CSP. Ce pipeline est d'ailleurs généralement dénommée CSP-LDA et est encore utilisée aujourd'hui pour faire des ICO assez simples sans avoir à rentrer dans des complexités géométriques comme le cadre riemannien. Il était donc assez naturel d'utiliser cette méthode sur les vecteurs tangents \mathbf{z}_i afin de les classifier. Cette nouvelle méthode est dénommée Tangent Space Linear Discriminant Analysis (TSLDA) et a été introduite dans le domaine des ICO par Alexandre Barachant en 2011 [7] et suite à des premières recherches plus exploratoires en 2010 sur la LDA et les vecteurs tangents [6]. L'idée de la LDA est de considérer que chacune des classes i peut être représentée comme une loi de probabilité normale de loi $(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ dans l'espace de caractéristiques. À cela s'ajoutent quelques assumptions en plus, notamment l'homoscédasticité (tous les $\boldsymbol{\Sigma}_i$ sont égaux, ce qui est généralement le cas ou proche de l'être dans notre cas). Le but premier de la LDA est de définir une projection dans l'espace de caractéristique qui maximise l'espace entre les densités de probabilité afin de faciliter la classification à venir comme présenté en figure 1.10. Pour ce faire, celle-ci sélectionne un filtre spatial qui maximise la variance inter-classe par rapport à la variance intra-classe. Dans les faits, cette méthode a ensuite été étendue pour effectuer la classification à l'aide de cette projection. Cela permet de classifier des points étant donné leur probabilité d'appartenance à une classe.

La Machine à Vecteur de Support ou Support Vector Machine (SVM) a dans son cas un fonctionnement très différent [17]. Le but d'une SVM est de créer un hyperplan permettant de séparer au mieux deux classes. Pour cela, cette

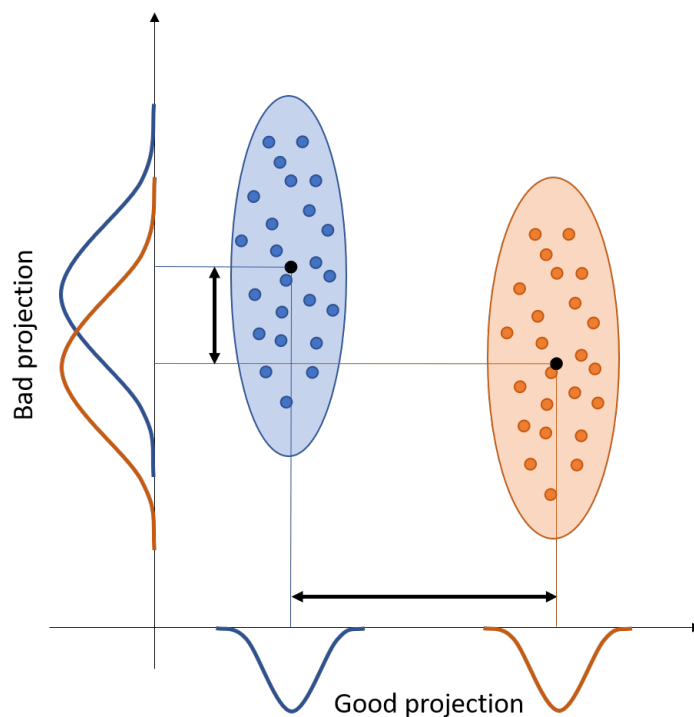


FIGURE 1.10 – Représentation schématisée de la LDA et l'impact de la projection sur la classification à suivre.

méthode s'aide de vecteurs de supports dans chacune des classes qui représentent les vecteurs les plus proches de l'autre classes. On sait que l'hyperplan se placera à équidistance de ces points (plus ou moins modifiée dans le cas de multiples vecteurs de supports et d'autres paramètres) et il suffit désormais de trouver l'orientation que cet hyperplan va suivre. Celle-ci est choisie dans le but de maximiser les marges entre les points de support comme montré en figure 1.11.

La Régression Logistique ou Logistic Regression (LR) [52] quant à elle reprend le cadre dans lequel on se place ici pour déterminer une direction principal permettant la séparation des classes à l'aide d'une régression logistique unidimensionnelle comme vu à gauche dans la figure 1.12. Cette direction principale ainsi que la régression logistique unidimensionnelle calculées, il suffit désormais de les appliquer à tout l'espace de caractéristique, donnant l'espace "colorisé" à droite de la figure 1.12. Il suffit ensuite de placer un seuil permettant de dire à quel classe appartient un nouveau point qui arriverai dans l'espace de caractéristique. Il est important de noter que lors du calcul de la LR on utilise généralement la méthode de contraction LASSO (pour Least Absolute Shrinkage and Selection Operator), ce qui ajoute une contrainte de parcimonie sur les coefficients.

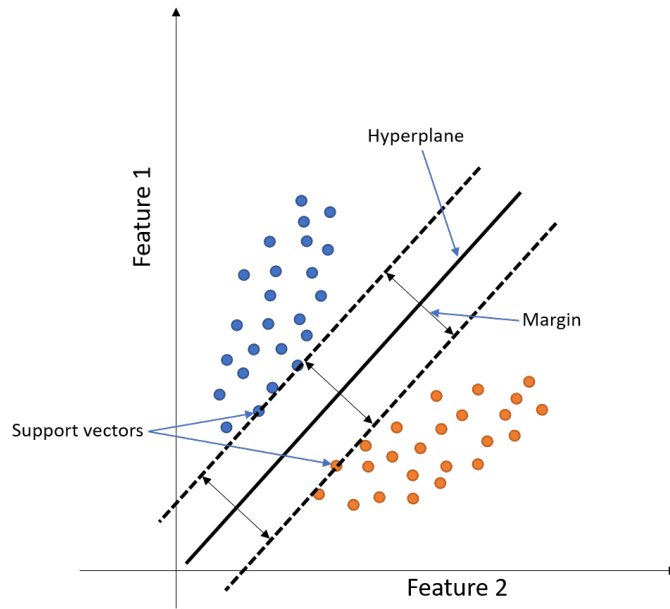


FIGURE 1.11 – Représentation schématisée d'une SVM.

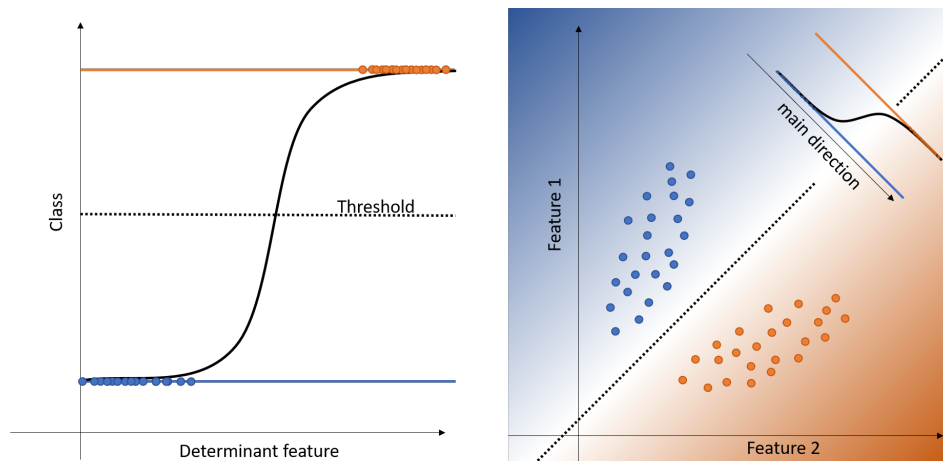


FIGURE 1.12 – Représentation schématisée d'une LR.

Chacune de ces méthodes possède ses avantages et ses inconvénients. Il est aujourd'hui très facile de passer de l'une à l'autre pour vérifier si les résultats obtenus sont meilleurs pour tel ou tel paradigme. Dans le cadre de cette thèse, la majorité des calculs de classification a été faite en utilisant la LR mais une partie conséquente a également été faite à l'aide d'une SVM notamment car dans certains cas les résultats sont de meilleure qualité.

1.5 Apprentissage par transfert

1.5.1 La variabilité en ICO

Aujourd'hui les ICO restent dans leur très grande majorité des dispositifs médicaux utilisés dans des cadres très précis ou au mieux des preuves de concept à destination du grand public mais sont encore loin de pouvoir fonctionner efficacement dans un environnement qui n'est pas optimisé pour leurs performances (peu de mouvements, clignements et mouvements des yeux les plus rares possibles, bruit magnétique ou électrique contrôlés...). Cela est notamment dû aux limitations des EEG actuels, principalement en terme de résolution spatiale et de gestion du bruit électrique ambiant, tout du moins pour le grand public car certaines technologies permettent déjà une utilisation sans fil et avec un bon contrôle du bruit mais leur prix est généralement loin d'être abordable. Par ailleurs les limitations techniques telles que celles que je viens d'évoquer sont loin d'être les seuls problèmes dans la simplification de l'utilisation des ICO aujourd'hui. En effet, une des problèmes majeurs des ICO est la variabilité des données [73]. Cette variabilité peut prendre de nombreuses formes dont je ne vais présenter ici qu'une liste non exhaustive.

Tout d'abord, il semble assez logique que d'une expérience à l'autre, même si le paradigme reste le même, le changement de configuration, que ce soit dans les instructions données au sujet, les feedbacks qui lui sont envoyés ou encore plus simplement les différences dans le matériel utilisé, vont avoir un impact non trivial sur les données récoltées. Cet impact peut aller d'un décalage visible dans les signaux à une hétérogénéité complète des données possédant un nombre de canaux différents, représentant des positions différentes sur le scalp avec une fréquence d'échantillonnage différente, des amplitudes plus ou moins élevées et une électrode de référence qui passe du point le plus central du scalp (Cz) au point du visage le plus éloigné possible ou presque qu'est le lobe de l'oreille. On parle alors de variabilité inter-configurations ou cross-setup variability. Ce type de variabilité est probablement le plus difficile à gérer en ICO étant donné l'hétérogénéité potentielle des données mais est également l'étape la plus attrayante à franchir car elle pourrait permettre d'utiliser tous les jeux de données sur internet et de croiser pour obtenir l'information la plus précise et robuste possible.

Ensuite il existe un type de variabilité déjà beaucoup plus simple à gérer bien que tout de même non trivial. Il s'agit de la variabilité inter-sujet ou cross-subject variability. Comme son nom l'indique, ce type de variabilité a lieu d'un

sujet sur l'autre. En effet, pour une même expérience avec des conditions expérimentales très proches, les physiologie d'un sujet va avoir un impact très important sur les données. Cela s'explique de différentes manières. Tout d'abord, la forme du crane peut varier beaucoup d'un individu à l'autre, parfois même au point qu'il faille changer de taille de bonnet EEG d'une personne à l'autre. Cela a pour finalité que les électrodes ne vont pas être positionnées exactement proche des mêmes aires du cerveau d'une personne à l'autre. Par ailleurs concernant ces aires du cerveau, celles-ci vont également varier d'une personne à l'autre, entraînant une variabilité encore plus importante. Généralement ces aires restent tout de même très proches, notamment les aires sensori-motrices qui possèdent des positions bien connues permettant de savoir quelles électrodes sont à prioriser étant donné la tâche d'imagerie motrice indiquée, mais elles possèdent tout de même une certaine variabilité inter-individuelle qu'il faut pouvoir gérer si on veut pouvoir réutiliser les données d'anciens sujets pour aider le système à apprendre sur le nouveau. L'avantage ici est que les données restent généralement homogènes en terme de nombre et position d'électrodes, fréquence d'échantillonnage et autres paramètres liés à la configuration de l'expérience en elle-même, rendant plus abordable l'apprentissage par transfert ou transfer learning (TL).

Il existe également un troisième type de variabilité se rapprochant beaucoup du deuxième dans sa façon de déformer les données, il s'agit de la variabilité intra-sujet ou intra-subject variability. Celle-ci prend généralement une importance élevée dans le cadre d'expériences sur plusieurs jours avec une même personne et où l'EEG est mis et retiré plusieurs fois. Cette variabilité peut avoir plusieurs causes et prendre de fait plusieurs formes. Tout d'abord, la forme la plus "mécanique" est liée à l'EEG lui-même. En effet, lorsque le bonnet est retiré puis remis le lendemain ou après un certain temps, il peut arriver qu'il soit positionné de façon légèrement différente ou bien que l'impédance de certaines électrodes ait varié. Cela a pour conséquence de déformer le signal, que ce soit en terme de rotations dans la matrice de mélange des signaux source ou bien simplement d'amplitude de l'électrode en elle-même. Par ailleurs, à ces variabilités liées à l'EEG en lui-même s'ajoute la condition du sujet qui peut varier, en effet d'un jour sur l'autre des variations dans l'état psychologique du sujet (stress, fatigue...) peuvent moduler les signaux qu'il ou elle génère. En particulier, un sujet très fatigué peut parfois générer plus d'ondes alpha. Les ondes alpha sont des ondes qui apparaissent lorsque le sujet a les yeux fermés (bien qu'elles n'apparaissent pas pour tout le monde) en étant éveillé, mais qui sont également liées à beaucoup d'autres processus cognitifs. Ces ondes peuvent avoir une assez forte amplitudes, au point où elles sont souvent visibles sur l'EEG directement. Il est d'ailleurs possible de les utiliser pour regarder rapidement si les électrodes (surtout occipitales donc) réagissent correctement en demandant simplement au sujet de fermer les yeux et en regardant l'EEG si celui-ci est affiché pour tester mais en plus les signaux générés seront d'une très mauvaise qualité en plus d'être bruités par des signaux qui se situent en plus à des fréquences comparables (notamment par rapport aux ondes mu générées en imagerie motrice) ou sur les mêmes électrodes (les ERP sont souvent situés dans le lobe occipital également).

1.5.2 Les pistes pour résoudre cette variabilité

Tous ces types de variabilité ont un impact non seulement sur le traitement des données en tant que tel mais surtout sur leur capacité à être réutilisées plus tard. En effet, on a vu un peu plus tôt comment arriver à l'espace tangent dans lequel faire la classification mais il faut bien comprendre que dans cet espace les essais peuvent être globalement répartis de façon très différentes d'une séance à l'autre en fonction de la variabilité inter et intra-sujets. Voire même dans le cadre de la variabilité inter-configurations, l'espace tangent ne possède parfois simplement même pas le même nombre de dimensions. Pour palier cela, de nombreuses méthodes d'apprentissage automatique ont émergées. Il existe majoritairement deux approches allant dans ce sens. Premièrement, il y a la mise en place de systèmes suffisamment robustes pour prendre en compte la variabilité dans le modèle-lui même en apprenant sur toutes les données sans réellement les adapter les unes aux autres. Il s'agit généralement de modèles de réseaux de neurones profonds [30], [103], [72] qui vont agir comme une boîte noire apprenant sur tout type de données et donnant en sortie un résultat en espérant très fort qu'il soit bon, car l'aspect "boîte noire" des réseaux de neurones (en particulier profonds) pose encore aujourd'hui beaucoup de problèmes pour comprendre pourquoi un modèle fonctionne correctement ou non. Cette thèse portant sur l'utilisation de méthodes découlant de la deuxième approche pour résoudre la variabilité inter- et intra-sujets en ICO, la première approche n'a pas été traitée et est si vaste qu'elle mériterait de toute façon une autre thèse complète pour produire des avancées notables.

La deuxième approche est la construction de modèles dits d'apprentissage par transfert ou transfer learning (TL) [42]. Les méthodes de TL ont émergées au cours des dernières années dans le domaine des ICO [16, 45]. Certaines vont également jusqu'à combiner des signaux provenant de systèmes différents dans le but de faire une prédiction plus robuste [29]. Le domaine est aujourd'hui en plein essor au point que des compétitions réunissant un grand nombre de chercheurs ont même vu le jour ces dernières années [88]. L'apprentissage par transfert n'est par ailleurs certainement pas un domaine réservé aux ICO, il s'agit d'un concept qui vient d'une réalisation fondamentalement simple. "Mes méthodes d'apprentissage automatique ont besoin d'informations sur la tâche à accomplir pour fonctionner correctement donc au lieu de simplement extraire de l'information de la tâche exacte à accomplir (une tâche pour un sujet spécifique selon la configuration actuelle très exactement), ce qui va demander d'obtenir beaucoup de données dans cette configuration exacte (en pratique cela correspond à une longue séance de calibration), je pourrais prendre des données analogues bien qu'un peu différentes (la même tâche mais avec d'autres sujets et des configurations qui peuvent varier) et en extraire le plus d'information possible. Il ne me restera alors qu'à extraire les informations liées à la configuration dans laquelle je me trouve pour retomber sur mes pieds." Le terme "apprentissage par transfert" s'explique donc de lui-même par le fait d'apprendre grâce à ce transfert d'information d'une configuration à l'autre. De ce domaine de recherche découle un grand nombre de problématiques liées aux méthodes ayant

pour but d'extraire l'information mais aussi liées aux données elle-mêmes sur lesquelles on se base lors de l'extraction de ces informations. La première problématique coté méthode relève d'une grande importance afin de pouvoir réduire la calibration des ICO, aujourd'hui trop longue, sans que la performance ne baisse trop. Dans cette optique, le Chapitre 2 porte sur une nouvelle méthode d'apprentissage par transfert permettant d'utiliser les données de sujets précédents pour prédire les données d'un nouveau sujet avec une faible calibration. La seconde problématique que j'ai énoncé plus haut se place plutôt du coté des données en elles-même. En effet, lors de l'utilisation de méthode de TL, les données desquelles l'information est extraite peuvent avoir un impact fort sur les performances de la méthode. Ainsi, le Chapitre 3 se concentre quant à lui sur une seconde contribution visant la mise en place d'un concept de source mathématique dite "arbitraire" permettant de créer un jeu de données avec des propriétés connues jouant le rôle de base dans l'extraction des informations à venir. Cette source arbitraire permet notamment de s'affranchir du choix des données réelles dont les caractéristiques ne sont généralement pas entièrement connues pour servir de données de base.

Il existe deux façon principales de transférer de l'information d'un domaine à l'autre et de l'adapter afin d'obtenir le meilleur résultat possible sur les nouvelles données. Considérons le système de gauche en figure 1.13. Ce système représente deux classes qu'un modèle d'apprentissage automatique a appris à séparer. Dans le cas des ICO, ce système peut représenter une séance de MI dans laquelle les points bleus représentent les essais "main gauche" et les points orange les essais "main droite". On remarque que le sujet arrive à générer des données très bien séparées et que le système arrive parfaitement à les séparer. Admettons que pour apprendre la fonction de prédiction du premier sujet, il a fallu plusieurs heures à un ordinateur ainsi que la totalité des données du premier sujet. Maintenant considérons un nouveau sujet qui veut participer à cette expérience. L'expérience étant la même, les données générées par le second sujet devraient avoir une certain forme proche de celles du premier. Cependant on remarque que si on reprend la même fonction de prédiction, on obtient de très mauvais résultats. Par ailleurs, il serait possible d'entraîner à nouveau un modèle sur les données du nouveau sujet mais on ne possède alors que quelques essais de calibration, rendant ce modèle peu robuste. On peut alors se demander comment faire. Comme expliqué plus haut, il existe deux façons de résoudre ce problème.

Tout d'abord, il est possible de chercher comment modifier un petit peu la fonction de prédiction afin que celle-ci puisse prédire correctement les essais du nouveau sujet. Il s'agit donc de trouver une transformation à appliquer à cette fonction pour qu'elle s'applique au jeu de données. Par exemple en figure 1.14 cette transformation est une simple rotation de la fonction de prédiction pour coller aux données du nouveau sujet qui avait potentiellement un bonnet EEG posé légèrement différemment.

L'autre possibilité est de modifier cette fois-ci non pas la fonction de prédiction mais les données elles-mêmes. Dans le cas de la figure 1.15, il s'agit de trouver la transformation à appliquer aux données du nouveau sujet pour que celles-ci collent avec les données du premier sujet. Dans ce cas précis, il s'agit

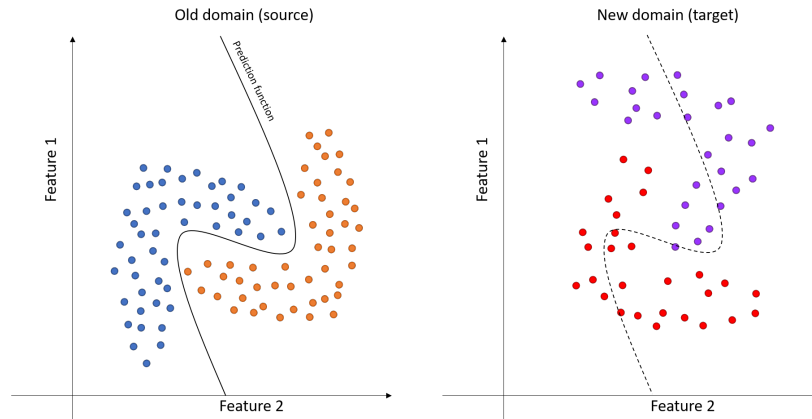


FIGURE 1.13 – À gauche un système basique présentant deux classes (bleu et orange) qu'un modèle d'apprentissage automatique a appris à séparer via une fonction de prédiction. À droite un nouveau jeu de données très proche du précédent dans sa forme mais réparti différemment.

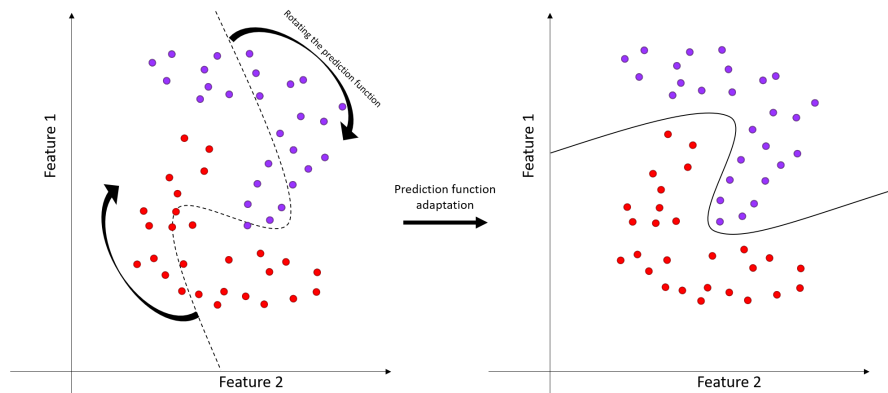


FIGURE 1.14 – L'adaptation de fonction de prédiction permet de modifier le modèle d'apprentissage automatique de telle sorte qu'il puisse prédire correctement les données du nouveau sujet.

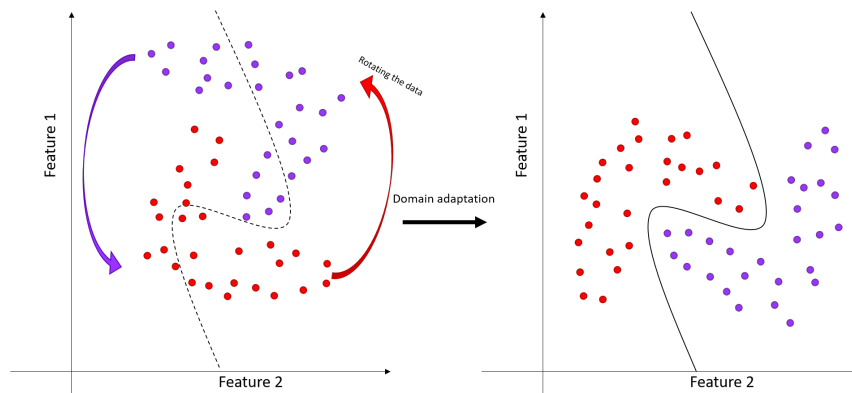


FIGURE 1.15 – Dans le cas de l’adaptation de domaine, on cherche à transformer les données elles-mêmes plutôt que la fonction de prédiction.

également d’une rotation à appliquer aux données (plus exactement la rotation inverse de celle appliquée à la fonction de prédiction dans le cas précédent), mais cela n’est pas forcément le cas, notamment lorsque les transformations à appliquer sont non-linéaires. Cette approche a plusieurs avantages comparée à la première. Cela permet tout d’abord de ne pas toucher à la fonction de prédiction et de garder simplement un modèle déjà construit et dont l’apprentissage a été finalisé. De plus, la façon la plus simple de résoudre le problème par cette approche est d’utiliser les données du premier sujet et de faire en sorte que les données du nouveau sujet leur ressemble le plus possible, ce qui est une approche généralement plus évidente que d’adapter une fonction de prédiction à des données. Cela permet par ailleurs d’entraîner le modèle sur un plus grand nombre de données puisque celui-ci peut apprendre la distribution de plusieurs ensembles transportés dans le même espace. Dans le cadre de cette thèse nous nous intéresseront exclusivement à cette dernière approche.

1.5.3 État de l’art des méthodes d’apprentissage par transfert pour les ICO riemanniennes

Il existe aujourd’hui beaucoup de méthodes d’apprentissage par transfert prenant place dans le domaine des ICO. La majorité d’entre elles se concentrent sur le l’apprentissage par transfert entre les sujets d’une même base de donnée ([86]), ([85]), ([91]) mais quelques uns d’entre eux proposent également des méthodes d’apprentissage par transfert entre différentes bases de données pour un même paradigme. ([93])

Dans les dernières années, l’apprentissage par transfert pour les ICO riemanniennes s’est beaucoup développé. Nous allons ici énoncer les premières méthodes étant aujourd’hui les plus connues. Cette thèse se place dans la continuité directe de ces méthodes.

La première méthode a été mise au point par Paolo Zanini et publiée en 2018

[101]. On a fait à plusieurs reprises mention du fait que le recentrage riemannien possède des propriétés souhaitable concernant l'apprentissage par transfert en ICO. Cette affirmation provient en particulier de cette méthode. En effet, dans leur article, Zanini et collaborateurs observent qu'en ICO riemannienne, les essais d'une session avec plusieurs classes peuvent être représentées comme des mélanges de distributions gaussiennes riemanniennes de la forme $(\bar{\mathbf{P}}, \sigma)$ dont l'estimateur de maximum de vraisemblance de $\bar{\mathbf{P}}$ (la moyenne riemannienne de cette gaussienne) coïncide avec le barycentre riemannien \mathbf{R}_C de l'équation 1.21. Il propose donc une première méthode dans le but de faire coïncider les centre de masse des gaussiennes riemanniennes de deux sessions ou deux sujets différents. Pour cela il suffit de calculer les moyennes des sujets ou sessions S (source) et T (target) que l'on notera respectivement \mathbf{R}_S et \mathbf{R}_T estimés sur les jeux de données respectifs \mathcal{C}_S et \mathcal{C}_T . Ensuite, en effectuant le recentrage tel que

$$\forall \mathbf{C} \in \mathcal{C}_S, \mathbf{C}' = \mathbf{R}_S^{-\frac{1}{2}} \mathbf{C} \mathbf{R}_S^{\frac{1}{2}} \quad (1.23)$$

pour les matrices de S et

$$\forall \mathbf{C} \in \mathcal{C}_T, \mathbf{C}' = \mathbf{R}_T^{-\frac{1}{2}} \mathbf{C} \mathbf{R}_T^{\frac{1}{2}} \quad (1.24)$$

pour les matrices de T . On notera respectivement \mathcal{C}'_S et \mathcal{C}'_T les ensemble contenant ces nouvelles matrices \mathbf{C}' dont les centres de masses coïncident en \mathbf{I}_{N_c} . Une fois ces ensembles définis, deux possibilités s'offrent à nous. Utiliser directement les nouveaux ensembles en apprenant un modèle avec ces données, ou les passer dans l'espace tangent. Via le passage dans l'espace tangent au point \mathbf{I}_{N_c} tel que, en utilisant l'équation 1.22,

$$\forall \mathbf{C}' \in \mathcal{C}'_S \cup \mathcal{C}'_T, \mathbf{M} = \mathbf{I}_{N_c}^{\frac{1}{2}} \text{Log} \left(\mathbf{I}_{N_c}^{-\frac{1}{2}} \mathbf{C}' \mathbf{I}_{N_c}^{-\frac{1}{2}} \right) \mathbf{I}_{N_c}^{\frac{1}{2}} = \text{Log}(\mathbf{C}') \quad (1.25)$$

Cette façon de définir \mathbf{M} correspond bien à la version recentrée présentée dans la partie précédente. Cela permet d'obtenir des jeux de données source et cible qui se situent dans des espaces caractéristiques semblables. Nous pouvons ensuite procéder à la vectorisation des matrices symétriques puis à la classification comme expliqué plus haut.

La seconde méthode la plus connue et de laquelle cette thèse s'inspire est l'Analyse de Procruste Riemannienne ou Riemannian Procruste Analysis (RPA) [71]. Cette méthode s'inspire des méthodes de transport optimal [97] ainsi que l'analyse de Procruste dans un espace euclidien. Cette méthode se résume en trois étape.

La première étape consiste en un recentrage. Les matrices sont recentrées en utilisant les équations 1.23 et 1.24 qui permettent de construire les ensembles \mathcal{C}'_S et \mathcal{C}'_T comme décrits. Cela permet de réduire l'écart entre le centre de masse des distributions.

Une fois cette étape terminée, il reste plusieurs variables à prendre en compte pour rapprocher les distributions statistiques de la source et de la cible. L'une

d'entre elles est la dispersion statistique des distributions. On définit ces dispersion d_S et d_T pour, respectivement, la source et la cible, par

$$d_S = \sum_{\mathbf{C} \in \mathcal{C}_S} \delta_R^2(\mathbf{R}_S, \mathbf{C}) \quad (1.26)$$

et

$$d_T = \sum_{\mathbf{C} \in \mathcal{C}_T} \delta_R^2(\mathbf{R}_T, \mathbf{C}). \quad (1.27)$$

Afin de rapprocher ou éloigner le plus possible ces dispersions, l'idée a été de faire varier les matrices de la cible sur la géodésique allant jusqu'à l'identité \mathbf{I}_{N_c} afin de se rapprocher des matrices sources. Cela se justifie par une propriété de la distance riemannienne qui est telle que $\forall \mathbf{C} \in \mathcal{S}_{++}^{N_c}, \forall s \in \mathbb{R}$,

$$\delta_R^2((\mathbf{C})^s, \mathbf{I}_{N_c}) = s^2 \delta_R^2(\mathbf{C}, \mathbf{I}_{N_c}), \quad (1.28)$$

qui montre que la dispersion autour de l'identité peut être modulé par la variation le long de la géodésique $s \rightarrow \mathbf{C}^s \mathbf{I}_{N_c}^{1-s} = \mathbf{C}^s$ qui lie \mathbf{C} à \mathbf{I}_{N_c} . Pour cela, on considère les nouvelles matrices étirées \mathbf{C}_{str} telles que

$$\forall \mathbf{C}' \in \mathcal{C}'_T, \mathbf{C}_{str} = (\mathbf{C}')^s \quad (1.29)$$

tel que s soit un réel défini par [71]

$$s^2 = \frac{d_S}{d_T}. \quad (1.30)$$

On définit par \mathcal{C}_{str} le nouvel ensemble des données cible étirées afin d'obtenir une dispersion autour de \mathbf{I}_{N_c} identique à celle des données source.

La dernière étape est l'étape de rotation de la distribution cible pour la faire coller au mieux à la distribution source. Pour cela, on remarque tout d'abord que pour une matrice \mathbf{U} orthogonale et en prenant une matrice $\mathbf{C}_{str} \in \mathcal{C}_{str}$,

$$\delta_R^2(\mathbf{U}^T \mathbf{C}_{str} \mathbf{U}, \mathbf{U}^T \mathbf{I}_{N_c} \mathbf{U}) = \delta_R^2(\mathbf{C}_{str}, \mathbf{I}_{N_c}) \quad (1.31)$$

par invariance de la distance riemannienne. Cela permet de dire qu'il est possible d'utiliser une matrice orthogonale \mathbf{U} pour effectuer une rotation sur l'ensemble de données \mathcal{C}_{str} sans en changer la distance au centre de masse \mathbf{I}_{N_c} . Cette étape consiste donc à trouver une matrice orthogonale \mathbf{U} de telle sorte à minimiser la distance globale entre les distributions source et cible. Pour cela on définit le centre de masse d'une classe k pour respectivement source et cible $\mathbf{R}_{S,k}$ et $\mathbf{R}_{T,k}$. On définit également \mathcal{C}'_S^k et \mathcal{C}'_{str}^k la restriction des ensembles \mathcal{C}'_S et \mathcal{C}'_{str} aux éléments de la classe k . Ces centres de masse sont calculés de manière analogue à ce qui a été vu jusque là, à savoir

$$\mathbf{R}_{S,k} = \arg \min_{\mathbf{C} \in \mathcal{S}_{++}^{N_c}} \left(\sum_{\mathbf{C}' \in \mathcal{C}'_S^k} \delta_R^2(\mathbf{C}, \mathbf{C}') \right) \quad (1.32)$$

et

$$\mathbf{R}_{T,k} = \arg \min_{\mathbf{C} \in \mathcal{S}_{++}^{N_c}} \left(\sum_{\mathbf{C}_{str} \in \mathcal{C}_{str}^k} \delta_R^2(\mathbf{C}, \mathbf{C}_{str}) \right). \quad (1.33)$$

La matrice \mathbf{U} est calculée de telle sorte à minimiser la somme

$$\sum_k w_k \delta_R^2(\mathbf{R}_{T,k}, \mathbf{U} \mathbf{R}_{S,k} \mathbf{U}^T) \quad (1.34)$$

avec w_k des coefficients dans $[0, 1]$ servant à équilibrer l'optimisation en fonction de la qualité de l'estimation de la moyenne de chacune des classes. La minimisation de cette somme correspond à minimiser l'éloignement entre les centres de masse des classes correspondantes dans la source et la cible. Une fois \mathbf{U} trouvé par optimisation, on peut ensuite effectuer la rotation

$$\forall \mathbf{C}_{str} \in \mathcal{C}_{str}, \mathbf{C}_{rot} = \mathbf{U}^T \mathbf{C}_{str} \mathbf{U} \quad (1.35)$$

sur les matrices étirées de \mathcal{C}_{str} , permettant ainsi la création du nouvel ensemble \mathcal{C}_{rot} des matrices dont le centre de masse de chacune des classes se sont rapprochés de ceux de la source recentrée \mathcal{C}'_S .

Une fois toutes ces étapes possibles, les ensembles \mathcal{C}'_S et \mathcal{C}_{rot} possèdent des distributions très proches l'une de l'autre, ce qui permet d'utiliser un modèle entraîné à l'aide des données de \mathcal{C}'_S pour classifier les données de \mathcal{C}_{rot} en perdant peu de précision. Il est toutefois important de noter que quelques données de \mathcal{C}_{rot} sont nécessaires afin de calculer certains éléments de cette méthode. Notamment le calcul de $\mathbf{R}_{T,k}$ demande non seulement de connaître une partie des données cible mais également de connaître leur appartenance en terme de classe. On dit que cette méthode est "supervisée" en comparaison aux méthodes "non-supervisées" qui n'ont pas besoin de connaître les labels cible pour pouvoir être utilisées.

Bien que les méthodes qui ont été décrites en détail jusque là ont été l'inspiration majeure à la production des méthodes développées durant cette thèse, un grand nombre d'autres méthodes ont permis de positionner plus clairement les recherches qui ont été faites dans ce milieu. De nombreuses méthodes de TL ont donc été développées et ce pour plusieurs types de paradigmes tels que P300 ([71]), ([57]), MI ([101]), SSVEP ([45]) et c-VEP ([100]). Par ailleurs, les auteurs de ([91]) présentent un rapport sur méthodes de TL pour chaque paradigme d'ICO, même si le nombre de méthodes est parfois faible selon du paradigme choisi. On y retrouve par exemple, les auteurs de ([95]) qui montrent la variabilité entre les ensembles de données MI et proposent une stratégie de pré-alignement en ligne pour la réduire à l'aide de la géométrie riemannienne. Dans ([92]), les auteurs décrivent une méthode qui permet d'effectuer du TL d'un casque à l'autre pour les applications ICO P300 en utilisant des données étiquetées. Ils montrent une augmentation de la précision compte tenu du même nombre de données étiquetées avec le nouveau casque. Dans la continuité de ce type de méthodes, les auteurs de ([70]) ont développé une méthode qui vise à

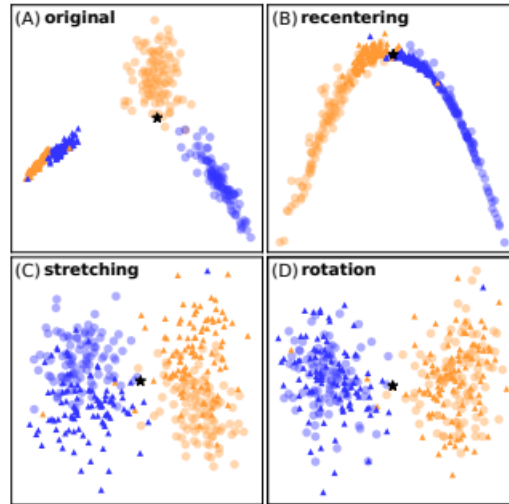


FIGURE 1.16 – Image extraite et légende traduite directement de l'article [71]. Représentation de la séquence d'opérations de la RPA appliquée à un ensemble de données simulé. Chaque point du diagramme de dispersion représente une matrice SDP et les axes des figures ont été obtenus à l'aide de cartes de diffusion. Les points remplis (degré de transparence fixé à $\alpha = 0,30$) représentent l'ensemble de données cible, tandis que les cercles représentent l'ensemble de données source. Chaque couleur représente une classe et l'étoile noire est la matrice d'identité. (A) Distribution des matrices SDP dans les ensembles de données source et cible telles qu'elles sont disponibles à l'origine et (B) après recentrage de leurs moyennes géométriques sur l'Identité. En (C) la distribution après l'opération d'étirement et (D) après la rotation.

aligner des données de tailles différentes dans la variété riemannienne des matrices de covariance. Ils parviennent à transférer des informations pertinentes qui seraient autrement impossibles à obtenir. Cette méthode est d'ailleurs la suite directe de la méthode de RPA détaillée plus tôt. Des méthodes non supervisées d'adaptation de domaine ont également vu le jour, comme dans ([77]), où les auteurs blanchissent les données avant de les transporter vers la distribution cible de façon non-supervisée. C'est dans ce domaine de recherche en rapide évolution qu'ont pu être menées les recherches de cette thèse, menant à une première contribution prenant la forme d'une méthode d'apprentissage par transfert.

Chapitre 2

Apprentissage par transfert en ICO

Table des matières

2.1	Le Tangent Space Alignment (TSA) origine et première version .	46
2.1.1	État de l'art dans le TL	46
2.1.2	Première version du TSA	47
2.1.3	Données et algorithmes	49
2.1.4	Résultats	51
2.1.5	Premières conclusions et discussion	54
2.2	Étude des améliorations apportées au TSA	55
2.2.1	Les défauts de la première version	55
2.2.2	Améliorer la matrice des produits croisés \mathbf{K}	58
2.2.3	Utiliser un assemblage d'estimations de moyennes varié et robuste	59
2.2.4	TSA améliorée par PCA (TSAxPCA)	60
2.2.5	TSA améliorée par CSP (TSAxCSP)	62
2.2.6	TSA améliorée à l'aide de valeurs propres (TSAxEIG)	66
2.2.7	TSA améliorée à l'aide de bootstraps de moyennes (TSAx-BOOTSRAP)	66
2.3	Étude du TSAxPCA élargie à plusieurs paradigme et sur un grand nombre de données	67
2.3.1	Description des données	67

2.3.2	Résultats obtenus	69
2.3.3	Discussion des résultats obtenus avec la TSAxPCA	78

2.1 Le Tangent Space Alignment (TSA) origine et première version

Après quelques travaux de recherche annexes et introductifs à l'entrée en thèse, la première grande partie de cette thèse s'est développée autour de la réalisation d'une nouvelle méthode d'apprentissage par transfert inspirée de la RPA et des autres méthodes présentées plus tôt mais agissant dans le domaine euclidien des vecteurs tangents. Celle-ci a donné lieu à une première contribution en conférence afin d'expliquer la méthode et ses inspirations :

A. Bleuzé, J. Mattout and M. Congedo, "Transfer Learning for the Riemannian Tangent Space : Applications to Brain-Computer Interfaces," 2021 International Conference on Engineering and Emerging Technologies (ICEET), Istanbul, Turkey, 2021, pp. 1-6, doi : 10.1109/ICEET53442.2021.9659607.

Toute la partie 2.1 est donc très largement inspirée de cette contribution.

2.1.1 État de l'art dans le TL

Les méthodes d'apprentissage par transfert évoluant à une vitesse importante depuis quelques années, il a semblé approprié de prendre quelques lignes pour poser le contexte de création de la méthode à la date où elle a été développée. Ainsi à ce moment là, des progrès assez voire très récents ont été réalisés dans le domaine de l'adaptation de domaine à l'aide de la géométrie riemannienne. La première proposition visait à adapter le domaine en utilisant le concept de géométrie différentielle du transport parallèle sur la variété des matrices SDP ; les points de données du sujet source et du cible sont recentrés sur l'origine de la variété, qui est la matrice identité. Il s'agit de la méthode développée par Paulo Zanini [101] et décrite précisément un peu plus tôt dans la thèse en partie 1.5.3. Il s'avère qu'un tel recentrage est équivalent à un blanchiment euclidien des données. De même, dans [96], les auteurs ont recentré les données sur un autre point commun, à savoir le point médian entre les centres de masse des données source et cible. Comme on pourrait s'y attendre, des travaux ultérieurs [71] ont montré que ces deux méthodes donnent les mêmes résultats. Les travaux en question sont d'ailleurs également ceux présentant la RPA détaillée un peu plus tôt en 1.5.3. L'article fait d'ailleurs une comparaison avec un bon nombre de méthodes proches de la RPA, permettant de comprendre l'apport de chacune des étapes de la RPA. Les méthodes décrites jusqu'à présent effectuent l'apprentissage par transfert sur la variété et les données sont ensuite projetées sur l'espace tangent à des fins de classification [25]. Les auteurs de [59] proposent un recentrage sur la variété est suivi d'une projection sur l'espace tangent, où les vecteurs tangents sont soumis à une PCA, indépendamment pour les ensembles de données

source et cible. Cette méthode présente l'avantage d'être non supervisée. Toutefois, contrairement à l'objectif même de l'apprentissage par transfert, la PCA n'aboutit pas à l'alignement des vecteurs tangents de la source et de la cible. En outre, comme le reconnaissent les auteurs, la PCA souffre d'une ambiguïté de signe, qui nécessite une correction manuelle du signe des vecteurs principaux, ce qui rend la méthode moins viable dans la pratique. Il convient de noter qu'une telle PCA a déjà été proposée dans [7], mais uniquement pour la sélection des variables. Finalement, une autre méthode dans le même esprit, bien que ne s'inscrivant pas dans un cadre riemannien, est la méthode du CORrelation ALignment (CORAL) [77]. Les auteurs ont proposé une adaptation non supervisée du domaine qui peut être interprétée dans le cadre du transport parallèle sous certaines conditions. Elle consiste à blanchir le domaine source et à le recolorer avec la covariance du domaine cible. Il s'agit d'un transport parallèle du point de base source vers l'identité puis vers le domaine cible. L'approche de cette méthode possède certaines similarités avec la méthode du TSA développée au cours de cette thèse, elle a donc été choisie comme première comparaison pour la version de base de l'algorithme du TSA.

2.1.2 Première version du TSA

Dans cette partie, nous allons détailler la mise en place de la méthode de Tangent Space Alignment (TSA) développée au cours de cette thèse. Ici nous proposons d'aligner les vecteurs tangents en utilisant la méthode de Procruste dans l'espace tangent. Sous certaines conditions, cela équivaut à aligner les vecteurs tangents source et cible en utilisant une analyse de covariance maximale ou maximum covariance analysis (MCA) [24]. La méthode d'apprentissage par transfert que nous proposons englobe les méthodes susmentionnées ; bien qu'elle puisse être conçue comme une MCA sur les vecteurs de caractéristiques, donc comme l'extension bivariée de la PCA appliquée dans [59], elle est présentée comme une analyse de Procruste. Il en résulte une version pour l'espace tangent de la RPA appliquée dans [71]. Comme notre méthode s'applique dans l'espace tangent, contrairement à la RPA, elle peut être utilisée dans toutes les circonstances, et pas seulement dans le cadre de l'apprentissage automatique riemannien. Pour la même raison, elle peut être utilisée avec n'importe quel type de données, elle se généralise bien avec n'importe quel nombre de classes et pas seulement en ICO, notre méthode est donc de la plus grande généralité. Elle ne nécessite pas non plus d'algorithme itératif puisqu'elle donne une solution déterministe. Ainsi, l'algorithme est plus rapide que la RPA et a le potentiel d'être plus précis mais la comparaison à la RPA sera faite plus précisément dans la suite de la thèse, notamment sur la version améliorée du TSA.

Comme cela a pu être expliqué plus tôt, pour l'apprentissage automatique, nous utilisons le cadre offert par la géométrie riemannienne [7], [8]. Ce cadre s'est avéré très efficace dans le domaine des ICO, à la fois en termes de précision et de robustesse, comme le montre le score gagnant obtenu dans cinq concours internationaux de classification de données ICO [26]. On se place donc toujours avec des matrices SDP ensuite traitées comme des points dans la variété rieman-

nienne de courbure non positive adoptant la métrique affine-invariante [12], [61] puis projetées sur l'espace tangent définis plus tôt dans la thèse et finalement vectorisé comme détaillé en 1.4.

Dans cette première partie, on s'est concentré sur les ICO basés sur le paradigme des ERP en P300. Comme expliqué plus tôt, dans le cadre de ce paradigme, il existe deux classes, la classe + et la classe -. La classe + correspond à la réponse aux stimulations des éléments que l'utilisateur souhaite sélectionner. La classe - correspond à la réponse à toutes les autres stimulations.

Considérons

$$\mathbf{Z}_T = [\mathbf{z}_{T,1} \cdots \mathbf{z}_{T,N_t}] \text{ et } \mathbf{Z}_S = [\mathbf{z}_{S,1} \cdots \mathbf{z}_{S,N_t}] \quad (2.1)$$

les matrices contenant respectivement dans leurs colonnes un paquet de N_t vecteurs de caractéristiques provenant de la cible et N_t provenant de la source. Il n'aurait aucun changement à avoir un nombre différent d'essais, cette notation sert simplement à simplifier la lecture et la compréhension.

Afin d'obtenir des distributions proches entre la source et la cible, un changement d'échelle est appliqué à la cible et la source afin d'égaliser la norme moyenne des vecteurs tangent. Celui-ci est obtenu en fixant la norme moyenne de chaque ensemble à 1, ce qui est facilement obtenu par les transformations

$$\tilde{\mathbf{Z}}_S = \frac{\mathbf{Z}_S}{\frac{1}{N_t} \sum_{i=1}^{N_t} \|\mathbf{z}_{S,i}\|} \quad (2.2)$$

et

$$\tilde{\mathbf{Z}}_T = \frac{\mathbf{Z}_T}{\frac{1}{N_t} \sum_{i=1}^{N_t} \|\mathbf{z}_{T,i}\|}. \quad (2.3)$$

Il est également possible, si souhaité, de ne pas toucher à la source et de simplement mettre à l'échelle la cible en la divisant par le rapport de moyenne des normes tel que

$$\tilde{\mathbf{Z}}_T = \mathbf{Z}_T \frac{\frac{1}{N_t} \sum_{i=1}^{N_t} \|\mathbf{z}_{S,i}\|}{\frac{1}{N_t} \sum_{i=1}^{N_t} \|\mathbf{z}_{T,i}\|}. \quad (2.4)$$

Considérons maintenant la concaténation des vecteurs tangents moyens de chacune des classes

$$\bar{\mathbf{Z}}_S = [\bar{\mathbf{z}}_{S,-}, \bar{\mathbf{z}}_{S,+}] \text{ et } \bar{\mathbf{Z}}_T = [\bar{\mathbf{z}}_{T,-}, \bar{\mathbf{z}}_{T,+}], \quad (2.5)$$

où $\bar{\mathbf{z}}_{T,-}$ est le vecteur tangent moyen pour la classe - et $\bar{\mathbf{z}}_{T,+}$ est le vecteur tangent moyen pour la classe + pour le sujet cible et de manière analogue pour le sujet source. Cette notation est réservée à ce paradigme en particulier mais la généralisation à d'autres paradigmes se fait de manière évidente en considérant

$$\bar{\mathbf{Z}}_S = [\bar{\mathbf{z}}_{S,1}, \dots, \bar{\mathbf{z}}_{S,N_{class}}] \text{ et } \bar{\mathbf{Z}}_T = [\bar{\mathbf{z}}_{T,1}, \dots, \bar{\mathbf{z}}_{T,N_{class}}], \quad (2.6)$$

avec N_{class} le nombre de classes du paradigme et $\bar{\mathbf{z}}_{S,i}$ et $\bar{\mathbf{z}}_{T,i}$ les vecteurs moyen de la classe i pour respectivement la source et la cible.

Afin de réaligner les vecteurs tangents de la cible sur ceux de la source, nous résolvons le problème de Procruste

$$\arg \min_{\mathbf{P} \in \mathcal{O}} \|\mathbf{P}\bar{\mathbf{Z}}_T - \bar{\mathbf{Z}}_S\|_F, \quad (2.7)$$

où \mathcal{O} est le groupe orthogonal. Pour ce faire, nous considérons le produit croisé des deux paquets, c'est-à-dire,

$$\mathbf{K} = \bar{\mathbf{Z}}_T \bar{\mathbf{Z}}_S^T \quad (2.8)$$

et calculons sa décomposition en valeurs singulières comme suit

$$\mathbf{K} = \mathbf{U}^T \Delta \mathbf{V}. \quad (2.9)$$

Il est bien connu que le problème (2.7) a pour solution $\mathbf{P} = \mathbf{V}\mathbf{U}^T$, où \mathbf{U} et \mathbf{V} sont les matrices orthogonales dans (2.9). Par conséquent, les vecteurs de caractéristiques transformés du problème de Procruste pour la cible sont obtenus par

$$\mathbf{Z}'_T = \mathbf{V}\mathbf{U}^T \mathbf{Z}_T. \quad (2.10)$$

En pratique, la totalité des matrices de rotation n'est pas gardée à chaque fois. Un certain nombre de colonne de \mathbf{U} et \mathbf{V} sont sélectionnées. Cette sélection a plusieurs propriétés intéressantes. Elle permet déjà de filtrer une partie du bruit en ne sélectionnant que les vecteurs de la rotation qui expliquent une partie significative de la covariance croisée. Elle permet également de ne pas sélectionner les vecteurs qui ne font pas de sens géométrique à cause de la déficience de rang de la matrice des produits croisés, dont nous parlerons un peu plus tard en partie 2.2. Dans cette première version, le ratio est choisi de manière à expliquer 99% de la covariance croisée.

2.1.3 Données et algorithmes

Pour évaluer et comparer notre approche, nous construisons des paires d'ensembles de données \mathbf{Z}_T et \mathbf{Z}_S (2.1) en suivant la méthode de traitement décrite dans le Chapitre 1. En utilisant les données d'entraînement, nous prenons chaque sujet d'une base de données comme cible et tous les autres comme sources successives afin de construire le produit croisé \mathbf{K} dans l'équation (2.8) et de résoudre le problème de Procruste. Considérons \mathbf{Z}_{Test} un troisième ensemble de données de vecteurs de caractéristiques du sujet correspondant au jeu de données de test. La méthode de référence est la procédure classique d'entraînement et de test.

1. *TSA* : En utilisant la méthode du TSA nous considérons $\mathbf{Z}'_T = \mathbf{V}\mathbf{U}^T \mathbf{Z}_T$ conformément à (2.10). Finalement, \mathbf{Z}_S forme l'ensemble d'apprentissage et \mathbf{Z}_T ne sera plus utilisé. $\mathbf{Z}'_{Test} = \mathbf{V}\mathbf{U}^T \mathbf{Z}_{Test}$ est utilisé comme ensemble de test. Une description schématique de la méthode d'apprentissage par transfert TSA est présentée en figure 2.1.

2. *CORAL* : Afin de permettre une comparaison équitable avec le TSA, les vecteurs de caractéristiques source et cible utilisés pour CORAL sont les mêmes données \mathbf{Z}_S et \mathbf{Z}_T , car il s’agit de vecteurs situés dans l’espace tangent euclidien. En outre, les données de test $\mathbf{Z}_{T_{est}}$ sont également conservées pour les tests. Il convient cependant de noter que la méthode CORAL n’utilise pas la labélisation des données, étant une méthode non supervisée.

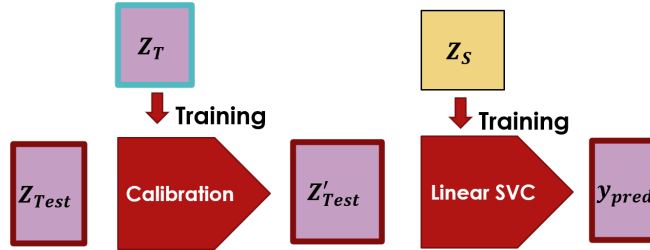


FIGURE 2.1 – Procédure du TSA : Un Support Vector Classifier linéaire est entraîné sur les données source \mathbf{Z}_S . Les vecteurs tangents d’entraînement de la cible \mathbf{Z}_T sont utilisés comme données d’alignement. Enfin, les vecteurs tangents des données de test de la cible $\mathbf{Z}_{T_{est}}$ sont alignés et utilisés comme ensemble de test. Voir le texte pour plus de détails.

Lors de cette première étude, toutes les analyses ont été effectuées à l’aide d’un code interne écrit dans le langage de programmation Julia [11]. L’estimation des matrices de covariance est réalisée à l’aide de la bibliothèque *CovarianceEstimation.jl*¹. Pour la manipulation des données dans la variété riemannienne des matrices SDP, on utilise les bibliothèques *Diagonalizations.jl*, *PosDefManifold.jl* et *PosDefManifoldML.jl* [28], toutes accessibles publiquement². Enfin, la classification a été réalisée à l’aide d’un Support Vector Classifier (SVC) importé de la bibliothèque Scikit-Learn de Python³. Cette partie est pertinente à garder en tête car il s’agit de la seule partie de la thèse qui a été faite sous Julia. Dans la suite, une migration vers Python a été effectuée, notamment dans le but de pouvoir tester les méthodes sur un très grand nombre de données grâce à certaines bibliothèques spécifiques.

Ainsi, dans cette partie, nous utilisons une seule base de données publique pour l’analyse. Elle a été enregistrée au laboratoire GIPSA, à Grenoble (France), et comprenait une expérience ICO utilisant le jeu vidéo Brain Invaders basé sur le paradigme P300 [27]. La base B1.EEG.2012-GIPSA [82] contient des enregistrements EEG de 25 sujets sains. Les signaux EEG ont été acquis au moyen du NeXus-32 (MindMedia, Herten, Allemagne), un amplificateur de niveau recherche. Le bonnet EEG était équipé de 16 électrodes humides en argent/chlorure d’argent, placées selon le système international 10-20. Les électrodes étaient placées en F7, F3, F4, F8, T7, C3, CZ, C4, T8, P7, P3, PZ, P4,

1. <https://github.com/mateuszbaran/CovarianceEstimation.jl>

2. <https://github.com/Marco-Congedo>

3. <https://github.com/scikit-learn/scikit-learn>

P8, O1 et O2. La machine NeXus-32 n'utilise pas d'électrode de référence mais une référence moyenne commune matérielle. Les données de trois sujets se sont avérées corrompues et, pour un autre sujet, le nombre d'essais disponibles était insuffisant. Par conséquent, 21 sujets ont été conservés pour l'analyse ultérieure. La tâche de classification dans ce paradigme ICO est un problème à 2 classes, avec un ratio d'essais par classe de 5 :1 (non cible :cible). Chaque sujet a effectué un étalonnage et un test.

2.1.4 Résultats

Les classes étant déséquilibrées, la précision équilibrée (ou balanced accuracy) est utilisée dans toutes les analyses. Elle est définie comme suit

$$\text{Précision équilibrée} = \frac{\text{sensibilité} + \text{spécificité}}{2}, \quad (2.11)$$

avec la sensibilité étant la probabilité qu'un essai soit classifié dans la classe A (généralement appelée classe positive) quand il y appartient et la spécificité étant la probabilité qu'un essai soit classifié dans la classe B (généralement appelée classe négative) lorsqu'il y appartient.

La figure 2.2 montre les précisions équilibrées en fonction du nombre d'essais utilisés pour l'alignement pour TSA et CORAL par rapport à une méthode classique d'entraînement puis test utilisant les données d'alignement comme données d'entraînement pour le classifieur. On utilisera de manière égale "golden standard" et "entraînement-test" pour parler de cette comparaison. L'écart-type pour l'entraînement classique est plus élevé, car la courbe représente la moyenne des 21 sujets, alors que pour les deux méthodes d'alignement, la courbe représente la moyenne des 20*21=420 paires source-cible.

Nous observons qu'avec un faible nombre d'essais utilisés pour l'alignement, la méthode CORAL donne une meilleure précision par rapport à la méthode TSA et à l'évaluation entraînement-test. Cependant les performances du TSA augmentent rapidement avec l'ajout de nouvelles données et finit par dépasser CORAL. Pour l'analyse qui suit, nous nous concentrons sur trois parties distinctes de ces courbes :

- L'alignement faible basé sur moins de 120 essais
- L'alignement moyen basé sur un nombre d'essais compris entre 120 et 240.
- L'alignement élevé basé sur plus de 240 essais.

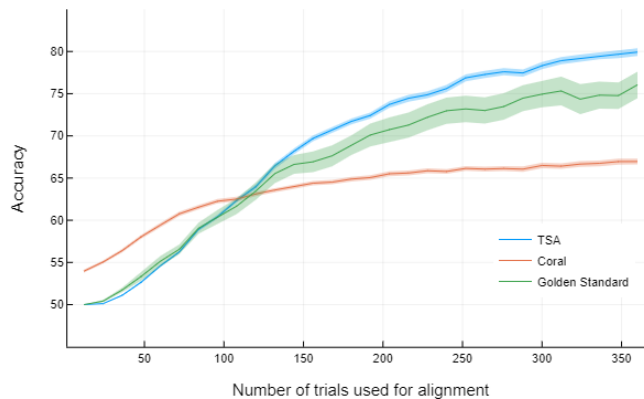


FIGURE 2.2 – Moyenne (ligne) et écart-type (zone grisée) de la précision équilibrée sur le nombre d’essais utilisés pour l’alignement pour toutes les méthodes présentées plus tôt.

En s’inspirant des méthodes de visualisation des résultats de Rodrigues et al. [71], nous utilisons des graphiques de sériation pour afficher la précision équilibrée pour toutes les paires source-cible possibles pour les méthodes TSA et CORAL. Nous évaluons les précisions équilibrées obtenues pour l’alignement faible, moyen et élevé, respectivement. Les graphiques de sériation correspondants sont présentés à la figure 2.3. Pour chaque sous-graphique, chaque ligne correspond à un sujet cible et chaque colonne correspond à un sujet source. Les lignes et les colonnes sont triées par ordre croissant de précision pour les cibles obtenues par la méthode TSA. Notons que chaque ligne comporte une valeur fictive (en noir), qui correspond au cas où le sujet source est également le sujet cible. En outre, un graphique de sériation de l’entraînement-test a également été calculé, même si la source n’a aucun impact sur la précision puisqu’elle n’est pas utilisée. L’objectif de ces tracés est uniquement de faciliter la comparaison visuelle avec TSA et CORAL.

Nous observons que pour la méthode CORAL, la précision d’un sujet donné dépend fortement à la fois de la cible et de la source. Pour la méthode TSA, nous observons qu’elle dépend principalement du sujet cible uniquement. Cela montre que lorsque l’on dispose d’un nombre suffisant d’essais d’alignement, il est possible d’obtenir une faible dépendance à l’égard du choix des données sources. Cela signifie que nous pouvons aligner les données entre les sujets sans nous soucier de leur "compatibilité". Il s’agit d’une caractéristique intéressante pour un algorithme d’apprentissage par transfert pour les ICO, car cela signifie que si les sujets sont alignés avec suffisamment de données, leur espace de caractéristiques sera aligné, quelle que soit la distance qui les sépare à l’origine. Les utilisations possibles d’une telle caractéristique seront examinées plus en détail un peu plus tard en discutant de ces premiers résultats.

Enfin, nous avons effectué une analyse statistique pour comparer la précision globale (moyenne sur le nombre d’essais utilisés pour l’alignement) obtenue par

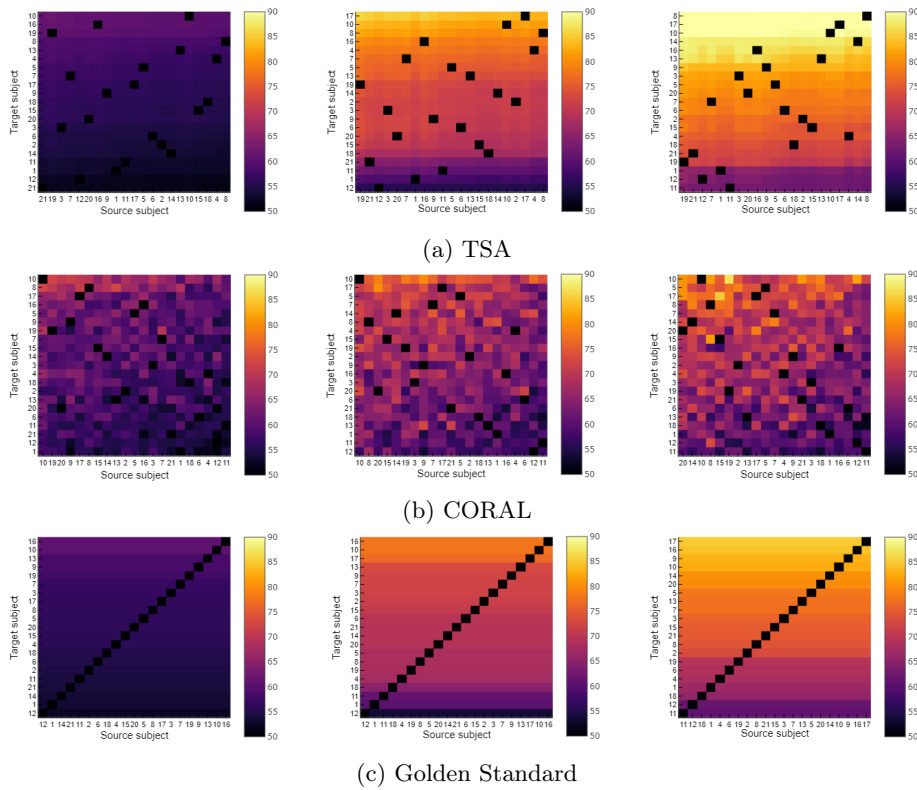


FIGURE 2.3 – Graphique de sériation obtenu avec TSA (première ligne), CORAL (deuxième ligne) et le golden standard (troisième ligne) pour, de gauche à droite de la colonne, une situation d’alignement faible, moyen et élevé, respectivement.

chacun des deux pipelines par rapport au golden standard. Pour chaque sujet source, nous comparons la précision entre les trois méthodes en appliquant des t-tests de Student appariés aux données présentées dans la Fig. 2.2. Pour tous les sujets source, TSA est supérieur au golden standard et à CORAL. La p-value est inférieure à 0,001 pour 17 des 21 sujets comparant TSA à CORAL et pour 7 des 21 sujets comparant TSA au golden standard.

2.1.5 Premières conclusions et discussion

Cette première contribution à la communauté scientifique est une nouvelle méthode d'apprentissage par transfert permettant de réduire la variabilité inter- et intra-sujets en ICO. Elle permet un transfert très efficace des informations d'un sujet à l'autre, avec des résultats supérieurs à ceux d'une méthode bien établie, à condition d'utiliser un nombre suffisant d'essais pour l'alignement. Il s'agit également de la première tentative, à notre connaissance, de réaliser l'apprentissage par transfert dans l'espace tangent riemannien pour le traitement des signaux. La méthode TSA présentée ici démontre un bon potentiel pour l'apprentissage par transfert dans le cas des données ICO basées sur le P300, où le décalage entre les distributions de données des sujets est assez important en général. Dans cette étude, la méthode est utilisée pour aligner les vecteurs tangents vectorisés de matrices SDP vivant dans une variété riemannienne, mais elle n'est pas limitée à ce cas et pourrait être appliquée à tout type de vecteurs de caractéristiques se trouvant dans un espace euclidien.

L'utilisation de l'espace tangent riemannien pour la classification en ICO suscite actuellement un intérêt croissant, car il permet une grande précision sans ajouter trop de complexité. La méthode TSA résulte de la volonté de faire avancer la recherche dans cette direction. En outre, cette méthode ne dépend pas de nombreux paramètres pour fonctionner correctement. Seul le ratio de covariance expliquée considéré avant la rotation (choisi ici à 99%) est un paramètre variable. Cependant, son seul usage est d'annuler le bruit dans le processus de rotation. Si l'utilisateur estime que son utilisation n'est pas nécessaire, il peut la fixer à 100% et l'ignorer.

Nous pouvons résumer cette première partie par :

- A faible alignement, la méthode CORAL obtient généralement de meilleurs résultats que les méthodes TSA et golden standard.
- En cas d'alignement moyen, les méthodes TSA et golden standard surpassent CORAL, la précision de TSA augmentant plus rapidement que celle de golden standard.
- Lorsque l'alignement est élevé, les trois méthodes se stabilisent. TSA atteint une précision de 80%, tandis que golden standard plafonne à 75% et CORAL à 65%.

En conclusion, TSA présente une amélioration systématique de la précision pour tous les sujets, quelle que soit leur précision golden standard, dès qu'environ 120 essais sont disponibles pour l'alignement. Nous observons qu'avec la méthode TSA, la précision obtenue pour une cible donnée dépend peu des données source. Ce type de caractéristique rend possible la création d'une base de

données composée de l’alignement de plusieurs sujets. Une telle base de données a le potentiel d’aider le domaine en tirant parti de l’utilisation de réseaux neuronaux profonds qui dépendent fortement de la quantité de données afin de créer un algorithme robuste avec une adaptation fiable entre les sessions et entre les sujets.

Il convient de souligner que la méthode n’a été testée que sur un seul ensemble de données jusque là. Dans la suite de cette thèse, d’autres bases de données seront utilisées afin de garantir l’efficacité de la méthode dans d’autres paradigmes d’ICO. D’autres études pourraient également porter sur la précision de cette méthode avec plus de deux classes mais malheureusement cela n’a pas pu être fait au cours de cette thèse par manque de temps, même si quelques tests visuels ont été effectués donnant des alignements prometteurs allant dans ce sens. Enfin, la recherche sur l’apprentissage par transfert entre bases de données semble également prometteuse puisque la méthode TSA peut aligner des bases de données avec un nombre différent de canaux. Nous verrons d’ailleurs plus tard dans la thèse qu’en effet le TSA montre de très bons résultats allant dans ce sens. Cela est dû au fait qu’elle utilise la décomposition en valeurs singulières (SVD) afin d’aligner les données, et que la SVD est possible sur les matrices rectangulaires de produits croisés.

2.2 Étude des améliorations apportées au TSA

Cette seconde sous-partie fait suite directe à la première version du TSA présentée plus tôt. Dans cette partie on s’est intéressé à rechercher des moyens d’améliorer la méthode. Les améliorations proposées sont très largement inspirées du travail publié dans l’article :

Bleuzé A, Mattout J and Congedo M (2022) Tangent space alignment : Transfer learning for Brain-Computer Interface. *Front. Hum. Neurosci.* 16 :1049985. doi : 10.3389/fnhum.2022.1049985.

Toutefois bon nombre des problèmes expliqués en détail ici ainsi que quelques idées de solutions se rapportent à du travail d’arrière plan qui, bien que suffisamment pertinent pour figurer dans cette thèse, n’a pas été publié dans des journaux académiques ou conférences et ne le sera probablement jamais.

2.2.1 Les défauts de la première version

Dans la première version de l’algorithme du TSA, la matrice des produits croisés \mathbf{K} est calculée à l’aide des matrices $\bar{\mathbf{Z}}_S$ et $\bar{\mathbf{Z}}_T$, or ces matrices sont simplement composées de la concaténation de N_{class} vecteurs moyens colonne. Dans les cas qui ont été étudiés de manière approfondie au cours de cette thèse cela correspond à deux vecteurs seulement. Ce nombre est très faible comparé à la taille des vecteurs et donc à la dimension de l’espace étudié dans lequel on souhaite calculer une rotation. Ainsi le rang de la matrice \mathbf{K} obtenue est obligatoirement faible comme schématisé en figure 2.4 et la rotation qui en

découle par SVD ne peut pas agir de manière appropriée dans l'espace des caractéristiques.

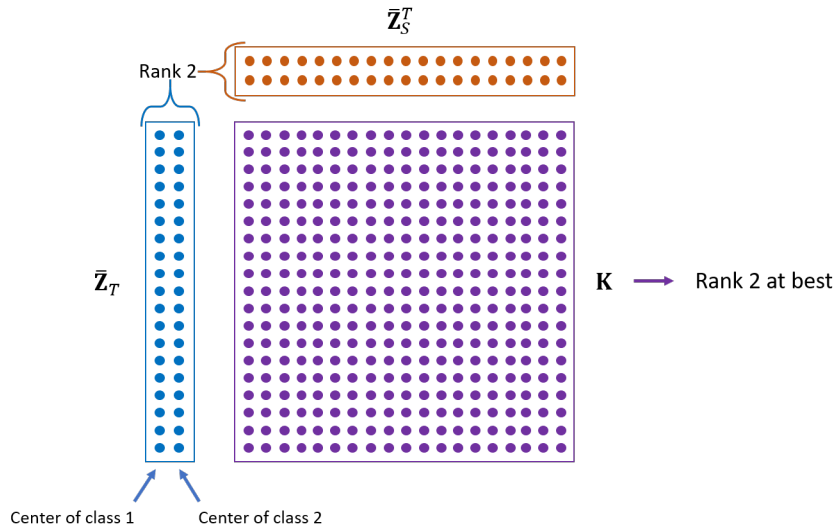


FIGURE 2.4 – Schématisation du calcul de la matrice des produits croisés \mathbf{K}

Une autre défaut de cette version est que le fait de prendre seulement le point moyen de chacune des classes pose un problème d'estimation dans la rotation. En effet, une fois le recentrage et la mise à l'échelle effectués pour rapprocher les distributions, comme on peut le voir sur des données générées de manière aléatoire en figure 2.5, il reste la rotation à effectuer.

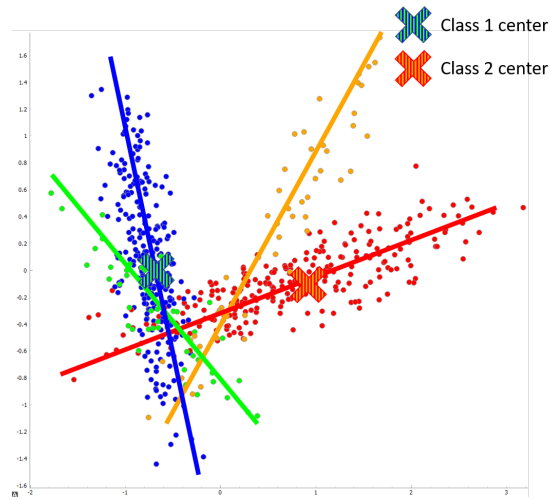


FIGURE 2.5 – Représentation de la source (rouge et bleu) et de la cible (orange et vert) après recentrage et mise à l'échelle. Les données sont simulées pour rendre les explications plus évidentes visuellement.

Celle-ci paraît assez évidente à première vue, il suffit d'effectuer une rotation de l'ensemble cible vers l'ensemble source comme schématisé en figure 2.6. Cependant, dans les faits, il ne s'agit pas simplement d'une rotation 2D. Les figures montrées ici sont des projections 2D des données (parfaitement exécutée car les données sont générées pour cette démonstration) qui elles sont en 3D.

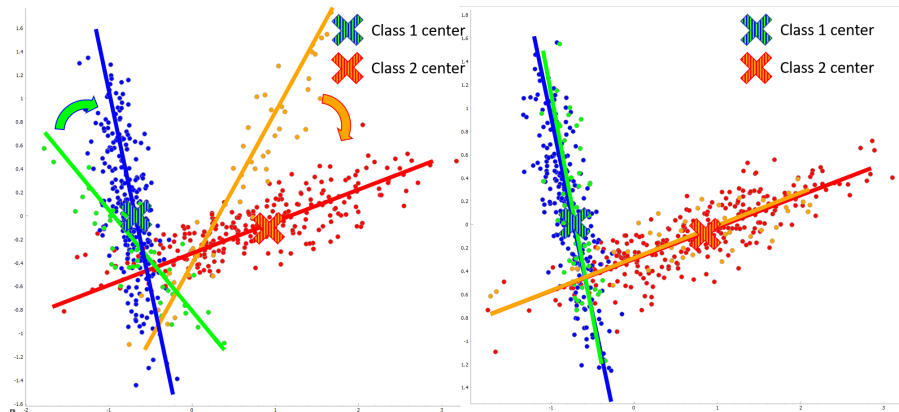


FIGURE 2.6 – Exemple de rotation des données permettant de faire coïncider correctement les distributions source et cible.

De fait la rotation est en 3D et bien qu'on obtienne souvent le meilleur alignement comme on peut le voir en figure 2.6, il arrive très régulièrement que la rotation qui découle de l'algorithme soit mauvaise et donne de piètres résultats

comme on peut le voir en figure 2.7. Cette rotation fait pourtant totalement sens pour ce qui est demandé à l’algorithme. En effet, son seul but est de faire coïncider les centre des classes 1 et 2 pour la source et la cible et il le fait très bien. Il est pourtant évident qu’avec une représentation des données un peu plus complète que la seule moyenne des classes, il serait possible d’éviter ce type de problème sans trop de difficulté.

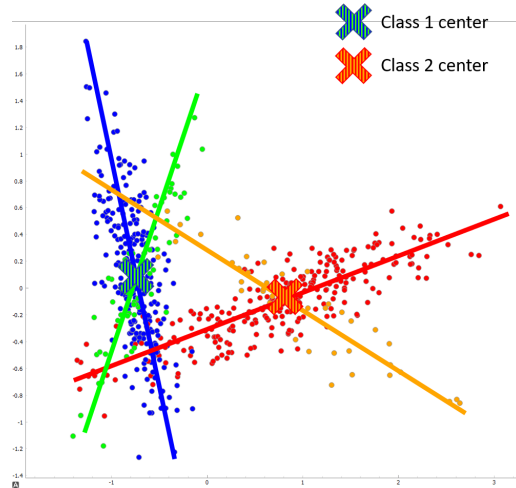


FIGURE 2.7 – Mauvaise rotation classique qui peut arriver dans la première version de l’algorithme.

De la même façon, il arrive également régulièrement que lors de la sélection de vecteurs liés au ratio de la cross-covariance expliquée (qui a été posé à 99% dans la première version du modèle), on obtienne un nombre très faible de vecteurs sélectionnés. Découle de cela une forte projection des données plutôt qu’une rotation, tel qu’on peut l’obtenir dans la figure 2.8. Il reste cependant à noter que cette projection simpliste des données donne tout de même de très bons résultats en terme de précision pure sur un classifieur entraîné sur les données sources. Mais cela n’est pas étonnant car on remarque visuellement qu’un modèle capable de classifier correctement la source (en bleu et rouge) n’aura aucun problème avec la classification des données cible projetées (en vert et orange), car elles sont placées correctement. Il semble tout de même évident qu’un tel alignement n’est pas souhaitable pour d’autres utilisations, notamment si on souhaite aligner plusieurs sujets voire bases de données ensemble afin de créer une source globale plus robuste. Il est donc important de tenir compte de ce problème dans les améliorations de la méthode.

2.2.2 Améliorer la matrice des produits croisés K

Comme expliqué un peu plus tôt, une première façon de tenir compte de certains défauts de la méthode a été d’utiliser le ratio de cross-covariance comme

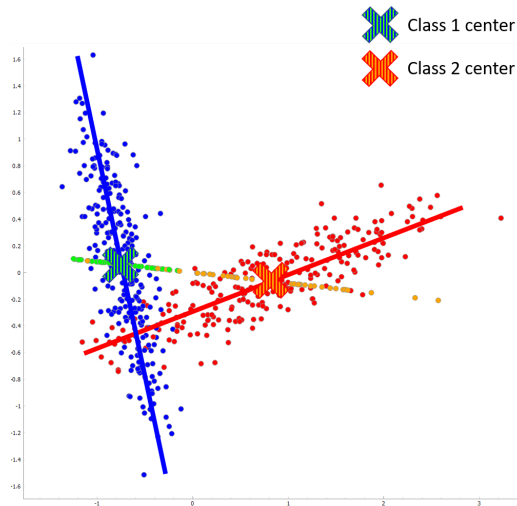


FIGURE 2.8 – Projection des données trop forte donnant une représentation des données cible alignées trop simpliste.

expliqué plus haut afin de ne garder que les vecteurs d'intérêt de la rotation permettant une projection dans un espace caractéristique de dimension plus faible plutôt qu'une rotation complète mais moins pertinente, voire contre-productive. Dans la première version du TSA, nous avons choisi de garder 99% de la variance expliquée, ce qui va donc irrémédiablement amener à une projection sur au plus 2 dimensions. Il est alors évident qu'augmenter le rang de la matrice pourrait avoir un impact bénéfique sur les résultats finaux de la méthode. La déficience de rang de la matrice de cross-covariance n'est pas irrémédiable. Son estimation pourrait être améliorée de plusieurs manières. Dans cette section, on présentera certaines des méthodes que nous avons développées afin de palier cette déficience de rang en nous concentrant majoritairement sur une qui a été publiée dans [15].

2.2.3 Utiliser un assemblage d'estimations de moyennes varié et robuste

Premièrement, tant que le TL supervisé est possible, puisque nous nous appuyons sur le calcul de la moyenne des vecteurs tangents, il est possible d'utiliser des estimateurs moyens robustes. Par exemple, nous pouvons prendre en compte les moyennes tronquées, la médiane ou les moyennes de puissance pour estimer des points d'ancrage adéquats. Nous pouvons également empiler plusieurs de ces estimateurs moyens pour obtenir des matrices plus grandes $\bar{\mathbf{Z}}_S$ et $\bar{\mathbf{Z}}_T$, qui peuvent fournir des informations plus robustes sur la tendance centrale réelle des données. Ainsi, considérons de nouveaux ensembles de moyennes équivalentes à celles calculées en équation 2.6 :

$$\bar{\bar{\mathbf{Z}}}_S = [\bar{\bar{\mathbf{z}}}_{S,1}, \dots, \bar{\bar{\mathbf{z}}}_{S,N_{class}}] \text{ et } \bar{\bar{\mathbf{Z}}}_T = [\bar{\bar{\mathbf{z}}}_{T,1}, \dots, \bar{\bar{\mathbf{z}}}_{T,N_{class}}],$$

mais cette fois-ci calculés à l'aide d'estimateurs robustes ou tout du moins différents de l'estimation précédente. Il est alors possible de faire plusieurs choses avec ces nouveaux ensembles. Soit ils peuvent remplacer les précédents ensembles $\bar{\mathbf{Z}}_S$ et $\bar{\mathbf{Z}}_T$ si on considère que la moyenne est simplement plus robuste, soit on estime qu'ils peuvent ajouter de l'information à celle apportée par cette première estimation. Auquel cas il est possible de simplement les concaténer et d'obtenir une description plus robuste des espaces de caractéristiques source et cible. On remplace alors les premiers ensembles $\bar{\mathbf{Z}}_S$ et $\bar{\mathbf{Z}}_T$ par la concaténation des deux ensembles selon

$$\bar{\mathbf{Z}}_S \leftarrow [\bar{\mathbf{Z}}_S, \bar{\bar{\mathbf{Z}}}_S] \text{ et } \bar{\mathbf{Z}}_T \leftarrow [\bar{\mathbf{Z}}_T, \bar{\bar{\mathbf{Z}}}_T]. \quad (2.12)$$

Il est tout à fait possible de répéter cette opérations plusieurs fois en ajoutant à chaque fois des estimateurs de la moyenne différents lorsqu'on estime qu'ils apportent de l'information. Dans les faits, ceux-ci apportent rarement grand chose de mesurable en terme de d'amélioration de la précision finale du classifieur mais ils peuvent apporter dans le cas de données très bruitées et avec des artefacts pouvant rendre l'estimation de la moyenne difficile, surtout dans le cas d'un nombre de points faible.

2.2.4 TSA améliorée par PCA (TSAxPCA)

Deuxièmement, dans une méthode un peu plus subtile et plus complexe nous avons choisi de découper les nuages de points de chacune des classes en plusieurs sous-ensembles décrivant la forme globale de l'ensemble de données plus précisément. Ensuite, il est possible de calculer des moyennes distinctes dans chacun de ces groupes et d'obtenir des ensembles de vecteurs tangents moyens $\bar{\mathbf{Z}}_S$ et $\bar{\mathbf{Z}}_T$ de plus grande taille et donnant des informations supplémentaires sur l'espace de caractéristiques.

Pour ce faire, la première idée a été d'utiliser une PCA sur chaque classe indépendamment pour créer des groupes de points repartis en fonction de leur position selon la première dimension de cette PCA comme le montre la figure 2.10. Il est d'ailleurs totalement possible de faire un premier découpage selon la première dimension de la PCA et ensuite d'autres selon les autres dimensions de la PCA pour rendre cette description des données plus précise encore. Cependant lors de ces recherches, nous avons pu remarquer que ne sélectionner que la première dimension était généralement le meilleur choix, bien que parfois ajouter la seconde peut également avoir un effet bénéfique. Mathématiquement, considérons les ensembles de vecteurs tangents \mathbf{Z}_S et \mathbf{Z}_T . Répartis en $\mathbf{Z}_{S,i}$ et $\mathbf{Z}_{T,i}$ selon leur classe respective i allant de 1 à N_{class} . Une PCA est calculée sur chacun des nuages de points $\mathbf{Z}_{S,i}$, donnant les vecteurs propres $\mathbf{e}_{i,j}$ avec j représentant la dimension, allant donc de 1 à $\frac{N_c(N_c+1)}{2}$. Sans perdre de généralité, on considère que les vecteurs sont rangés par ordre décroissant de valeur propre. On considère alors la projection

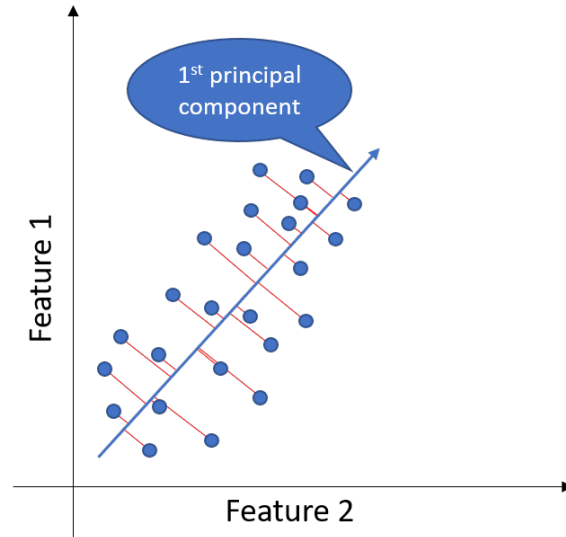


FIGURE 2.9 – Schématisation de la projection des vecteurs de $\mathbf{Z}_{S,i}$ selon la dimension principale $\mathbf{e}_{i,1}$.

$$\mathbf{p}_{i,j} = \mathbf{Z}_{S,i}^T \cdot \mathbf{e}_{i,j}. \quad (2.13)$$

Ce vecteur correspond à la projection de chacun des points de $\mathbf{Z}_{S,i}$ sur le vecteur propre $\mathbf{e}_{i,j}$. Ainsi à chaque vecteur \mathbf{z} de $\mathbf{Z}_{S,i}$ correspond une valeur $p_{\mathbf{z},j}$ étant sa position selon dimension principale comme montré en figure 2.9.

Les valeurs sont ensuite triées et les vecteurs sont regroupés en plusieurs groupes étant donné leur projection selon la dimension principale. Le point central de ces groupes est ensuite calculé et utilisé comme point d'ancrage en plus du centre global de la classe venant de la méthode TSA de base, comme on peut le voir en jaune sur la figure 2.10. Afin d'obtenir le regroupement pour l'ensemble de données source et cible, nous considérons pour chaque classe une PCA formée sur la source et utilisée comme telle sur l'ensemble de données source et cible. Cette façon d'utiliser la PCA permet de ne pas avoir à se soucier de l'ambiguïté de signe des vecteurs propres lorsqu'on considère des rotations à effectuer suffisamment faibles. Il a d'ailleurs été vérifié visuellement un grand nombre de fois au cours de cette thèse que la rotation allant de la cible à la source était systématiquement suffisamment faible pour garder la même PCA entre les deux. Cependant, garder la PCA de la source pour la cible possède d'autres problèmes, notamment dans le cadre de l'apprentissage par transfert inter-configurations. Mais tout cela sera expliqué en détails un peu plus tard dans la discussion.

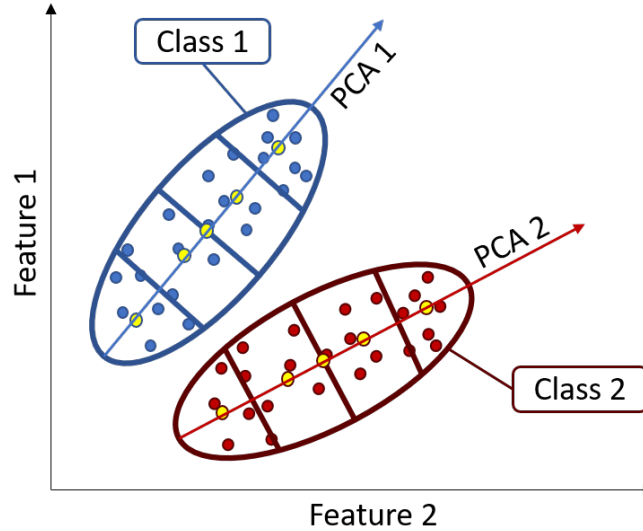


FIGURE 2.10 – Schéma d'un ensemble de données à deux classes utilisant la première dimension d'une PCA pour chaque classe.

2.2.5 TSA améliorée par CSP (TSAxCSP)

La PCA n'a pas été le seul outil de filtrage spatial que l'on a pu utiliser pour tenter d'améliorer le produit croisé \mathbf{K} . On a également porté un regard intéressé au CSP afin de créer des points permettant d'ajouter de l'information au produit croisé \mathbf{K} . Comme nous avons pu l'expliquer plus tôt en partie 1.3.2, le but du CSP est de donner des filtres spatiaux permettant la séparation de deux classes via la maximisation du rapport de variance r entre les classes, par exemple 1 et 2

$$r = \frac{V_{i,1}}{V_{i,2}}.$$

Dans ce cadre, en utilisant les outils de décomposition en valeur propre de matrices de covariance, il est possible de reprendre le concept du CSP pour isoler des filtres spatiaux permettant la séparation maximale des classes. Nous allons reprendre une version simplifiée du CSP comparée à celle expliquée en introduction. La récupération des filtres spatiaux du CSP se fait de la manière suivante. Considérons, la matrice $\mathbf{Z}_{S,i}$ et $\mathbf{Z}_{S,j}$ introduites plus haut, correspondant aux points des classes i et j respectivement. On s'intéresse à la relation entre les variances spatiales de chacune de ces classes. Pour cela il semble cohérent qu'il va y avoir besoin de la matrice de covariance spatiales des classes i et j que l'on nommera de manière originale \mathbf{R}_i et \mathbf{R}_j et qui sont telles que

$$\mathbf{R}_i = \frac{1}{N_t} \mathbf{Z}_{S,i} \cdot \mathbf{Z}_{S,i}^T$$

et

$$\mathbf{R}_j = \frac{1}{N_i} \mathbf{Z}_{S,j} \cdot \mathbf{Z}_{S,j}^T.$$

Ces matrices sont de taille $(\frac{N_c(N_c+1)}{2}, \frac{N_c(N_c+1)}{2})$ comme on pourrait s'en douter de la part de matrices de covariances spatiale vectorisées⁴. Nous avons désormais la covariance spatiale de nos essais pour chacune des classes. Le but d'un CSP est de trouver le filtre spatial maximisant le rapport de covariance entre les deux classes i et j , pour cela on résout le problème aux valeurs propres généralisé

$$\mathbf{R}_i \mathbf{v} = \lambda \mathbf{R}_j \mathbf{v} \quad (2.14)$$

avec λ une valeur propre généralisée de \mathbf{R}_i et \mathbf{R}_j et \mathbf{v} le vecteur propre généralisé associé à la valeur propre λ .

La solution la plus simple au problème arrive si \mathbf{R}_j est inversible. Ce qui est notre cas car une matrice de covariance est SDP donc inversible et le cas de non-inversibilité qui pourrait arriver avec le fait de traiter des données réelles est réglé avec une bonne régularisation. Ainsi, on peut récrire le problème sous la forme

$$\mathbf{R}_j^{-1} \mathbf{R}_i \mathbf{v} = \lambda \mathbf{v} \quad (2.15)$$

et directement chercher à obtenir les valeurs propres de $\mathbf{R}_j^{-1} \mathbf{R}_i$. On obtient alors les matrices \mathbf{P} et \mathbf{D} telles que

$$\mathbf{R}_j^{-1} \mathbf{R}_i = \mathbf{P}_{ij} \mathbf{D}_{ij} \mathbf{P}_{ij}^{-1} \quad (2.16)$$

et définies classiquement avec \mathbf{P}_{ij} étant la matrice possédant les vecteurs propres dans ses colonnes et \mathbf{D}_{ij} possédant les valeurs propres associées sur la diagonale et des 0 sinon. Un CSP classique se termine généralement en choisissant les vecteurs de \mathbf{P}_{ij} et \mathbf{P}_{ji} dont les valeurs propres associées sont les plus élevées, permettant de bien séparer la classe i de la classe j et inversement. Ces vecteurs propres représentent les patterns spatiaux permettant cette séparation. C'est à dire en simplifiant la figure 1.6, les vecteurs donnant les direction qu'on peut retrouver dans la figure 2.11.

Lorsqu'on les regarde de près, ces vecteurs de direction sont très intéressants pour décrire l'espace des caractéristiques que l'on souhaite classifier. Déjà car le simple fait de représenter la variance des points de l'espace en font de bons descripteurs de l'espace et ensuite car ils sont créés spécifiquement pour optimiser la séparation des classes i et j afin de faciliter leur classification. Cependant les

4. On remarquera d'ailleurs que l'équation de construction de la matrice de covariance spatiale est très proche quoiqu'un peu différente de celles qui ont pu être calculées en partie 1.4, notamment car on moyenne ici sur les essais et plus les échantillons temporels. Il est aussi bon de noter qu'on est en train de faire les matrices de covariance spatiales de matrices de covariance spatiales vectorisées qui sont donc traitées comme des points classiques d'un espace euclidien, d'où le fait qu'on obtienne des matrices de covariance spatiale de taille $(\frac{N_c(N_c+1)}{2}, \frac{N_c(N_c+1)}{2})$ et non pas (N_c, N_c) comme précédemment.

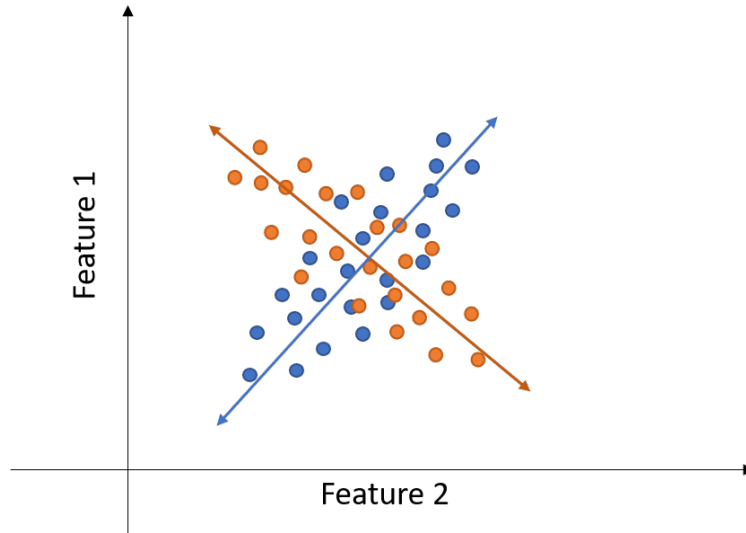


FIGURE 2.11 – Simplification de la schématisation du CSP.

vecteurs en tant que tel ne sont que des directions et les aligner à l'aide du TSA ayant plutôt pour but d'aligner des points d'ancrage plutôt que des directions semblerait fondamentalement faux. Ainsi, afin de créer des points en utilisant ces directions, il suffit de considérer nos moyennes $\bar{\mathbf{z}}_{S,i}$ pour une classe i donnée et d'y ajouter (ou soustraire) les vecteurs de \mathbf{P}_{ij} pour créer de nouveaux points d'ancrage. Cela va donner une représentation proche du schéma de la figure 2.12. Ce sont ensuite ces points qui vont être alignés de la cible vers la source. Dans le cas de la cible, les calculs sont exactement les mêmes. Le seul défaut peut venir d'une ambiguïté de signe lié à nos directions découlant du CSP mais cela se règle de la même manière que ce qui sera discuté pour la PCA plus tard lors de l'adaptation nécessaire à l'étude de bases de taille de données différentes avec cette amélioration.

Ainsi considérons la $k^{\text{ième}}$ composantes du CSP source et cible respectivement $\mathbf{e}_{s,k}$ et $\mathbf{e}_{t,k}$. Dans le cas général, prenant en compte la potentielle hétérogénéité des données, la manière la plus simple de résoudre au moins partiellement l'ambiguïté de signe, est de s'intéresser à la direction relative entre les composantes et un vecteur analogue pour les deux. Ici le vecteur qui a été choisi est simplement un vecteur de uns. Étant donné l'orientation des données que l'on retrouve usuellement lors de la visualisation, ce vecteur semble approprié, il est également très simple à s'imaginer pour raisonner géométriquement. On calcule donc le signe du produit scalaire entre la composante et le vecteur de uns que nous appellerons \mathbf{j} .

$$\tilde{\mathbf{e}}_{s,k} = \text{sign}(\mathbf{e}_{s,k} \cdot \mathbf{j}) \mathbf{e}_{s,k}$$

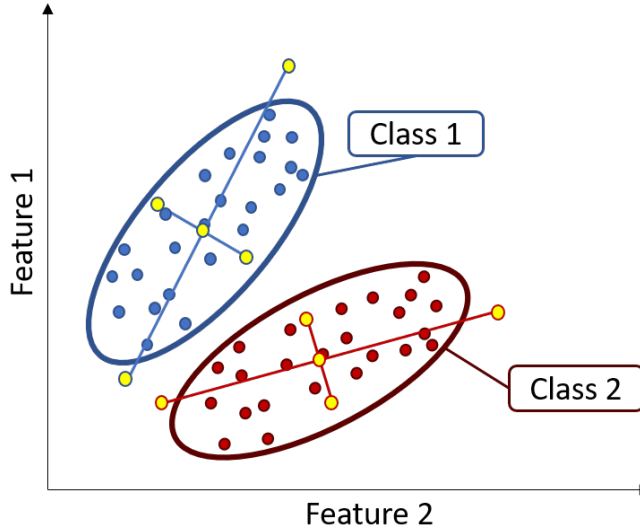


FIGURE 2.12 – Schématisation des points d'ancrage obtenus avec la méthode TSAxCSP.

et

$$\tilde{\mathbf{e}}_{t,k} = \text{sign}(\mathbf{e}_{t,k} \cdot \mathbf{j}) \mathbf{e}_{t,k}.$$

Cette façon d'estimer la direction du vecteur permet de résoudre rapidement la très large majorité des ambiguïtés dans l'orientation du vecteur extrait des deux CSP. Cette résolution repose cependant sur beaucoup d'a priori sur la relation entre les données source et cible. Notamment le fait que les deux sont orientées relativement de la même manière par rapport au vecteur de uns. Dans les faits, étant donné les pré-traitements et les premiers réalignements (notamment le recentrage) déjà effectués, les distributions de données sont déjà très proches et généralement orientées de façon équivalente, ce qui fait que cette résolution donne de très bons résultats. Cependant, dans certains cas particuliers il est tout à fait possible que des vecteurs avec une orientation proche se retrouvent avec une direction dans un sens opposé comme dans l'exemple présenté en figure 2.13. Ainsi, cette proposition simpliste bien qu'efficace ne sera par la suite utilisée que dans le cas d'alignement entre bases de données de tailles différentes. En effet, le vecteur \mathbf{j} utilisé pour la source et pour la cible n'a pas nécessairement la même dimension, le but étant de se poser une base commune d'orientation des données, même si celles-ci devaient avoir une dimension différente.

Une autre méthode, inspirée de ([18]), consiste à calculer le signe de $\mathbf{e}_{s,k}^T \mathbf{e}_{t,k}$ et à modifier le signe de $\mathbf{e}_{s,k}$ ou de $\mathbf{e}_{t,k}$ pour que ce signe soit positif, ce qui permet d'obtenir deux vecteurs allant globalement dans la même direction. Encore une fois, cette façon de résoudre l'ambiguïté de signe impose un angle relativement faible entre les deux distributions ($< 90^\circ$). Toutefois, dans toutes les

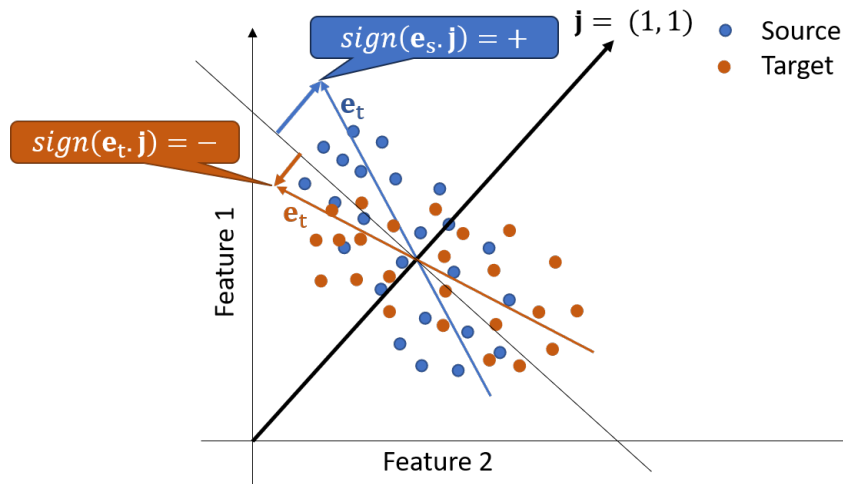


FIGURE 2.13 – Cas pratique dans lequel les composantes source et cible sont proches mais donnent un signe de produit scalaire différent par rapport à \mathbf{j} .

données que nous avons pu observer jusque là cela a toujours été le cas car la nature même des données ainsi que les pré-traitements et alignements qui leur sont appliqués auparavant forcent une certaine homogénéité. Une problème persiste cependant. Cette méthode exige que $\mathbf{e}_{s,k}$ et $\mathbf{e}_{t,k}$ aient la même dimension et n'est donc pas adaptée à l'alignement de bases de données avec des nombres de canaux différents, à moins qu'un filtrage spatial soit appliqué au préalable.

2.2.6 TSA améliorée à l'aide de valeurs propres (TSAxEIG)

La méthode du TSAxEIG peut être présentée comme le mélange entre les méthodes TSAxPCA et TSAxCSP, bien qu'elle soit antérieure à cette dernière en terme de temporalité. En effet, dans cette méthode on s'intéresse aux valeurs propres des matrices de covariance spatiale des points de chacune des classes pour la source et la cible, cependant cette fois-ci une PCA est calculée à la fois pour la source et pour la cible et au lieu d'utiliser les vecteurs principaux extraits pour créer des groupes dans les nuages de points, ces vecteurs sont directement utilisés comme des descripteurs de la direction principale de chacune des classes source et cible et sont traités exactement de la même manière que dans la TSAxCSP présentée plus tôt et avec le même objectif, simplement effectué de manière différente.

2.2.7 TSA améliorée à l'aide de bootstraps de moyennes (TSAxBOOTSRAP)

Une autre façon d'augmenter le rang de \mathbf{K} est de ne plus considérer simplement la moyenne globale de chacune des classes mais plutôt des bootstraps de

moyennes de points. Ainsi un nombre fixé N_b de bootstraps contenant un nombre fixé d’essais tirés aléatoirement est calculé, permettant ainsi de multiplier par $N_b + 1$ (en gardant la moyenne globale) la taille de \mathbf{Z}_S et \mathbf{Z}_T . Cette méthode a le bon goût d’améliorer très facilement la déficience de rang de \mathbf{K} et de ne plus donner de résultats tels que ceux présentés en figure 2.8 mais au prix de nouveaux problèmes. En effet, les bootstraps étant tirés de manière aléatoire, il n’y a aucune raison que les bootstraps correspondants de \mathbf{Z}_S et \mathbf{Z}_T soient liés d’une quelconque manière, cela a pour finalité d’ajouter du bruit lors de l’alignement, donnant une rotation dont l’alignement ne se fait plus nécessairement sur la moyenne et dont les erreurs d’orientation telles que présentées en figure 2.7 sont plus fréquentes. Une méthode permettant des bootstraps de moyenne gardant une certaine cohérence entre la source et la cible donnerait sûrement de meilleurs résultats et serait à investiguer, cependant nous n’avons pas eu le temps d’aller plus loin dans cette recherche au cours de cette thèse, d’autant plus que la méthode TSAXPCA existait déjà lorsque cette idée d’amélioration a été proposée et jouait un rôle analogue.

2.3 Étude du TSAXPCA élargie à plusieurs paradigme et sur un grand nombre de données

Après avoir développé ces différentes améliorations, on s’est intéressé au résultat que chacune pouvait donner ainsi qu’à l’utilisation de plusieurs d’entre elles en parallèle. En effet, comme précisé un peu plus tôt, rien n’empêche de regrouper les points d’ancrages de plusieurs améliorations afin d’obtenir la meilleure description possible de l’espace de caractéristique qu’on souhaite aligner. Après de nombreux tests, il s’est avéré que la meilleure version du TSA était simplement la version TSAXPCA et globalement toujours les mêmes hyperparamètres à savoir une dimension à la PCA pour chaque classe et généralement un découpage en 5 sous-groupes. C’est donc cette version qui a été publiée dans l’article [15]. Une fois cette version choisie, on a pu s’atteler à obtenir des résultats très détaillés de cette version et ce en utilisant tous les outils à disposition, notamment grâce à la bibliothèque MOABB [43]. Celle-ci donne accès à une grande quantité de données ICO de différents paradigmes qui peuvent être pré-traitées très facilement de façon automatique et normalisée. Ensuite les tests ont été effectués en utilisant tout ce qui a déjà été présenté dans cette thèse et en suivant le diagramme présenté en figure 2.14

2.3.1 Description des données

Lors de l’étude du TSAXPCA, de nombreuses bases de données ont été utilisées. Nous allons reprendre ici leur description ainsi que certaines caractéristiques permettant de mettre en relief les résultats qui seront montrés par la suite. Le tableau 2.15 donne une description détaillée de la quantité de donnée présente dans chacune des bases de données. Celle-ci est particulièrement intéressante car le nombre d’essais qui peuvent être utilisés pour entraîner le clas-

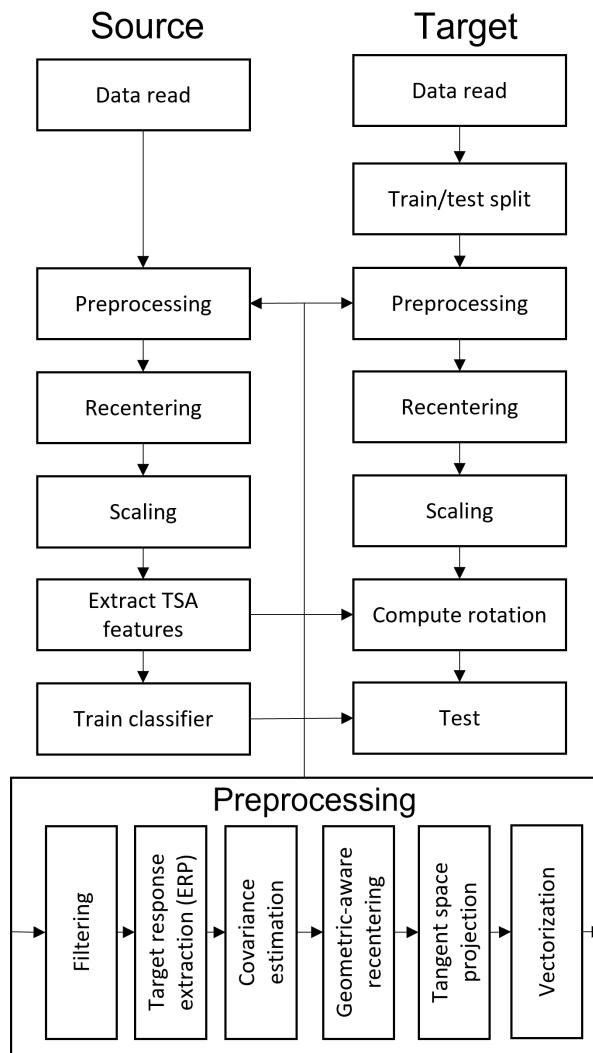


FIGURE 2.14 – Diagramme représentant la pipeline utilisée pour faire les tests de la méthode TSAxPCA.

	Type	Ch	Trials	Sess	Sub
ERP Databases					
003-2015	P300	8	1800	1	10
008-2014	P300	8	4200	1	8
009-2014	P300	16	1728	3	10
Brain Invaders 2013a	P300	16	480	1	24
EPFL P300 dataset	P300	32	3268	4	8
MI Databases					
001-2014	MI	22	288	2	9
001-2015	MI	13	400	2	12
002-2014	MI	15	160	1	14
004-2014	MI	3	720	5	9
Alexandre Motor Imagery	MI	16	40	1	8
Cho2017	MI	64	200	1	49
Grosse-Wentrup 2009	MI	128	300	1	10
Physionet Motor Imagery	MI	64	45	1	109
Weibo 2014	MI	60	160	1	10
Zhou 2016	MI	14	290	3	4
SSVEP Databases					
SSVEP Exoskeleton	SSVEP	8	160	1	12
SSVEP Nakanishi	SSVEP	8	180	1	9
SSVEP Wang	SSVEP	62	240	1	34

FIGURE 2.15 – Liste des bases de données traitées pour les ERP, MI et SSVEP. Légende : Ch=Nombre de canaux ; Trials=Nombre usuel d’essais par sujet ; Sess=Nombre de sessions ; Sub=Nombre de sujets.

sifieur a généralement un impact élevé sur la précision de ce dernier. De même le nombre de canaux est un élément important à prendre en compte car il est aussi déterminant dans la précision finale en plus d’être lié de façon quadratique au temps de calcul des algorithmes dans le domaine des ICO riemanniennes.

Cette étude a été menée sur 5 bases de donnée en ERP citées respectivement avec le même ordre qu’en table 2.15 soit [38], [69], [4], [81] et [39] ; 10 bases de donnée en MI [79], [31], [76], [56], [5], [22], [36], ([74] + [67])⁵, [99], [104] ; et 3 bases en SSVEP [44], [62], [87].

2.3.2 Résultats obtenus

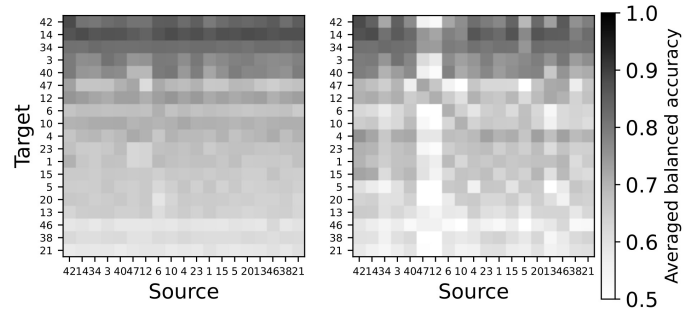
Dans cette étude, l’apprentissage par transfert se fait en inter-sujets pour toutes les paires de sujets source/cible possibles dans chaque base de données. La précision est évaluée en utilisant la précision équilibrée puisque le nombre d’essais par classe est souvent déséquilibré (toujours pour le paradigme ERP).

5. Base de donnée disponible via <https://physionet.org/pn4/eegmidb/>

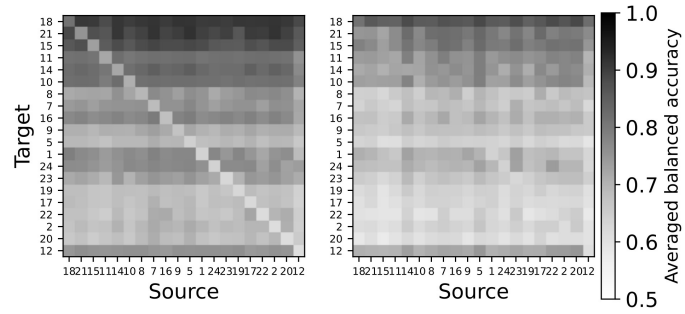
Étant donné la très grande quantité de données de toutes les comparaisons par paire, nous avons commencé par évaluer visuellement toutes les précisions équilibrées obtenues pour une base de données donnée au moyen de diagrammes de sériation, c'est-à-dire des diagrammes montrant la précision pour chaque paire de sujets cible et source disposés sous forme de matrice. La moyenne de la précision est calculée sur tous les nombres d'essais d'alignement pour chaque paire. Le cas où la cible et la source sont identiques, c'est-à-dire sur la diagonale, est remplacé par la précision classique de la validation croisée train-test, qui offre une référence directe. En outre, les sujets cibles et sources sont triés sur les lignes par ordre décroissant de la précision par validation croisée du train-test. Il convient de garder à l'esprit que la procédure de train-test étant entièrement supervisée et optimisée pour les données d'entraînement, elle devrait être plus performante qu'une méthode d'apprentissage par transfert. La figure 2.16 montre un diagramme de sériation représentatif pour chaque paradigme, ce qui permet de comparer visuellement les performances de la méthode d'apprentissage par transfert TSA par rapport à la méthode d'apprentissage par transfert RPA ; toutes les figures sont disponibles en tant que matériel supplémentaire dans la partie Annexe.

Afin d'évaluer la performance moyenne, nous représentons la moyenne de la précision équilibrée entre tous les sujets d'une base de données pour chaque méthode en fonction du nombre d'essais d'alignement. Étant donné que la moyenne est calculée sur l'ensemble des sujets, ne sont inclus dans cette analyse que les sujets présentant une précision d'au moins 60% dans le cadre d'une validation croisée train-test classique. Cette restriction exclut environ la moitié des sujets, soit 178 sujets sur 349. Il est important de noter que les sujets ayant une précision globale de 60% ont généralement une précision supérieure à 70% avec tous les essais d'entraînement disponibles. La figure 2.17 montre un graphique représentatif pour chaque paradigme. Toutes les figures de l'ensemble des bases sont disponibles en tant que matériel supplémentaire.

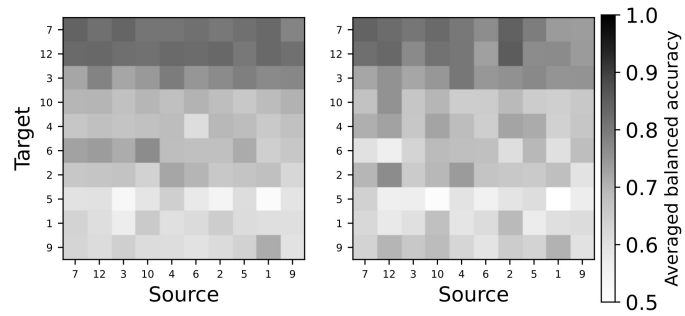
Nous résumons ensuite toutes les informations relatives à la précision par paire dans les tableaux de précision tels que le tableau 2.1. Ces tableaux donnent pour chaque sujet cible la précision moyenne sur le nombre d'essais d'alignement et de sujets sources. Pour cette base de données, il y a 30 nombres d'essais d'alignement considérés et 18 sujets sources possibles, ce qui donne une moyenne sur 540 valeurs. L'erreur standard est donc généralement faible. Les tableaux de précision pour toutes les bases de données figurent en annexe.



(a) Base de données Cho2017 (MI)

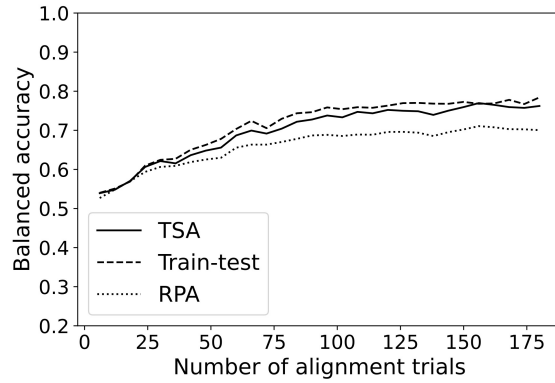


(b) Base de données Brain Invaders 2013a (ERP)

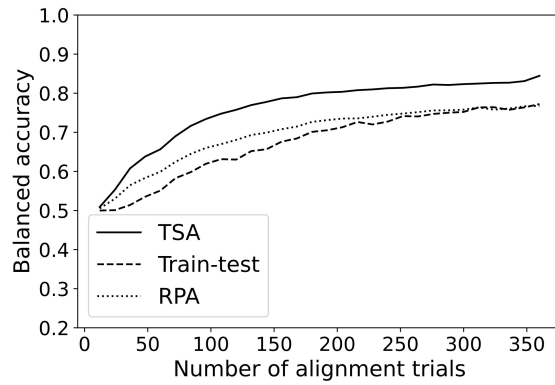


(c) Base de données SSVEP Exoskeleton (SSVEP)

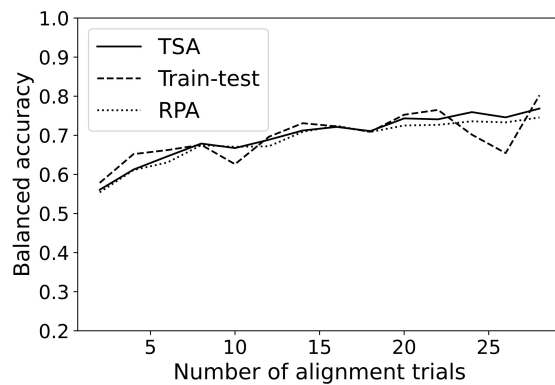
FIGURE 2.16 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour chaque paradigme. Voir le texte pour plus de détails.



(a) Base de données Cho2017 (MI)



(b) Base de données Brain Invaders 2013a (ERP)



(c) Base de données SSVEP Exoskeleton (SSVEP)

FIGURE 2.17 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour une base de données représentative dans chaque paradigme (MI, ERP, SSVEP).

TABLE 2.1 – Précision équilibrée \pm erreur type pour les bons sujets de la base de données Cho2017 moyennée sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA.

	Train-test acc	TSA acc	RPA acc
Subject 1	67.86 \pm 2.00	66.70 \pm 0.43	65.84 \pm 0.43
Subject 3	81.39 \pm 0.81	77.78 \pm 0.18	74.96 \pm 0.32
Subject 4	69.26 \pm 1.49	69.24 \pm 0.40	69.25 \pm 0.50
Subject 5	65.90 \pm 2.12	65.03 \pm 0.50	59.12 \pm 0.55
Subject 6	70.74 \pm 2.11	67.65 \pm 0.38	59.80 \pm 0.53
Subject 10	70.55 \pm 1.91	70.71 \pm 0.45	62.32 \pm 0.43
Subject 12	70.97 \pm 1.61	73.01 \pm 0.39	67.68 \pm 0.29
Subject 13	63.60 \pm 1.14	64.30 \pm 0.25	60.83 \pm 0.25
Subject 14	88.26 \pm 1.52	86.99 \pm 0.37	82.06 \pm 0.45
Subject 15	66.15 \pm 1.64	65.63 \pm 0.38	65.20 \pm 0.37
Subject 20	64.96 \pm 1.13	64.64 \pm 0.26	60.79 \pm 0.33
Subject 21	60.63 \pm 1.50	58.63 \pm 0.24	56.54 \pm 0.30
Subject 23	68.64 \pm 1.62	66.69 \pm 0.37	64.94 \pm 0.41
Subject 34	82.73 \pm 0.37	81.89 \pm 0.09	79.97 \pm 0.15
Subject 38	61.98 \pm 1.27	61.51 \pm 0.27	58.53 \pm 0.35
Subject 40	76.73 \pm 1.89	76.30 \pm 0.44	72.21 \pm 0.47
Subject 42	88.34 \pm 3.32	83.02 \pm 0.73	75.78 \pm 0.77
Subject 46	62.55 \pm 1.79	59.08 \pm 0.37	54.64 \pm 0.26
Subject 47	72.34 \pm 1.66	67.98 \pm 0.36	63.03 \pm 0.42

Les tableaux de précision confirment ce qui peut être évalué visuellement dans les diagrammes de précision moyenne et les diagrammes de sériation ; en moyenne, il y a une différence d’environ 1% entre la précision de la validation croisée du test d’entraînement classique et la méthode TSA et d’environ 5% entre la validation croisée du test d’entraînement classique et la méthode RPA. Cela plaide en faveur d’une nette amélioration de la méthode TSA par rapport à la méthode RPA. Le tableau 2.2 résume toutes les précisions équilibrées pour chaque ensemble de données. En moyenne, sur l’ensemble des bases de données, il n’y a pas de perte de précision avec la méthode TSA par rapport à la précision optimale de l’entraînement et du test. Ce n’est pas le cas pour la RPA.

Enfin, nous avons effectué des tests statistiques sur tous les résultats de précision source/cible par paire que nous avons recueillis. À cette fin, nous avons suivi la procédure introduite dans [70]. En bref : on calcule d’abord des t-tests par paires signés pour chaque sujet cible en comparant la précision entre les méthodes, ce qui donne des statistiques $T_{m,i}$ et des valeurs $p_{m,i}$ pour chaque paire de méthodes m et de sujets cibles i . Afin de corriger la multiplicité des tests

TABLE 2.2 – Précision équilibrée \pm erreur standard pour les bons sujets de chaque base de données en moyenne sur le nombre d’essais d’alignement, la cible et la source pour le train-test, TSA et RPA.

	Train-test acc	TSA acc	RPA acc
ERP Databases			
003-2015	69.04 \pm 0.74	76.05 \pm 0.57	69.63 \pm 0.57
008-2014	67.52 \pm 0.69	72.00 \pm 0.38	65.50 \pm 0.38
009-2014	70.48 \pm 0.50	77.38 \pm 0.21	70.90 \pm 0.21
Brain Invaders 2013a	67.26 \pm 0.45	76.07 \pm 0.11	69.59 \pm 0.11
EPFL P300 dataset	67.22 \pm 0.61	73.13 \pm 0.24	65.28 \pm 0.24
MI Databases			
001-2015	81.13 \pm 0.73	79.01 \pm 0.24	76.21 \pm 0.24
002-2014	78.40 \pm 0.70	76.51 \pm 0.20	72.55 \pm 0.20
004-2014	74.35 \pm 1.30	76.00 \pm 0.59	75.27 \pm 0.59
Alexandre Motor Imagery	78.51 \pm 2.82	76.30 \pm 1.33	75.93 \pm 1.33
Cho2017	71.24 \pm 0.52	69.83 \pm 0.12	65.97 \pm 0.12
Grosse-Wentrup 2009	77.20 \pm 0.91	73.12 \pm 0.34	72.16 \pm 0.34
Physionet Motor Imagery	70.75 \pm 1.35	64.28 \pm 0.27	63.26 \pm 0.27
Weibo 2014	73.97 \pm 1.25	69.66 \pm 0.50	71.27 \pm 0.50
Zhou 2016	82.82 \pm 1.59	82.43 \pm 0.86	81.31 \pm 0.86
SSVEP Exoskeleton	69.50 \pm 1.16	69.69 \pm 0.40	68.72 \pm 0.40
SSVEP Databases			
SSVEP Nakanishi	95.55 \pm 0.82	95.95 \pm 0.28	96.83 \pm 0.28
SSVEP Wang	68.26 \pm 2.63	57.42 \pm 0.76	59.11 \pm 0.76
Global	74.42 \pm 0.30	74.49 \pm 0.12	71.79 \pm 0.12

statistiques, nous avons utilisé la procédure de test multiple de rejet séquentiel de Holm ([40]) pour chaque sujet cible. Cette procédure produit des tableaux tels que le tableau 2.3. Les tableaux correspondants pour chaque base de données sont disponibles en annexe. Ensuite, on combine les valeurs p obtenues à l’aide de la méthode du score Z de Stouffer ([102]) pour chaque base de données, ce qui donne plusieurs valeurs p correspondant à chaque paire de méthodes pour chaque base de données. Ces valeurs p sont également corrigées au moyen de la procédure de Holm et sont résumées dans le tableau 2.4. Ce tableau montre que parmi les 18 bases de données analysées, 11 montrent une amélioration significative de TSA par rapport à RPA. Aucune différence significative n’a été constatée entre la précision de la validation croisée train-test classique et TSA, à l’exception de deux bases de données pour lesquelles TSA s’est avéré inférieur. Ce nombre est porté à cinq bases de données pour lesquelles la précision de la validation croisée du test de formation classique est comparée à la précision obtenue par la RPA. Enfin, à l’aide de la méthode du score Z de Stouffer, les valeurs p correspondant à chaque paradigme sont calculées et corrigées à l’aide

TABLE 2.3 – Valeurs p par sujet pour la base de données MI Cho2017. Légende : TRAIN=train-test ; *=valeurs p significatives après correction des comparaisons multiples.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.002*	0.818	0.420
Subject 3	0.033*	<0.001*	<0.001*
Subject 4	0.005*	0.690	0.248
Subject 5	<0.001*	0.980	0.301
Subject 6	0.001*	0.379	0.013*
Subject 10	<0.001*	0.503	0.007*
Subject 12	<0.001*	0.086	<0.001*
Subject 13	0.001*	0.997	0.700
Subject 14	0.443	<0.001*	<0.001*
Subject 15	0.004*	0.964	0.843
Subject 20	0.385	0.991	0.988
Subject 21	<0.001*	1.000	0.999
Subject 23	0.009*	0.836	0.514
Subject 34	<0.001*	<0.001*	<0.001*
Subject 38	<0.001*	1.000	0.980
Subject 40	0.020*	0.002*	<0.001*
Subject 42	0.361	<0.001*	<0.001*
Subject 46	0.126	1.000	0.995
Subject 47	<0.001*	0.038*	<0.001*

de la procédure de Holm (tableau 2.5).

Jusqu'à présent, nous nous sommes concentrés sur l'apprentissage par transfert entre sujets, mais la méthode proposée peut également être utilisée pour transférer différentes informations, par exemple d'une session à l'autre avec le même sujet ou d'une tâche à l'autre. Afin de garantir la capacité de la méthode à réduire la variabilité entre les sessions, nous avons utilisé l'ensemble de données comportant plusieurs sessions et traité les résultats pour une validation croisée entre les sessions. Pour ce faire, nous avons utilisé les données d'une session comme source, puis pour chaque autre session avec 80% de données de test et 20% de données d'entraînement, nous avons entraîné le modèle d'apprentissage par transfert et testé les résultats. Le processus est ensuite répété en utilisant chaque session comme source. Nous avons comparé quatre méthodes différentes :

- Tangent Space Alignment (TSA),
- Riemannian Procrustes Analysis (RPA) utilisée comme comparaison lors

TABLE 2.4 – Valeurs p par base de données. Légende : TRAIN=train-test ;
 *=valeurs p significatives après correction des comparaisons multiples.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
ERP Databases			
003-2015	<0.001*	1.000	0.663
008-2014	0.002*	0.877	0.326
009-2014	<0.001*	1.000	0.561
Brain Invaders 2013a	<0.001*	1.000	1.000
EPFL P300 dataset	<0.001*	1.000	0.102
MI Databases			
001-2014	0.004*	0.468	0.191
001-2015	<0.001*	0.133	0.005*
002-2014	<0.001*	0.139	<0.001*
004-2014	0.515	0.641	0.552
Alexandre Motor Imagery	0.636	0.353	0.280
Cho2017	<0.001*	0.098	<0.001*
Grosse-Wentrup 2009	0.097	0.083	0.071
Physionet Motor Imagery	<0.001*	<0.001*	<0.001*
Weibo 2014	1.000	0.099	0.189
Zhou 2016	0.071	0.453	0.398
SSVEP Databases			
SSVEP Exoskeleton	0.002*	0.702	0.477
SSVEP Nakanishi	1.000	0.464	0.886
SSVEP Wang	0.989	<0.001*	<0.001*

TABLE 2.5 – Valeurs p pour chaque paradigme et valeurs p globales pour tous les tests effectués. Les valeurs p globales sont la combinaison des valeurs p pour toutes les bases de données, quel que soit leur paradigme. Légende : TRAIN=train-test ; *=valeurs p significatives après correction des comparaisons multiples.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
p300	<0.001*	1.000	0.961
imagery	<0.001*	<0.001*	<0.001*
ssvep	0.954	<0.001*	<0.001*
global	<0.001*	1.000	<0.001*

cette étude,

- Méthode habituelle de train-test utilisant uniquement les données cibles,
- Test direct (DCT) utilisant des algorithmes entraînés sur la source sans aligner les données cibles par une rotation (recentrage uniquement).

Les résultats sont présentés dans la figure 2.18 et présentés numériquement dans le tableau 2.6.

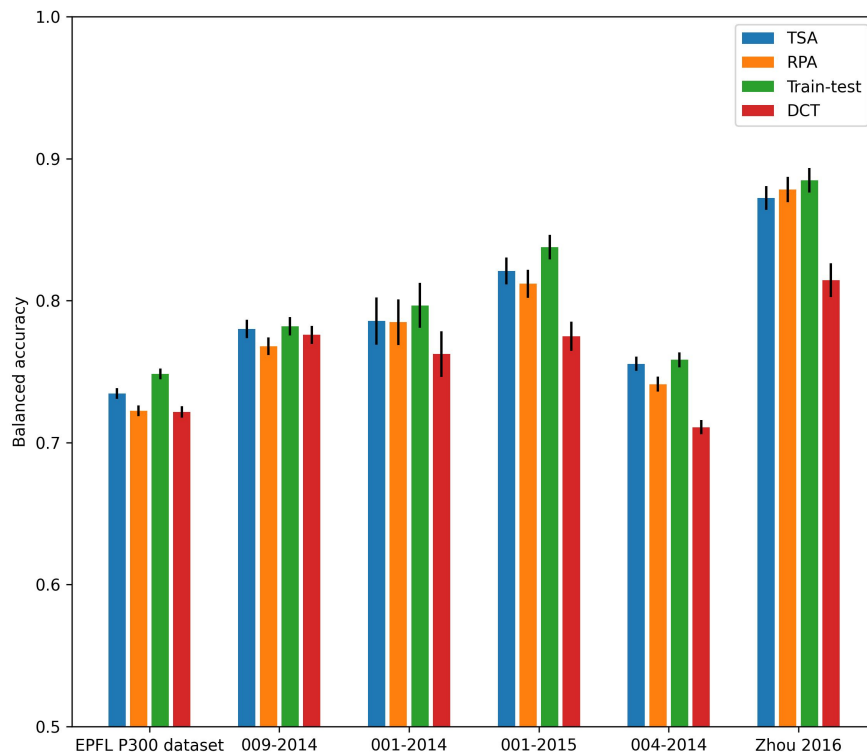


FIGURE 2.18 – Graphique à barres indiquant la précision équilibrée entre les sessions pour les bases de données possédant plusieurs sessions.

TABLE 2.6 – Tableau de précision équilibrée entre les sessions pour chaque ensemble de données et chaque méthode.

	Train-test	TSA	RPA	DCT
EPFL P300	74.85 ± 0.38	73.46 ± 0.38	72.24 ± 0.39	72.16 ± 0.40
009-2014	78.19 ± 0.64	78.00 ± 0.65	76.79 ± 0.63	77.59 ± 0.64
001-2014	79.66 ± 1.57	78.56 ± 1.66	78.48 ± 1.61	76.23 ± 1.60
001-2015	83.78 ± 0.86	82.10 ± 0.95	81.19 ± 0.99	77.49 ± 1.03
004-2014	75.83 ± 0.52	75.55 ± 0.50	74.11 ± 0.53	71.09 ± 0.50
Zhou 2016	88.48 ± 0.86	87.22 ± 0.84	87.83 ± 0.90	81.45 ± 1.19

Ces résultats sont cohérents avec les précédents. Ils donnent des précisions légèrement améliorées par rapport aux sessions mélangées toutes ensemble, ce qui est attendu. De plus, la méthode proposée est plus performante que RPA sur toutes les bases de données à l’exception de Zhou 2016.

2.3.3 Discussion des résultats obtenus avec la TSaxPCA

L’analyse approfondie que nous avons effectuée montre que pour le paradigme ERP, la TSaxPCA est nettement plus performant que la RPA. Pour le paradigme MI, nous observons que TSA est plus performant que RPA et que la validation croisée train-test classique est plus performante que TSA et RPA. Pour le paradigme SSVEP, la validation croisée train-test classique est plus performante que TSA et RPA. Globalement, la RPA est surclassée par la TSA et la validation croisée train-test classique. Sur la base du tableau 2.5, nous ne pouvons pas conclure quant à la supériorité de TSA par rapport au train-test en termes de précision, car la valeur globale qui peut être trouvée est principalement due à de très grandes valeurs trouvées pour le paradigme ERP. Pour conclure avec les résultats, il a été démontré que TSA surpasse RPA pour la grande majorité des bases de données que nous avons utilisées dans cette analyse, atteignant une précision assez proche de la méthode optimale de train-test. Il s’agit également d’une nette amélioration d’un point de vue méthodologique par rapport à la RPA, car la version de base de la TSA permet naturellement l’apprentissage par transfert entre des ensembles de données comportant un nombre différent de canaux, alors que la RPA a besoin de certaines extensions pour ce faire ([70]). Il reste cependant à noter que la version améliorée TSaxPCA utilisée ici a par contre aussi besoin d’une légère adaptation pour être étendue à un nombre différents de canaux. L’avantage indéniable est que cette adaptation peut se faire directement à l’intérieur de la méthode, comme il sera montré un peu plus tard et n’a pas besoin d’adapter les données contrairement à la méthode de dimensionality transcending (DT) [70] qui permet l’adaptation de la RPA aux bases à nombre de canaux hétérogènes.

En conclusion, cette étude a permis d’introduire une nouvelle méthode d’apprentissage par transfert, ou a minima son amélioration fonctionnant à la fois en inter- et intra-sujets pour les interfaces cerveau-ordinateur. Cette étude indique également que cette méthode est plus performante qu’une méthode analogue

de pointe (RPA). Cependant, elle n'atteint pas encore la même précision que celle obtenue avec une procédure classique de validation croisée train-test pour l'imagerie motrice et le paradigme SSVEP. Des recherches supplémentaires sont nécessaires pour comprendre pourquoi la performance de la méthode TSA est clairement supérieure pour le paradigme ERP. Une hypothèse est que cette bonne performance peut-être liée soit au déséquilibre dans les classes des ERP que la TSA prendrait bien en compte (bien que la méthode de train-test soit adaptée pour la prendre aussi en compte, de même pour la RPA), soit simplement à la forme de la distribution des ERP qui se prête bien à l'utilisation de cette méthode.

En termes de temps de calcul, puisque nous disposons d'une forme déterministe pour la rotation de la méthode TSA, celle-ci est beaucoup plus rapide que la méthode RPA, où une optimisation itérative sur la variété de Grassmann est effectuée. Cependant, même si la TSA peut être plus rapide d'un ordre de grandeur, cette rapidité a tendance à évoluer avec le nombre de canaux considérés. Soit N_c étant la taille du signal de pré-covariance (nombre de canaux plus nombre de canaux de la cible moyenne pour ERP, nombre de canaux pour MI, nombre de classes multiplié par le nombre de canaux pour SSVEP), nous calculons des rotations de taille $(\frac{N_c*(N_c+1)}{2}, \frac{N_c*(N_c+1)}{2})$ là où RPA calcule une rotation de taille (N_c, N_c) . Cela signifie que pour les ensembles de données comportant un nombre important de canaux, l'avantage de calcul de TSA tendra à disparaître voire à s'inverser dans le pire des cas. Le tableau 2.7 montre le temps de calcul moyen pour chaque ensemble de données si aucun filtre spatial n'était appliqué avant l'estimation de la covariance, ce qui donne parfois de grandes matrices de covariance. En général, les filtres spatiaux sont appliqués, de sorte que les valeurs de temps les plus faibles sont celles que l'on rencontre habituellement. De plus, lors du calcul des rotations pour TSA, nous pouvons considérer seulement un nombre de canaux qui résulteront en une proportion de la variance globale des données. Cela signifie qu'il n'est pas nécessaire de calculer l'ensemble de la matrice de rotation $(\frac{N_c*(N_c+1)}{2}, \frac{N_c*(N_c+1)}{2})$, nous pouvons sélectionner uniquement quelques valeurs singulières et leurs vecteurs correspondants, ce qui permet de réduire considérablement le temps de calcul si nécessaire⁶.

Nous avons également observé que dans certains cas, principalement pour les ERP, omettre la remise à l'échelle des données source et cible présentées dans les équations 2.2 et 2.3 donne de meilleurs résultats lors de la classification. Ces observations n'ont cependant pas été quantifiées lors d'une réelle étude et des recherches supplémentaires sont nécessaires pour comprendre pleinement le rôle de la remise à l'échelle dans les méthodes d'apprentissage par transfert de type Procruste.

Lors de cette étude, nous avons pu proposer une procédure visant à améliorer l'insuffisance du rang de la matrice des produits croisés, ce qui rend le résultat plus stable et plus précis. Toutefois, cette amélioration présente éga-

6. En effet il est possible d'utiliser des algorithmes de calcul de la SVD où seul un certain nombre, donné à l'avance, de valeurs singulières sont calculées mais cela n'a pas été fait lors de ces travaux.

TABLE 2.7 – Temps de calcul moyen par base de données pour les méthodes TSA et RPA.

	TSA (s)	RPA (s)	Ratio	Cov shape
ERP Databases				
003-2015	1.027	11.889	11.57	16x16
008-2014	1.711	10.874	6.36	16x16
009-2014	0.897	6.118	6.82	32x32
Brain Invaders 2013a	0.774	13.368	17.26	32x32
EPFL P300 dataset	1.341	8.607	6.42	64x64
MI Databases				
001-2014	0.999	5.075	5.08	22x22
001-2015	0.664	4.362	6.57	13x13
002-2014	0.416	2.241	5.39	15x15
004-2014	0.029	0.166	5.74	3x3
Alexandre Motor Imagery	0.067	0.777	11.61	16x16
Cho2017	5.299	16.858	3.18	64x64
Grosse-Wentrup 2009	161.942	113.926	0.70	128x128
Physionet Motor Imagery	2.739	4.432	1.62	64x64
Weibo 2014	3.719	13.282	3.57	60x60
Zhou 2016	0.468	3.752	8.02	14x14
SSVEP Databases				
SSVEP Exoskeleton	0.056	1.184	21.22	16x16
SSVEP Nakanishi	0.041	0.970	23.81	16x16
SSVEP Wang	125.949	22.320	0.18	124x124

lement quelques inconvénients. Lorsque l'on utilise uniquement le point moyen des données pour les données source et cible, la méthode sera assez sensible au bruit puisque le nombre de points utilisés pour le calcul des moyennes est considérablement réduit. L'ajout de moyennes et/ou de médianes tronquées pourrait rendre la méthode moins sensible au bruit. En outre, lors de l'augmentation de \mathbf{K} à l'aide de la PCA, plus le nombre de groupes créés est important, plus chaque groupe est sensible au bruit. En réduisant le nombre de points dans un groupe, on augmente l'impact des artefacts sur le point moyen. En outre, la PCA est un algorithme assez sensible au bruit et pourrait être remplacée par une de ses variantes robustes. Il est à noter qu'un faible nombre de groupes permet en général une bonne approximation de la forme des données. Bien entendu, la correction ou l'élimination des artefacts permettrait d'obtenir de meilleures performances.

Il convient de rappeler que lors de cette étude, nous n'avons appliqué aucune forme de pré-traitement poussé afin d'augmenter la précision globale. Il a été choisi de procéder ainsi car l'objectif de l'étude est de comparer les méthodes d'apprentissage par transfert sur une grande quantité de données et dans une grande variété de situations réelles, c'est-à-dire sur des données bruitées. Pour la

même raison, nous avons utilisé toutes les données d'un sujet sans faire d'adaptation d'une session à l'autre. Ce choix entraîne évidemment une baisse de la précision globale, ce qui se traduit par une diminution importante du nombre de sujets utilisés pour les résultats moyens finaux. Il a été jugé intéressant de procéder ainsi car, comme l'ont montré des études antérieures, il existe un écart important de performance des algorithmes entre les sujets en ce qui concerne la précision de l'apprentissage par transfert ([101, 71]).

Un autre point à mentionner est que même si notre méthode a été largement testée sur de nombreuses bases de données, il y a encore plus de bases de données à tester. En outre, certains paradigmes tels que les ICO affectives n'ont pas été étudiés dans cet article. L'étude de l'apprentissage par transfert entre bases de données n'a pas encore été abordé et sera d'ailleurs présentée dans la suite de cette thèse.

Comme toutes les méthodes efficaces d'apprentissage par transfert, la TSA peut être très utile lorsqu'elle est utilisée avec un modèle d'apprentissage automatique qui prend trop de temps pour l'entraînement sur de nouvelles données pour des sessions en ligne. la TSA permet également d'aligner les données de plusieurs sujets dans le même espace de caractéristiques. Cet alignement pourrait améliorer la précision de la classification de plusieurs sujets et permettre l'apprentissage d'un classificateur robuste sur des données alignées qui donnera de meilleurs résultats pour de nouveaux sujets une fois qu'ils auront été alignées.

Chapitre 3

Apprentissage par transfert inter-configurations

Table des matières

3.1	Généralisation inter-configurations du TSAxPCA	82
3.2	La source arbitraire	84
3.3	Résultats intra- et inter-configurations de la source arbitraire	88
3.3.1	Résultats intra-configuration	90
3.3.2	Résultats inter-configurations	94
3.3.3	Discussion	101

3.1 Généralisation inter-configurations du TSAxPCA

Le TL considéré jusque là est essentiellement d'un sujet cible unique vers un autre sujet source unique également. Pour aligner plusieurs sujets ensemble on peut choisir un sujet source et aligner tous les autres à lui. Cependant dès lors que cette stratégie est choisie, elle soulève plusieurs autres questions, notamment car le choix d'un sujet source vers qui aligner tout le monde plutôt qu'un autre n'est pas anodin. Il convient alors de se demander quel sujet conviendrait le mieux en tant que source globale, d'autant plus lorsque l'on se place, comme ici, dans le cadre de TL inter-configurations, c'est à dire entre des bases de données possédant des configurations souvent très différentes (nombre d'électrodes, rap-

port signal/bruit, stimuli envoyé à l'utilisateur...). Toute cette troisième partie portera sur la mise en place et les tests d'une source arbitraire mathématique possédant de bonnes propriétés et pouvant permettre d'aligner entre eux différents sujets y compris en inter-configurations en perdant le moins possible d'information. Dans le cadre de cette étude la méthode d'alignement restait libre et nous avons donc choisi de rester dans la continuité de cette thèse et de prendre la méthode TSAxPCA que nous avons présentée plus tôt pour jouer le rôle de méthode d'alignement.

Comme nous avons pu l'expliquer dans la partie 2.2.4, contrairement à la première version de la méthode TSA, nous ne calculons pas le produit croisé directement avec les points d'ancrage de base, mais nous ajoutons d'autres points d'ancrage basés sur la composante principale de l'ensemble de points pour chaque classe. De cette manière, nous calculons les matrices de covariance des deux ensembles $\{\tilde{\mathbf{z}}_{S,n}\}_{n \in [1, N_s]}$ et $\{\tilde{\mathbf{z}}_{T,n}\}_{n \in [1, N_t]}$ pour chaque classe k , ce qui donne les matrices $\tilde{\mathbf{C}}_{S,k}$ et $\tilde{\mathbf{C}}_{T,k}$ puis, avec la décomposition en valeurs propres, la composante principale $\mathbf{e}_{S,k}$ et $\mathbf{e}_{T,k}$ de chaque ensemble de points et de chaque classe est extraite. Toutefois, contrairement à la méthode précédente dans laquelle une composante principale était calculée uniquement pour la source et appliquée à la cible pour le tri, la source et la cible ne sont pas nécessairement de la même taille dans le cas de l'ensemble croisé TL. Par conséquent, une composante principale est calculée indépendamment pour la source et la cible. On se retrouve alors dans le cas analogue à ce qui se passe pour la version TSAxCSP de la méthode en partie 2.2.5. Dans ce cas particulier, une ambiguïté de signe peut survenir entre la direction principale de la source et celle de la cible. Dès lors, nous considérons la manière la plus simple de résoudre l'ambiguïté de signe, en calculant le produit scalaire entre la composante et un vecteur tout-unique de même longueur que nous appellerons \mathbf{j} .

$$\tilde{\mathbf{e}}_{S,k} = \text{sgn}(\mathbf{e}_{S,k} \cdot \mathbf{j}) \mathbf{e}_{S,k}$$

et

$$\tilde{\mathbf{e}}_{T,k} = \text{sgn}(\mathbf{e}_{T,k} \cdot \mathbf{j}) \mathbf{e}_{T,k}$$

avec $\text{sgn}(\cdot)$ la fonction signe qui vaut -1 sur $] -\infty, 0[$, 0 en 0 et 1 sur $]0, +\infty[$.

Une autre méthode inspirée par ([18]), aurait pu consister à calculer le signe de $\mathbf{e}_{S,k} \cdot \mathbf{e}_{T,k}$ et à modifier le signe de $\mathbf{e}_{S,k}$ ou de $\mathbf{e}_{T,k}$ pour que ce signe soit positif, ce qui aurait permis d'obtenir un vecteur aligné. Cependant, cette méthode exige également que $\mathbf{e}_{S,k}$ et $\mathbf{e}_{T,k}$ aient la même longueur et n'est donc pas adaptée à l'alignement de bases de données hétérogènes, c'est pourquoi je m'en suis tenu à la référence du vecteur de uns pour corriger l'ambiguïté de signe. Cette méthode quant à elle est tout de même utilisée pour corriger le signe dans le cas inter-sujets. Celle-ci permet d'obtenir de meilleurs résultats par rapport à l'utilisation d'une seule composante principale extraite de la source, car elle permet un meilleur regroupement des points de la cible.

Ces composantes principales sont ensuite utilisées comme dans [15] pour trier les ensembles $\{\tilde{\mathbf{z}}_{S,n}\}_{n \in [1, N_t]}$ et $\{\tilde{\mathbf{z}}_{T,n}\}_{n \in [1, N_t]}$ et les regrouper en $N_{cluster}$ groupes en fonction de leur position le long de la composante principale. Dans cette

étude, nous avons choisi $N_{cluster} = 5$ groupes le long de la première composante principale de chaque classe pour la source et la cible. La moyenne de chaque groupe donnera les matrices de points d’ancrage supplémentaires $\mathbf{A}_S = [\mathbf{a}_{S,i}, i \in [1, N_{cluster} * K]]$ pour la source et $\mathbf{A}_T = [\mathbf{a}_{T,i}, i \in [1, N_{cluster} * K]]$ pour la cible comme expliqué plus tôt en partie 2.2.4. Enfin, nous pouvons définir

$$\bar{\mathbf{Z}}_S \leftarrow [\bar{\mathbf{Z}}_S, \mathbf{A}_S] \text{ et } \bar{\mathbf{Z}}_T \leftarrow [\bar{\mathbf{Z}}_T, \mathbf{A}_T].$$

de manière analogue à ce qui a été défini en partie 2.2.4 et plus précisément en équation 2.12.

3.2 La source arbitraire

Depuis sa première introduction dans le domaine ICO par Alexandre Barachant en 2010 ([6]) la géométrie riemannienne a montré un grand potentiel, tant en termes de classification ([26]) que de TL et ce pour plusieurs types de paradigmes tels que le P300 ([71]), ([57]), la MI ([101]), les SSVEP ([45]) et les c-VEP ([100]). Tous ces résultats prometteurs montrent le potentiel des méthodes riemanniennes pour réduire l’écart entre deux sujets ou sessions et donc augmenter la vitesse d’étalonnage. Cependant, il n’existe que peu de méthodes traitant de la variabilité entre les ensembles de données et encore moins utilisant la géométrie riemannienne.

Les auteurs de ([91]) présentent une revue de méthodes existantes pour chaque paradigme ICO dans lequel le TL inter-configurations est pris en compte, même si le nombre de méthodes est faible. De cette manière, les auteurs de ([95]) montrent la variabilité entre les ensembles de données MI et proposent une stratégie de pré-alignement en ligne pour la réduire à l’aide de la géométrie riemannienne. Dans ([92]), les auteurs décrivent une méthode qui permet le TL d’un bonnet EEG à l’autre pour les applications ICO P300 en utilisant des données étiquetées. Ils montrent une augmentation de la précision avec le même nombre de données étiquetées pour le nouveau casque. Dans la continuité de ce type de méthodes, les auteurs de ([70]) ont développé une méthode qui vise à aligner des données de taille différente dans la variété riemannienne des matrices de covariance. Ils parviennent à transférer des informations pertinentes qui seraient autrement impossibles à obtenir.

L’étude que nous avons mené autour du transfert d’apprentissage inter-configurations dans le cadre de cette thèse s’inscrit dans la continuité de ces méthodes et introduit une nouvelle approche qui vise à conserver autant d’informations que possible lors de l’arrangement des données d’un ensemble de données à un autre. Les détails de cette méthode font l’objet d’un article en cours de soumission. Ainsi, la généralisation de la variante TSAxPCA à des données source et cible de forme différente ne représente qu’une petite partie de la contribution de cette partie de la thèse à la communauté scientifique. La partie la plus importante est la création et l’utilisation d’une source arbitraire créée mathématiquement pour servir de base à l’alignement de toutes les données sur cette source. L’objectif final est de pouvoir créer une base de données étalon de

grande taille permettant d'aligner n'importe quel ensemble de données connu en perdant le moins d'informations possible. Pour ce faire, nous avons recherché une nouvelle source de conception "arbitraire", ce qui signifie qu'elle n'est pas liée à d'autres structures de données telles que les bases de données elle-mêmes. L'idée ici est de comparer l'utilisation de cette source créée artificiellement pour posséder des propriétés mathématiques pertinentes au sujet ayant la meilleure précision dans un ensemble de données. Si ce type de source est capable de servir de source optimale pour le TL inter-configurations, cela peut s'avérer très utile à l'avenir, car ce sera un bon moyen d'améliorer et de simplifier la recherche autour des questions suivantes : "Quel est le meilleur sujet à utiliser comme source dans un ensemble de données ?" ou "Comment puis-je envoyer les données de ma nouvelle configuration aux données des utilisateurs précédents que j'ai déjà ?". Cette conception arbitraire de la source vise en effet à réduire la dépendance des alignements par rapport à l'ensemble de données source, qui peut actuellement être importante si toutes les données sont alignées sur un sujet de l'ensemble pour l'alignement intra-ensemble de données, ou pire encore, dans le cas de l'alignement inter-configurations, sur un seul sujet d'un seul ensemble de données. Ce choix permet de définir la source arbitraire définie mathématiquement comme ayant certaines bonnes propriétés souhaitables, et de ne pas passer trop de temps à rechercher quel est le "meilleur" sujet en termes d'alignement. Nous montrerons également que ce choix particulièrement attrayant n'a pas beaucoup d'impact sur la précision finale équilibrée par rapport au "meilleur" sujet (c'est à dire le meilleur en termes de précision équilibrée train-test à sujet unique). Pour ce faire, nous avons appliqué la méthode décrite dans ([15]) sur plusieurs ensembles de données avec deux archétypes principaux : 1) TL intra-ensemble et 2) TL inter-ensembles. Dans le premier cas, nous allons nous intéresser au TL inter-sujets à l'intérieur d'un ensemble de données et donc sans qu'il soit nécessaire de modifier la forme des données, ce qui facilite le transfert. Dans le second cas, nous allons nous pencher sur le TL inter-configurations, qui nécessite généralement une modification de la forme des données.

Nous avons ainsi conçu une source mathématique présentant les propriétés suivantes. Considérons un ensemble de points $\{\bar{\mathbf{x}}_k\}_{k \in [1, K]}$ représentant le centre de K classes. Tout d'abord, étant donné les propriétés des données réelles dont nous disposons, nous voulons que cet ensemble de points soit centré, ce qui donne

$$\frac{1}{K} \sum_k \bar{\mathbf{x}}_k = \mathbf{0}. \quad (3.1)$$

De même, nous voulons que ces points aient tous la même norme et comme il n'y a qu'un coefficient multiplicatif entre le choix de deux normes, prenons-le égal à 1. Cela donne l'équation suivante :

$$\forall k \in [1, K], \|\bar{\mathbf{x}}_k\| = 1 \quad (3.2)$$

qui se traduit par

$$\forall k \in [1, K], \sum_i \bar{x}_{k,i}^2 = 1$$

avec $\bar{x}_{k,i}$ étant la $i^{\text{ème}}$ valeur dans $\bar{\mathbf{x}}_k$.

Une autre condition que nous avons ajoutée pour créer cette source artificielle est l'égalité de la distance entre deux classes. Cette condition vise à ce qu'aucune classe ne soit prioritaire sur une autre afin d'avoir une source aussi générique que possible. Cette nouvelle règle peut être définie comme suit

$$\forall i \neq j, k \neq l, \|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\| = \|\bar{\mathbf{x}}_k - \bar{\mathbf{x}}_l\| \quad (3.3)$$

qui peut être développée en

$$\forall i \neq j, k \neq l, \sum_m (\bar{x}_{i,m} - \bar{x}_{j,m})^2 = \sum_m (\bar{x}_{k,m} - \bar{x}_{l,m})^2$$

Après calculs, la source la plus simple qui répondrait à la plupart des règles serait :

$$\forall k \in [1, K], \forall i \in [1, K], \bar{x}_{k,i} = \begin{cases} 1, & i = k \\ -\frac{1}{K-1}, & i \neq k \end{cases} \quad (3.4)$$

Cette définition est conforme à toutes les règles, à l'exception d'une seule. En effet

$$\begin{aligned} \frac{1}{K} \sum_k \bar{\mathbf{x}}_k &= \frac{1}{K} \left(1 + \sum_{k=1}^{K-1} \frac{-1}{K-1} \right) \\ &= 0. \end{aligned}$$

L'équation 3.1 est donc satisfaite. En outre, dans l'équation

$$\begin{aligned} \forall k \neq l \in [1, K], \sum_i (\bar{x}_{k,i} - \bar{x}_{l,i})^2 &= \left(1 - \frac{1}{K-1}\right)^2 + \left(-\frac{1}{K-1} + 1\right)^2 \\ &\quad + \sum_{i=1}^{K-2} \left(-\frac{1}{K-1} - \frac{1}{K-1}\right)^2 \\ &= 2 \left(\frac{K-2}{K-1}\right)^2 + (K-2) \left(\frac{2}{K-1}\right)^2 \\ &= \frac{2K(K-2)}{(K-1)^2} \end{aligned}$$

puisque le résultat ne dépend pas de k ou l s'ils sont différents, les conditions données dans l'équation 3.3 sont remplies.

Il ne reste plus qu'à satisfaire l'équation 3.2. Nous constatons que

$$\begin{aligned}
\forall k \in [1, K], \sum_i \bar{x}_{k,i}^2 &= 1 + \sum_{i=1}^{K-1} \left(-\frac{1}{K-1} \right)^2 \\
&= 1 + \frac{K-1}{(K-1)^2} \\
&= 1 + \frac{1}{K-1} \\
&= \frac{K}{K-1}
\end{aligned}$$

Nous pouvons donc simplement multiplier la source définie dans l'équation 3.4 par $\sqrt{\frac{K-1}{K}}$ pour que toutes les règles soient satisfaites, ce qui donne

$$\forall k \in [1, K], \forall i \in [1, K], \bar{x}_{k,i} = \begin{cases} \sqrt{\frac{K-1}{K}}, & i = k \\ -\frac{1}{\sqrt{K(K-1)}}, & i \neq k \end{cases}$$

comme une bonne définition des centres de chaque classe dans la source artificielle \mathbf{X} . Cela donne une source ressemblant à celle-ci

$$\bar{\mathbf{X}} = \begin{pmatrix} \sqrt{\frac{K-1}{K}} & -\frac{1}{\sqrt{K(K-1)}} & \cdots & -\frac{1}{\sqrt{K(K-1)}} & -\frac{1}{\sqrt{K(K-1)}} \\ -\frac{1}{\sqrt{K(K-1)}} & \sqrt{\frac{K-1}{K}} & \cdots & -\frac{1}{\sqrt{K(K-1)}} & -\frac{1}{\sqrt{K(K-1)}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -\frac{1}{\sqrt{K(K-1)}} & -\frac{1}{\sqrt{K(K-1)}} & \cdots & \sqrt{\frac{K-1}{K}} & -\frac{1}{\sqrt{K(K-1)}} \\ -\frac{1}{\sqrt{K(K-1)}} & -\frac{1}{\sqrt{K(K-1)}} & \cdots & -\frac{1}{\sqrt{K(K-1)}} & \sqrt{\frac{K-1}{K}} \end{pmatrix}$$

avec $\bar{\mathbf{X}}$ la source artificielle dite arbitraire.

Cette définition est équivalente à la définition précédente de $[\bar{\mathbf{z}}_{s,k}, k \in [1, K]]$. Maintenant que nous avons la moyenne de chaque classe décrivant notre source artificielle, nous pouvons définir les points résultant de la source artificielle finale \mathbf{X} . Ces points doivent être créés de manière à ce que l'extraction des points d'ancrage $[\mathbf{a}_{s,i}, i \in [1, N_{cluster} * K]]$ se passe bien et aboutisse à des classes séparables. Pour ce faire, nous avons choisi de définir des points d'ancrage le long de dimensions orthogonales entre elles afin que les classes soient facilement séparables. La façon la plus simple est de considérer que chaque classe a sa composante principale le long d'une dimension de l'espace des sources artificielles. Cet espace ayant déjà K dimensions, il est possible de définir les points d'ancrage comme suit

$$\begin{aligned}
&\forall j \in [1, N_{cluster}], \forall k \in [1, K], \\
\mathbf{x}_{j+(k-1)*N_{cluster}} &= \bar{\mathbf{x}}_k + \left(\frac{2(j-1) - (N_{cluster} - 1)}{N_{cluster} - 1} \right) \hat{\mathbf{e}}_k
\end{aligned}$$

avec $\bar{\mathbf{x}}_k$ étant le centre de la classe k définie précédemment et $\hat{\mathbf{e}}_k$ étant le vecteur unitaire le long de la k^{th} dimension. Cette définition crée des points qui sont uniformément distribués entre $\bar{\mathbf{x}}_k - \hat{\mathbf{e}}_k$ et $\bar{\mathbf{x}}_k + \hat{\mathbf{e}}_k$, mais d'autres types de distributions pourraient être utilisés, comme la distribution normale. Cette définition aboutit à une source artificielle de forme $(N_{cluster} * K, K)$ où chaque point de \mathbf{X} correspond à une ancre et leur moyenne se situe au centre $\bar{\mathbf{x}}_k$ de chaque classe $k \in [1, K]$ qui peut donc être extraite au cours de l'ajustement de la méthode TSA. Cette définition mathématique possède de nombreuses propriétés intéressantes, mais ne comporte que K dimensions, ce qui peut s'avérer trop faible pour obtenir de bons résultats dans la classification en ICO, et encore plus lorsque l'on utilise des modèles d'apprentissage profond qui tireront parti de données contenant des informations hautement dimensionnelles. Une façon simple de traiter ce problème est d'ajouter des dimensions de zéros après les K dimensions obligatoires de \mathbf{X} . Bien que cela ajoute un déficit de rang dans la matrice des produits croisés \mathbf{K} et n'augmente pas l'information présente dedans, cela permet aux données alignées de se situer dans un espace de dimension supérieure où ils peuvent garder une plus grande partie de leur information propre lorsqu'elles sont elles-mêmes de grande dimension. Dans cette étude, nous avons ajouté des dimensions afin de nous situer dans un espace de 1000 dimensions, ce qui est déjà un nombre élevé étant donné que nous extrayons des données avec 99,9% de la variance pendant la partie rotation, comme expliqué précédemment, ce qui se traduit généralement par un faible nombre de dimensions.

3.3 Résultats intra- et inter-configurations de la source arbitraire

La source arbitraire a été testée sur 20 ensembles de données divisés en 3 paradigmes étant MI, ERP et SSVEP. Pour ce faire, nous avons utilisé les paradigmes extraits de la bibliothèque Python MOABB ([43]). Étant donné qu'il s'agit d'alignements inter-configurations, nous exigeons que tous les ensembles de données soient comparables et donc possèdent les mêmes classes. Pour le paradigme ERP, les choses sont simples car nous n'avons que des bases d'ERP en P300 où les ERP peuvent généralement être comparés d'un ensemble de données à l'autre. De cette manière, il est possible de prendre presque tous les ensembles de données ERP disponibles sur MOABB, ce qui donne 10 ensembles de données ERP qui sont les suivants

- 003-2015 ([38]),
- 008-2014 ([69]),
- 009-2014 ([4]),
- Brain Invaders 2012 ([82]),
- Brain Invaders 2013a ([81]),
- Brain Invaders 2014a ([50]),
- Brain Invaders 2014b ([51]),
- Brain Invaders 2015a ([48]),

- Brain Invaders 2015b ([49]),
- EPFL P300 dataset ([39]).

Cependant, dans le cas du paradigme MI les classes peuvent être assez diverses. Nous avons choisi le paradigme "main gauche vs main droite", ce qui nous donne 5 ensembles de données, à savoir

- 001-2014 ([79]),
- Cho2017 ([22]),
- Schirrmester2017 ([75]),
- Weibo 2014 ([99]),
- Zhou 2016 ([104]).

En ce qui concerne le paradigme SSVEP, nous avons choisi de comparer des états stables avec des fréquences proches d'une configuration à l'autre, ce qui donne 5 ensembles de données où les fréquences étaient aussi comparables que possible :

- Lee2019_SSVEP ([55]) avec les fréquences 8.57Hz et 12.00Hz,
- SSVEP Exoskeleton ([44]) avec les fréquences 13.00Hz et 17.00Hz car il n'y avait pas de fréquences inférieures,
- SSVEP MAMEM3 ([64]) avec les fréquences 8.57Hz et 12.00Hz,
- SSVEP Nakanishi ([62]) avec des fréquences de 9,25Hz et 11,25Hz,
- SSVEP Wang ([87]) avec les fréquences 8.60Hz et 12.00Hz.

Dans cette étude, pour le sujet cible, nous ne faisons pas de différence entre les données d'entraînement et les données d'alignement car ces données seront utilisées pour l'alignement ou l'entraînement indépendamment (apprentissage par transfert vs train-test). À des fins d'analyse, nous aurions pu choisir de diviser les données d'entraînement et de test avec 50% des essais dans chaque cas, mais étant donné que l'intérêt principal de l'apprentissage par transfert en ICO est de pouvoir obtenir un modèle fonctionnel avec seulement quelques essais disponibles, nous avons ajouté également des configurations de division entraînement-test qui étaient différentes. Nous avons choisi au total trois configurations différentes pour le sujet cible. Il s'agit des suivantes :

- Fractionnement minimal entraînement-test où nous considérons le nombre minimum d'essais pour calculer correctement la TSA. Si les étiquettes sont distribuées uniformément entre les classes, nous avons besoin de $N_{cluster}$ essais pour chaque classe, ce qui signifie que nous avons besoin d'un total de $5 * 2 = 10$ essais pour extraire correctement les caractéristiques nécessaires expliquées dans la partie sur la méthode. C'est le cas pour les paradigmes MI et SSVEP, car les étiquettes des classes sont bien équilibrées. Dans le cas de l'ERP, les étiquettes ne sont généralement pas équilibrées entre la cible et la non cible. Par conséquent, pour conserver cette diversité dans les informations extraites, nous devons modifier le nombre d'essais pris dans l'ensemble d'entraînement. Pour l'ERP, en général, le nombre d'étiquettes non ciblées est toujours un multiple du nombre d'étiquettes ciblées. Par conséquent, comme nous avons besoin d'au moins 5 essais cibles, avec N_T le nombre d'essais cibles et N_N le nombre d'essais non cibles, nous avons besoin d'un total de $(\frac{N_N}{N_T} + 1) * 5$ d'essais. Dans le cas habituel des épileurs P300 où il y a 5 fois plus d'es-

sais non cibles que d'essais cibles, cela donne $(5 + 1) * 5 = 30$ essais pour un entraînement minimal, ce qui correspond à 2,5 lettres orthographiées, ce qui ne prend pas beaucoup de temps.

- Fractionnement moyen entraînement-test où nous prenons 4 fois plus d'essais par rapport à la répartition minimale afin d'avoir une bonne idée de ce que la méthode peut faire avec un nombre raisonnable d'essais. Dans le cas de SSVEP, pour la plupart des ensembles de données, le nombre total d'essais est assez faible et il est donc généralement impossible d'obtenir autant d'essais et d'avoir des essais de test. C'est pourquoi cette répartition n'a pas été calculée dans le cas de ce paradigme.
- Fractionnement classique 50/50 entraînement-test, comme dans de nombreux documents de recherche, ce qui permet de comparer les résultats obtenus par notre méthode à ceux obtenus par d'autres méthodes.

Considérons le sujet cible T sur lequel nous voulons tester les méthodes. Si l'ensemble de données cible est identique à l'ensemble de données source, nous retirons T des sujets source. Par ailleurs il est important de noter que dans l'ensemble de l'étude, le terme "précision" fera toujours référence à la précision équilibrée, sauf indication contraire. Nous avons choisi de comparer 3 méthodes différentes et ce, à la fois pour le TL inter-sujets et le TL inter-configurations, ces méthodes sont les suivantes

- Nous alignons tous les sujets sources sur le sujet présentant la meilleure précision entre l'entraînement et le test, que nous appellerons S , à partir de l'ensemble de données sources et nous entraînons le classificateur sur l'ensemble de données alignées. Nous calculons ensuite l'alignement du sujet cible T sur le sujet S en utilisant uniquement les données d'entraînement. Enfin, nous alignons les données de test sur l'alignement précédemment calculé et nous testons le classificateur sur ces données. Cette méthode est essentiellement la méthode TSA, mais elle crée un grand ensemble de données avec l'ensemble du jeu de données au lieu de l'alignement un sur un entre S et T décrit dans les travaux précédents. Cette méthode est désignée par "TSA" sur les figures.
- La deuxième méthode est proche de la première, mais cette fois S est la source arbitraire décrite précédemment. Tous les autres éléments restent inchangés. Cette méthode est désignée par "Arb source" sur les figures.
- La dernière méthode est un simple train-test qui utilise les données d'entraînement de T pour former un classificateur qui sera ensuite testé sur les données de test. Cette méthode est désignée par "Train-test" sur les figures.

Nous avons résumé les résultats sous forme de diagrammes à barres comparant la précision équilibrée obtenue pour chaque méthode et chaque répartition formation-test.

3.3.1 Résultats intra-configuration

Pour l'apprentissage par transfert intra-configuration, les diagrammes à barres montrent la précision équilibrée moyennée sur les sujets pour chaque ensemble de

données ainsi que la précision équilibrée ensuite moyennée sur les ensembles de données pour respectivement "TSA", "Arb source" et "Train-test". Pour le paradigme P300 présenté en figure 3.1, nous observons que dans la configuration de fractionnement minimal, "TSA" et "Arb source" obtiennent de bien meilleures performances que le "Train-test" avec une différence de 8% en termes de précision (TSA : $59.00 \pm 0.86\%$, Arb source : $58.98 \pm 0.87\%$, Train-test : $51.18 \pm 0.22\%$). Cette différence de performance diminue au fur et à mesure que le nombre d'essais d'entraînement augmente, avec une différence de 5% (TSA : $69,87 \pm 1,64\%$, Arb source : $69,32 \pm 1,67\%$, train-test : $64,11 \pm 1,43\%$) dans la configuration moyenne pour finalement atteindre une différence non significative dans la configuration 50/50, avec une précision équilibrée d'environ 77,5% (TSA : $77.91 \pm 1.37\%$, Arb source : $77.58 \pm 1.46\%$, Train-test : $77.56 \pm 1.42\%$). Cela montre que pour le paradigme P300, les méthodes d'apprentissage par transfert présentée dans cette étude s'avèrent extrêmement utiles pour un étalonnage rapide avec quelques essais.

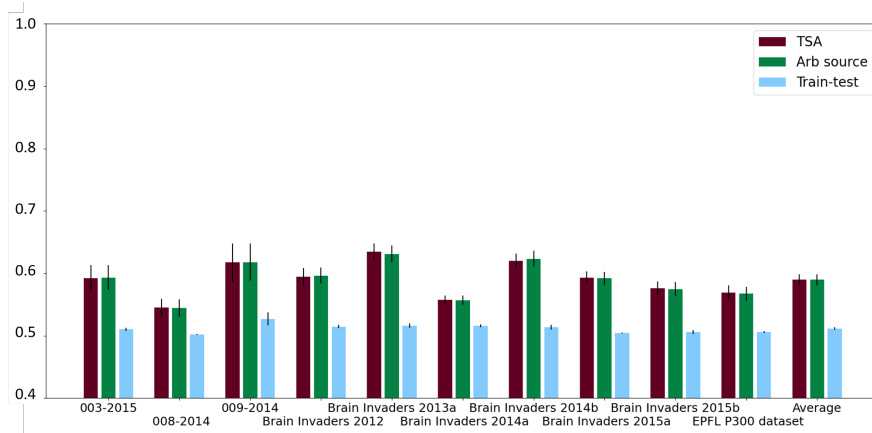
Pour le paradigme d'imagerie motrice présenté en figure 3.2, les résultats sont assez différents. Aucune précision équilibrée significative n'est enregistrée pour les trois conditions de fractionnement différentes. Cela donne en moyenne :

- TSA : $65.32 \pm 3.28\%$, Arb source : $64.02 \pm 3.10\%$, Train-test : $65.97 \pm 3.28\%$ avec fractionnement minimal,
- TSA : $71.99 \pm 3.20\%$, Arb source : $70.45 \pm 3.32\%$, Train-test : $72.71 \pm 3.38\%$ avec fractionnement moyen,
- TSA : $74.39 \pm 3.19\%$, Arb source : $73.34 \pm 3.36\%$, Train-test : $75.96 \pm 3.65\%$ avec fractionnement 50/50.

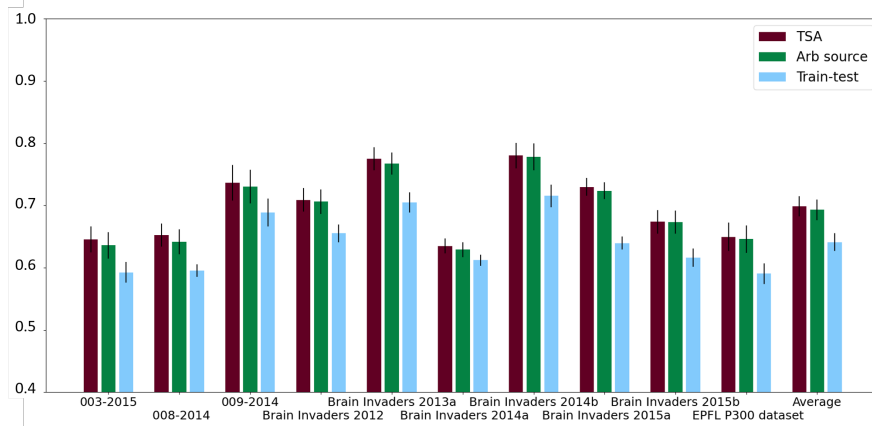
Ces résultats sont conformes à ceux obtenus dans [15] où le paradigme de l'imagerie motrice ne montre pas d'augmentation significative de la précision par rapport au train-test et étendent ces résultats à l'utilisation de la source arbitraire sans aucun changement significatif par rapport à TSA.

Dans le même ordre d'idées, pour le paradigme SSVEP présenté en figure 3.3, tant pour la configuration minimale que pour la configuration 50/50 (la configuration moyenne a été exclue comme expliqué précédemment), les résultats continuent de s'aligner sur les travaux précédents et ne montrent pas non plus de perte significative d'informations en utilisant la méthode de la source arbitraire.

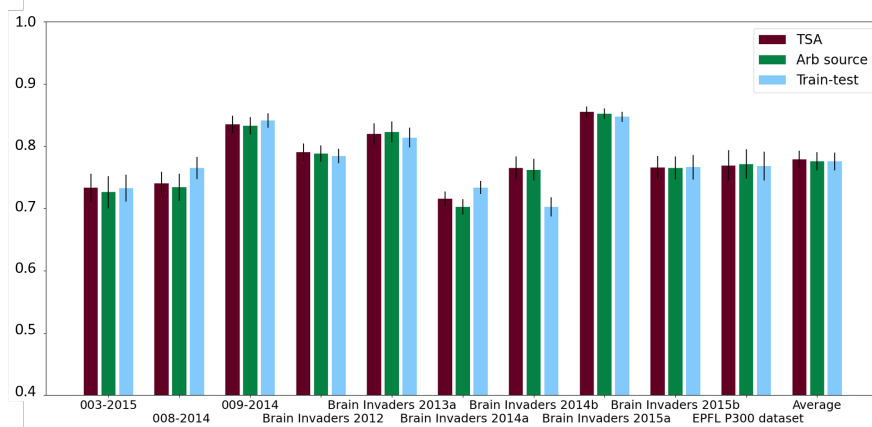
Tous ces résultats montrent qu'il est tout à fait possible d'utiliser une source arbitraire bien définie mathématiquement afin d'aligner des données sur d'autres ayant préalablement entraîné un modèle et d'obtenir de très bons résultats, parfois même meilleurs que la méthode de train-test, typiquement pour l'ICO P300. Cependant, les résultats ne se limitent pas au TL intra-configuration.



(a) avec fractionnement minimal

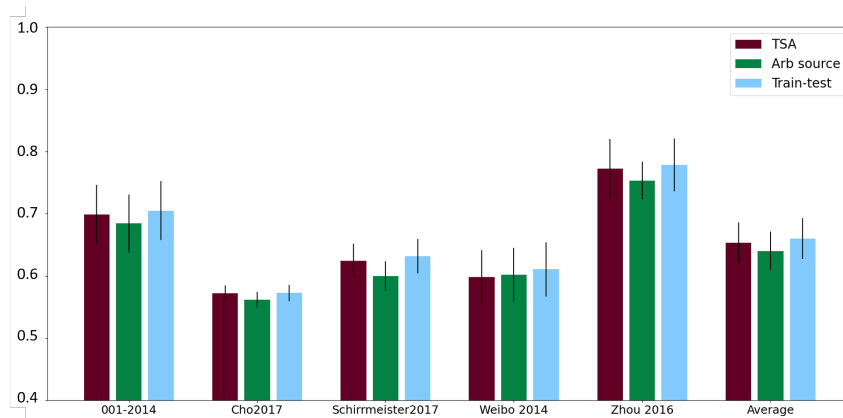


(b) avec fractionnement moyen

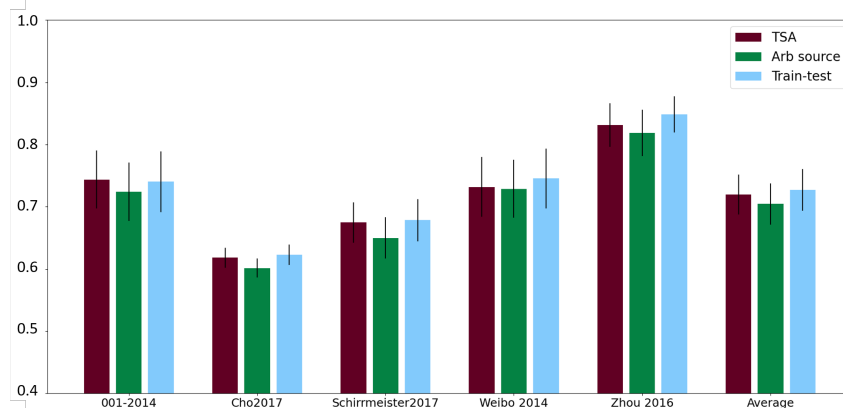


(c) avec fractionnement 50/50

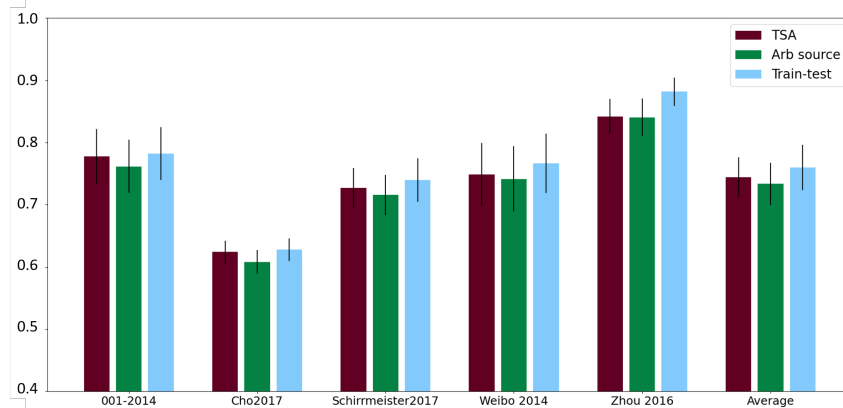
FIGURE 3.1 – Précision équilibrée intra-configuration pour le paradigme P300



(a) avec fractionnement minimal

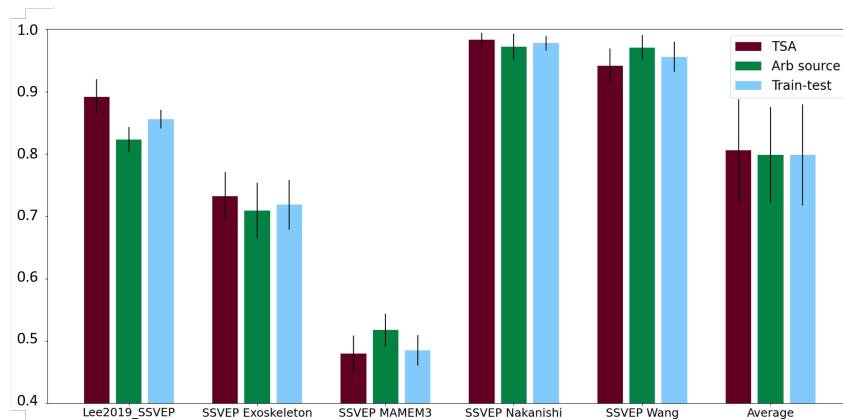


(b) avec fractionnement moyen

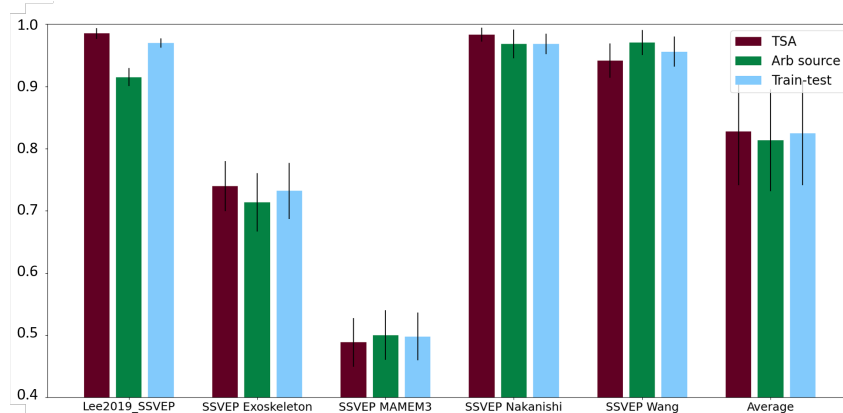


(c) avec fractionnement 50/50

FIGURE 3.2 – Précision équilibrée intra-configuration pour le paradigme MI.



(a) avec fractionnement minimal



(c) avec fractionnement 50/50

FIGURE 3.3 – Précision équilibrée intra-configuration pour le paradigme SSVEP.

3.3.2 Résultats inter-configurations

La présente méthodologie, bien que particulièrement pertinente dans le contexte de l'apprentissage par transfert intra-configuration, possède une applicabilité plus large. Plus précisément, l'objectif premier de la méthode des sources arbitraires est de fournir un moyen efficace d'aligner les données obtenues à partir de dispositifs expérimentaux disparates. À cet égard, le processus d'alignement des données décrit précédemment a été appliqué à des données provenant d'ensembles de données distincts. Les données résultantes ont ensuite été formatées d'une manière équivalente à celle employée dans l'analyse de l'apprentissage par transfert intra-configuration, générant trois ensembles de diagrammes à barres correspondant aux trois configurations uniques de fractionnement des données pour chacune des bases utilisées comme cible. Il convient de noter que comme l'ensemble de données cible ne pouvait pas servir d'ensemble de données source

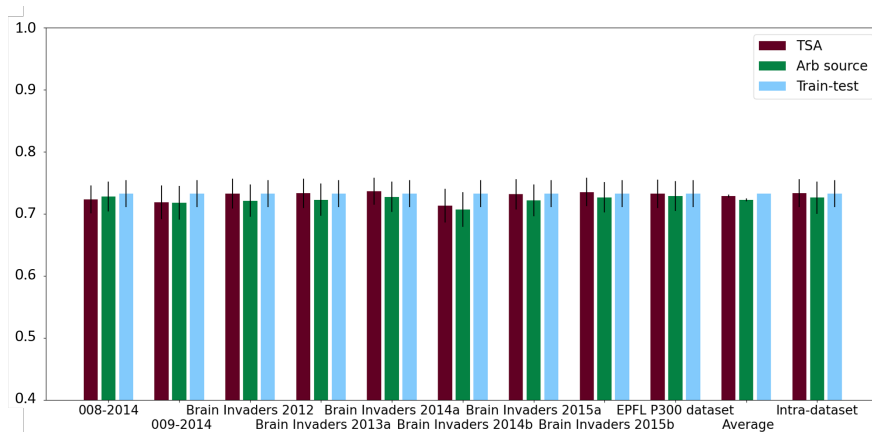


FIGURE 3.4 – Précision équilibrée inter-configurations pour la base de donnée 003-2015 avec un fractionnement 50/50. L’abscisse correspond aux bases de données source ainsi que "Average" pour la moyenne de toutes les bases sources et "Intra-dataset" pour les résultats intra-configuration présentés plus tôt et permettant la comparaison directe.

dans le contexte de l’analyse inter-configurations, les précisions équilibrées intra-configuration calculées précédemment ont été incluses comme moyen de comparaison avec les résultats croisés. Au total, 60 diagrammes à barres ont été générés, mais pour des raisons évidentes la très large majorité de ces données seront présentées en Annexe et seule une petite partie sera directement présentée ici. L’analyse globale est cependant réalisée sur la totalité des bases.

Après analyse de la grande quantité de données générées, une tendance claire s’est dégagée, indiquant un faible niveau de dépendance à l’égard de l’ensemble de données source utilisé. Les figures 3.4, 3.5 et 3.6 en donnent un bon exemple. En effet on remarque que pour la même cible (par exemple la base 003-2015), les précisions équilibrées sont très peu variable étant donnée l’ensemble de données source. Cette observation se vérifie par le fait que la moyenne globale inter-configurations possède une très faible erreurs standard pour TSA et la source arbitraire. Pour information, comme la partie train-test est toujours la même et indépendant de la source, l’erreur type de la moyenne est nulle. Il est important de noter que les barres "Average" et les erreurs standard correspondantes n’ont pas pour but de donner une signification statistique et sont utilisées pour quantifier la variabilité des résultats du inter-configurations.

Les résultats présentés ici s’appuient sur ceux détaillés dans [15], qui ont démontré que l’ensemble de données source utilisé avait un impact minimal sur la précision finale de l’ensemble de données cible dans le contexte de la définition arbitraire de la source et de l’apprentissage par transfert entre ensembles. Cependant, bien que la dépendance observée soit globalement faible, plusieurs tendances notables ont été identifiées. Notamment, l’ensemble de don-

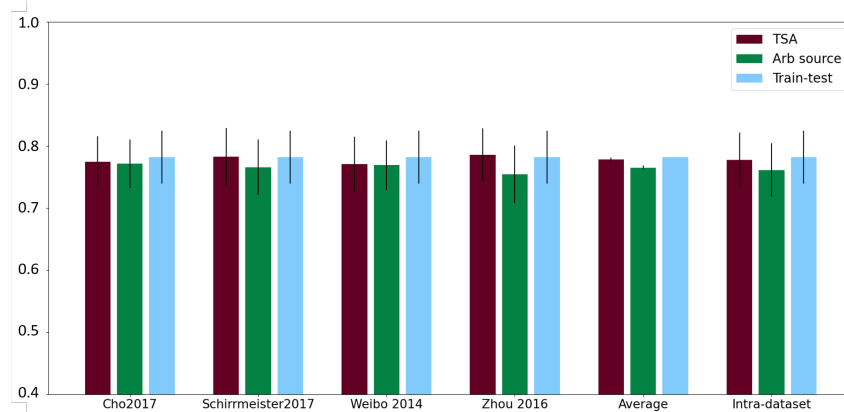


FIGURE 3.5 – Précision équilibrée inter-configurations pour la base de donnée 001-2014 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

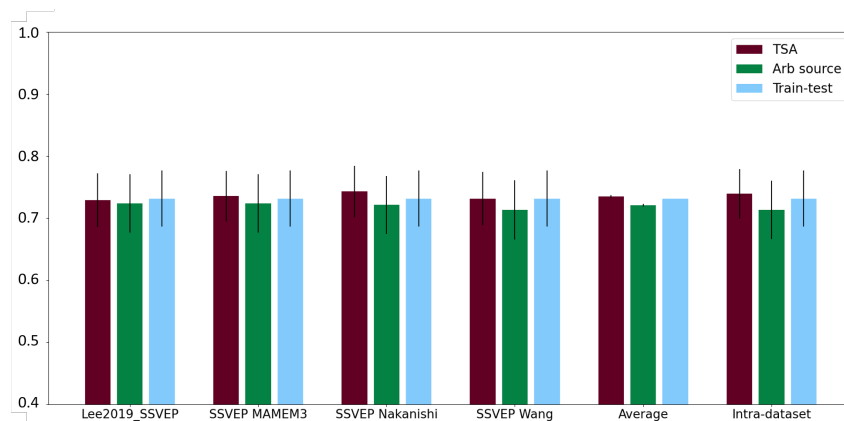
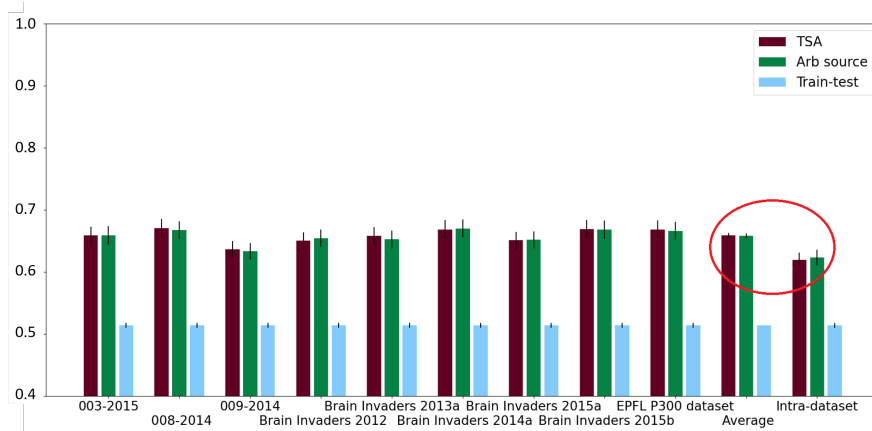
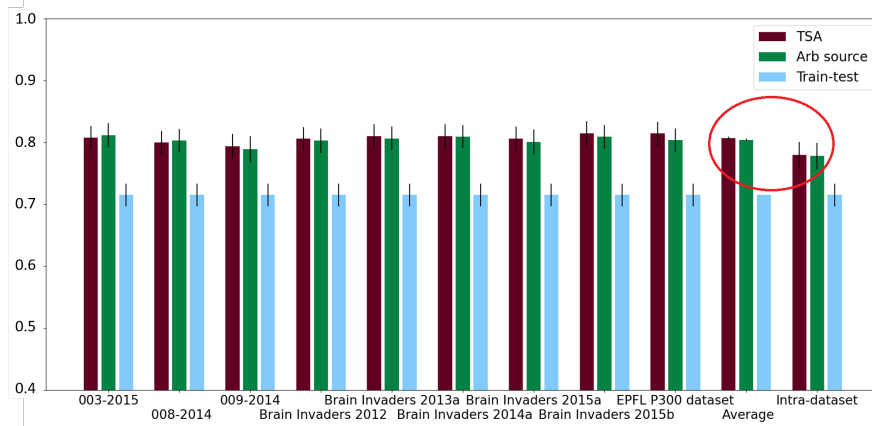


FIGURE 3.6 – Précision équilibrée inter-configurations pour la base de donnée SSVEP Exoskeleton avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

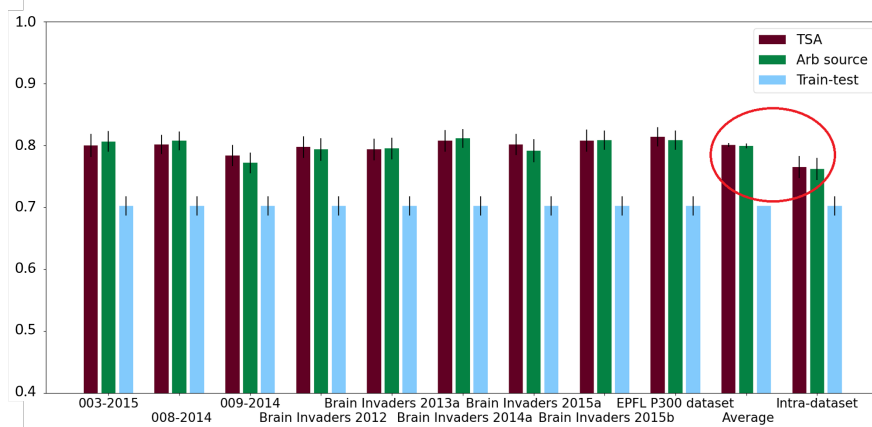
nées Brain Invaders 2014a (bi2014a) a systématiquement servi de source légèrement supérieure par rapport aux autres ensembles de données, tandis que l'ensemble de données Brain Invaders 2014b (bi2014b) a systématiquement été identifié comme une source sous-optimale. Ces tendances sont particulièrement intéressantes étant donné que la précision équilibrée de ces ensembles de données était comparable dans la configuration de partage 50/50 dans l'analyse d'apprentissage par transfert intra-configuration, et que bi2014b a démontré une meilleure précision moyenne que bi2014a dans la configuration de partage moyen. Ces résultats suggèrent que la précision d'un ensemble de données ne dicte pas nécessairement son efficacité en tant qu'outil d'entraînement pour un modèle donné. En outre, ces tendances étaient également évidentes lorsque l'on compare l'apprentissage par transfert inter-configurations et intra-configuration, produisant des résultats parfois contre-intuitifs, comme dans le cas de bi2014b, où même lors de l'analyse d'un sujet cible de cet ensemble de données, les autres ensembles de données ont constamment surpassé l'apprentissage par transfert intra-configuration, en dépit de leur composition commune. Cette observation est étayée par les données présentées en figure 3.7, où les barres associées aux autres ensembles de données sont nettement plus élevées que celles associées à l'apprentissage par transfert "Intra-dataset". Numériquement, cela donne une différence de précision équilibrée d'environ 4% pour la configuration minimale, d'environ 2% pour la configuration moyenne et de plus de 3% pour la configuration 50/50, comme le montre le tableau 3.1. Cela pourrait s'interpréter comme le fait que la base de données bi2014b est une source de mauvaise qualité, au point que même pour les sujet de sa propre base, apprendre sur les sujet d'une autre base est plus efficace que sur celle-ci. Il reste pourtant notable que la base de donnée en elle-même possède une précision en train-test très correcte, allant dans le sens de la théorie selon laquelle le fait d'être une mauvaise source est liée à d'autres paramètres.



(a) avec fractionnement minimal



(b) avec fractionnement moyen



(c) avec fractionnement 50/50

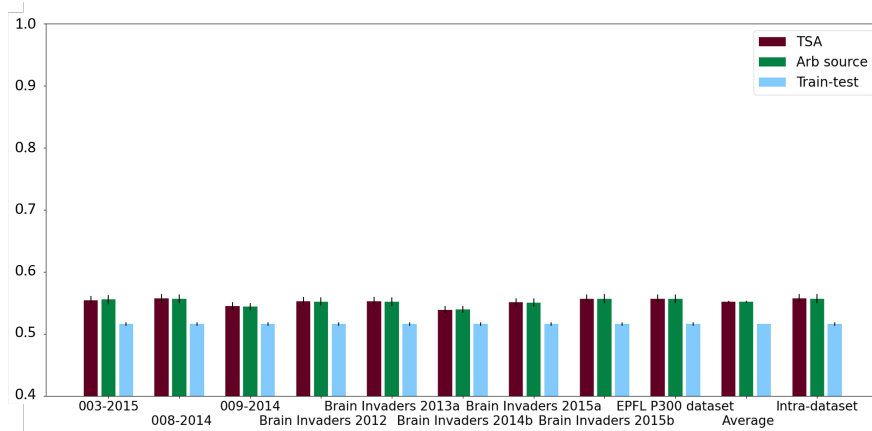
FIGURE 3.7 – Précision équilibrée inter-configurations pour la base de donnée bi2014b. Les résultats sont présentées comme en figure 3.4.

	TSA	Arb source
Inter-conf minimal	$65.92 \pm 0.36\%$	$65.83 \pm 0.37\%$
Intra minimal	$61.99 \pm 1.17\%$	$62.34 \pm 1.25\%$
Inter-conf moyen	$80.72 \pm 0.21\%$	$80.43 \pm 0.21\%$
Intra moyen	$78.02 \pm 2.06\%$	$77.82 \pm 2.15\%$
Inter-conf 50/50	$80.10 \pm 0.28\%$	$79.97 \pm 0.40\%$
Intra 50/50	$76.54 \pm 1.82\%$	$76.23 \pm 1.77\%$

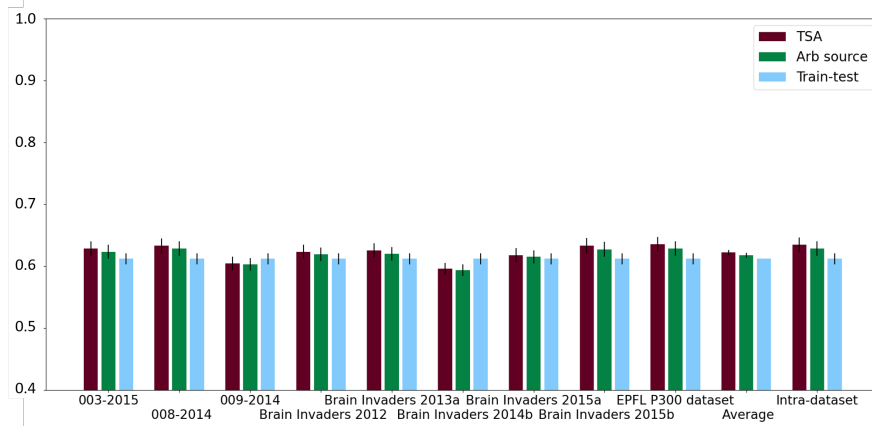
TABLE 3.1 – Tableau comparant la précision équilibrée moyenne inter-configurations et intra-configuration pour la base de données bi2014b.

Dans le cas de bi2014a, qui est généralement une source légèrement meilleure que les autres, c'est le contraire qui se produit comme on peut l'observer en figure 3.8. En effet, on observe que la moyenne des autres bases va se situer légèrement en dessous de l'intra-configuration. Ces résultats sembleraient démontrer que malgré la faible dépendance de la base source dans l'utilisation de cette méthode d'apprentissage par transfert, il reste tout de même possible de faire varier un peu les résultats finaux en utilisant des données source possédant de bonnes propriétés. Il est d'ailleurs très intéressant de noter que "possédant de bonnes propriétés" ne signifie pas nécessairement avoir une bonne précision équilibrée en tant que cible ou même simplement en train-test. C'est d'ailleurs le cas de bi2014a qui possède une précision équilibrée plutôt faible mais qui montre d'étonnamment bons résultats lorsqu'elle est utilisée en tant que source. Une étude poussée de ces bases pourrait sûrement permettre de déterminer ce qui fait que bi2014b est une mauvaise source quand à l'inverse bi2014a en est une bonne. Malheureusement cette étude n'a pas pu être menée au cours de cette thèse et cette question reste ouverte pour de futures recherches sur le sujet.

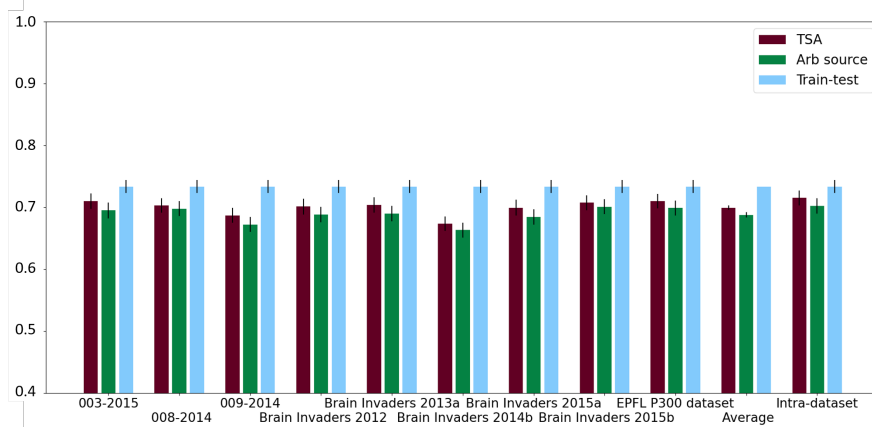
Les résultats présentés dans cette étude revêtent une importance considérable, car ils démontrent la faisabilité de l'utilisation d'un modèle entraîné pendant des heures, voire des jours, sur des données antérieures collectées à l'aide d'une configuration totalement différente, pour l'analyse de nouvelles données à la suite d'un processus d'alignement rapide, même dans des scénarios avec une disponibilité limitée des nouvelles données. Il convient de noter qu'un modèle de régression logistique a été utilisé comme classificateur dans cette étude, étant donné sa réputation de rapidité d'apprentissage, ce qui en fait un choix approprié pour cette analyse approfondie, qui a été limitée par des considérations de temps découlant du grand nombre de bases de données soumises au traitement.



(a) avec fractionnement minimal



(b) avec fractionnement moyen



(c) avec fractionnement 50/50

FIGURE 3.8 – Précision équilibrée inter-configurations pour la base de donnée bi2014a. Les résultats sont présentées comme en figure 3.4.

3.3.3 Discussion

L'utilisation de sources arbitraires dans le cadre de l'apprentissage par transfert pour les applications d'ICO a démontré une efficacité remarquable, permettant un potentiel changement significatif par rapport à l'approche conventionnelle du TL en ICO. Le déploiement de sources arbitraires élimine la nécessité de télécharger de vastes ensembles de données, offrant ainsi une nouvelle perspective intéressante. En effet, les résultats obtenus dans le cadre de cette étude suggèrent que les futurs modèles de décodeurs en ICO pourraient être construits avec un modèle pré-entraîné téléchargeable et une source arbitraire, qui peut être utilisée pour aligner de nouvelles données. Une telle configuration offre une solution très efficace et rentable pour la mise en œuvre d'applications en ICO qui s'appuieraient alors sur un apprentissage par transfert très largement fait hors ligne et potentiellement sur des machines différentes, ou par des groupes de recherche différents. Ces résultats sont très prometteurs pour l'avancement du domaine de l'ICO et ouvrent de nouvelles voies pour l'exploration des méthodologies basées sur le TL.

Il convient de noter que, dans le cadre de cette étude, la méthodologie Train-Test n'a utilisé que les échantillons utilisés pour l'alignement. Par conséquent, la quantité d'informations pouvant être obtenue par cette approche était intrinsèquement limitée par rapport aux techniques d'apprentissage par transfert qui intègrent à la fois l'alignement et les données sources, constituant généralement un ensemble de données plus important et plus diversifié malgré les problèmes d'hétérogénéité potentiels. Il convient toutefois de mentionner qu'étant donné l'utilisation d'un classifieur par régression logistique dans cette étude, la diversité d'informations pouvant être extraite des données était très limitée par rapport aux modèles à forte intensité de calcul tels que les réseaux neuronaux profonds qui auraient pu être utilisés, car un seul ensemble de données aligné aurait pu fournir suffisamment de données pour adapter correctement ces modèles.

Finalement, cette étude a également permis de faire ressortir des schémas intéressants, bien que pas nécessairement nouveaux, sur le fait que les critères d'utilisation d'une base ou d'un sujet comme source en apprentissage par transfert sont plus complexes que simplement sa précision, que ce soit en train-test ou même en apprentissage par transfert intra-configuration. Il reste cependant notable qu'une bonne source a tendance à agir comme une bonne source globalement et non pour des bases spécifique. Il semblerait donc que les critères soient intrinsèquement liés aux données de la source plus même qu'à sa compatibilité à la cible. Des recherches plus poussées sont cependant nécessaire pour en trouver l'origine exacte.

Chapitre 4

Conclusions et perspectives

Les interfaces cerveau-ordinateurs sont un sujet de recherche qui passionne de plus en plus la communauté scientifique, que ce soit pour les promesses qu'elles portent mais également par l'intérêt purement scientifique des défis à relever les concernant. En ce sens, l'apprentissage par transfert en ICO est un sujet très vaste et auquel la communauté scientifique commence tout juste à s'intéresser. Cependant, avec l'abondance de données déjà existantes et à venir, il est évident que ce sujet de recherche sera aux avant-garde des progrès à venir en ICO.

Au cours de cette thèse plusieurs méthodes ont été développées dans le but d'améliorer le transfert d'information en inter- et intra-sujets pour les ICOs. Dans cette conclusion nous allons revenir sur leur apport à la communauté scientifique et poursuivre sur les ouvertures qu'elles laissent pour des recherches futures dans le domaine.

4.1 Le TSA et l'apprentissage par transfert

La partie la plus aboutie de cette thèse est sans aucun doute la méthode de Tangent Space Alignment (TSA) dont la mise en place et l'amélioration continue ont été décrites lors du Chapitre 2. Cette méthode permet d'aligner les données d'un sujet à l'autre en minimisant la perte d'information. Elle tire parti des bonnes propriétés de la géométrie riemannienne pour les ICO tout en ayant la possibilité de se placer dans le cadre euclidien de l'espace tangent, la rendant non seulement plus efficace, que ce soit en terme de résultats ou de temps de calcul, mais également plus facile à comprendre et permettant à ses résultats d'être analysés plus simplement. L'algorithme qui en découle a pu être testé de façon exhaustive sur un grand nombre de données et a montré sa capacité à s'adapter à tous les paradigmes, tout en montrant des résultats particulièrement bons dans le cadre des ERP en P300.

Cette méthode a de plus la possibilité d'être améliorée de façon simple en ajoutant de nouveaux points d'ancrage, comme cela a pu être expliqué en détails en section 2.2. La simplicité de l'ajout de nouveaux vecteurs de caractéristiques

donne beaucoup de potentiel à la méthode et permet d'en faire un modèle très attrayant pour quiconque souhaiterait se l'approprier et rendre ce modèle plus performant. Plusieurs tentatives d'ajout de nouveaux vecteurs de caractéristiques potentiellement intéressants ont pu être essayés au cours de cette thèse et ont été détaillés dans cette même section 2.2. Ces idées ont également pour but de donner des pistes menant à l'amélioration du modèle dans les futures années.

Le TSA a également démontré une dépendance très faible à la source. Ainsi, même s'il est possible de gagner quelques pourcentages avec des données sources particulièrement bonnes, les résultats vont très largement dépendre de la qualité des données cibles. Cette indépendance est quelque chose de très intéressant dans le cadre des ICO où il est souhaitable de dépendre de sa propre configuration et de ses propres données plutôt que de la cohérence entre celles-ci et des données précédentes parfois anciennes ou peu connues. Retirer de l'équation les questionnements liés à la source est donc une caractéristique particulièrement attrayante.

Cependant, bien que cette méthode ait été testée sur un grand nombre de données, les algorithmes de classification utilisés au cours de cette thèse (SVM, LR ...) ne tiraient qu'un avantage léger de l'abondance de données avec lesquelles ils pouvaient être entraînés. Il aurait été très intéressant de pouvoir analyser des résultats obtenus à l'aide de réseaux de neurones un minimum profonds, mais avons préféré prendre du temps pour aller vers un autre sujet plutôt que faire le choix d'apprendre à utiliser des réseaux de neurones pour tester la méthode. Cela laisse donc le champ libre à toute personne souhaitant reprendre cette recherche et approfondir ces résultats. Pour cela nous feront en sorte de mettre la méthode en accès libre et facile d'utilisation¹.

Par ailleurs cette méthode cible un transfert d'informations dans le cadre d'un nombre potentiellement important de classes or bien que des tests effectués avec quatre classes semblent montrer qu'elle fonctionne tout aussi bien dans ce cas, les résultats présentés au cours de cette thèse n'en ont jamais fait part et se résument à deux classes seulement. Ainsi son efficacité réelle reste à être démontrée avec plus de deux classes.

4.2 La source arbitraire et le transfert d'information entre bases de données hétérogènes

Dans le Chapitre 3 nous nous sommes tournés vers l'étude d'un domaine peu traité du TL en ICO, à savoir le TL hétérogène inter-configurations et nous nous sommes intéressés plus précisément à l'étude de l'apport d'une source arbitraire mathématique dans ce cadre. Cette source arbitraire mathématique a été conçue de telle sorte à posséder de bonnes propriétés et séparer correctement les classes des cibles s'alignant dessus. Elle a ainsi montré sa capacité à conser-

1. Et si la méthode n'est pas directement disponible dans des bibliothèques Python dédiés telles que MOABB ou Pyriemann, le code source sera mis en accès ouvert sur GitHub.

ver l'information des données cibles et cela en gardant une grande simplicité mathématique.

Par ailleurs l'apport d'une source arbitraire mathématique permet d'imaginer une utilisation des outils de TL avec plus de sens et une compréhension plus importante des propriétés recherchées pour la source et la cible. En effet, aujourd'hui l'utilisation d'un sujet source en particulier est orienté par ses résultats sur des données connues plutôt que par les propriétés que ses données possèdent, ce qui peut parfois mener à des biais qui ne sont pas souhaitables en général. Malgré tout, le TL hétérogène appliqué à l'aide de la méthode TSA s'est également montré très efficace sans source arbitraire mathématique, montrant que cette méthode arrive à transférer les informations importantes même entre bases de données hétérogènes.

Cette recherche autour de la source arbitraire n'a pas la présomption de donner une solution définitive et immuable de la source arbitraire en tant que telle mais plutôt d'ouvrir une nouvelle porte vers une façon originale de faire du TL dans le cadre des ICO, en se tournant d'avantage sur la compréhension des données utilisées et des propriétés que l'on recherche dans ces données. De cette manière, la proposition analytique de source arbitraire qui a été présentée dans cette thèse peut sembler simpliste mathématiquement mais se place comme une base sur laquelle il est possible de construire dans le futur. De plus, bien que donnant une solution multi-classes de sa forme, l'alignement multi-classe n'a pas été testé et bien que les résultats à deux classes montrent que cette source arbitraire possède un potentiel, il n'est pas certain que cela se généralise dans les futures recherches. Il reste également beaucoup de travail sur l'étude des données cibles que l'on souhaite aligner. En effet, la source arbitraire actuellement définie reprend des propriétés connues ou parfois juste souhaitées des données mais ne tient absolument pas compte des données elles-mêmes. Ainsi il est fort probable qu'une source mathématiques possédant certaines propriétés additionnelles soit parfaite pour aligner des données P300 mais ne se prête absolument pas à l'alignement de données de MI. Il serait alors passionnant de regarder de plus près les nuages de points obtenus dans des paradigmes particuliers et tenter de trouver certaines généralités pour créer de nouvelles sources mathématiques plus proches des données cibles mais sans tomber dans le piège du surajustement de la source faisant à terme perdre en généralité.

4.3 Perspectives futures

De cette conclusion ressortent deux grandes perspectives futures liées à l'apport des travaux de cette thèse. Tout d'abord la méthode TSA a montré son efficacité pour transférer des informations entre la source et la cible mais n'a pas pu être testée avec des méthodes de classification tirant parti du grand nombre de données que cette méthode peut permettre d'aligner pour l'apprentissage. Il serait donc particulièrement intéressant de comparer les résultats obtenus avec des algorithmes profonds.

Ensuite comme nous avons pu en discuter lors de la conclusion sur la source

arbitraire, celle-ci a pour but de pouvoir aligner un grand nombre de données hétérogènes mais également d'ouvrir la porte à une recherche plus exploratoire sur les propriétés d'une bonne source ou d'une bonne cible, ce qui pourrait permettre d'aller vers une recherche qui se porte sur la qualité des données plutôt que leur quantité. En effet, aujourd'hui la quantité de données n'est plus un réel problème et le sera de moins en moins dans le futur notamment grâce à la mise en place de bibliothèques telles que MOABB, et avec la potentielle ouverture de ce type de système au grand public, pouvoir juger de la qualité des données et de leur capacité à servir dans la construction de sources va devenir un enjeu de taille. En effet, rien qu'en gardant les données en local, pouvoir utiliser du TL intra-sujet d'une session à l'autre peut permettre une meilleure robustesse des algorithmes. Par ailleurs l'idée de l'utilisation des données des utilisateurs pour créer des modèles globaux très robustes deviendrait aussi attrayante pour les grandes entreprises s'intéressant au sujet, ce qui ajoute une dimension sociétale sur la préservation des données personnelles à ne pas laisser de côté. Ensuite la mise en place de source arbitraire lors du TL ouvre la voie à des modèles d'apprentissage profond entraînés sur des données alignées pendant de longues heures voire plusieurs jours dont les poids et l'architecture seraient partagés en ligne ainsi que le modèle mathématique de la source arbitraire ayant servi à l'alignement des données. Avec ces informations, il devient alors possible de reprendre le même modèle et d'aligner les données de n'importe quel nouveau sujet avec n'importe quelle configuration sur la source arbitraire pour utiliser le modèle profond sans avoir besoin de l'entraîner à nouveau. On pourrait alors entrevoir une utilisation plus simple et "plug and play" des ICO dont les algorithmes peuvent juger rapidement de la qualité des données et guider l'utilisateur dans son utilisation de l'outil. Ce type de progrès serait tout particulièrement souhaitable pour le domaine de la santé dans lequel les ICO commencent à se généraliser. Réduire la calibration des systèmes pourrait permettre d'améliorer grandement la qualité de vie des patients prenant part aux essais cliniques et d'améliorer l'impact bénéfique des ICOs sur leur vie.

Annexe A

Résultats supplémentaires

Table des matières

A.1 Résultats détaillés du Chapitre 2	107
Database 003-2015 (ERP)	108
Database 008-2014 (ERP)	110
Database 009-2014 (ERP)	112
Database Brain Invaders 2013a (ERP)	114
Database EPFL P300 (ERP)	117
Database 001-2014 (MI)	119
Database 001-2015 (MI)	121
Database 002-2014 (MI)	123
Database 004-2014 (MI)	126
Database Alexandre Motor Imagery (MI)	128
Database Cho2017 (MI)	130
Database Grosse-Wentrup 2009 (MI)	133
Database Physionet Motor Imagery (MI)	135
Database Weibo 2014 (MI)	138
Database Zhou 2016 (MI)	140
Database SSVEP Exoskeleton (SSVEP)	142
Database SSVEP Nakanishi (SSVEP)	144
Database SSVEP Wang (SSVEP)	146
A.2 Résultats détaillés du Chapitre 3	148
Base de données 003-2015 (ERP)	149

Base de données 008-2014 (ERP)	151
Base de données 009-2014 (ERP)	153
Base de données Brain Invaders 2012 (ERP)	155
Base de données Brain Invaders 2013a (ERP)	157
Base de données Brain Invaders 2014a (ERP)	159
Base de données Brain Invaders 2014b (ERP)	161
Base de données Brain Invaders 2015a (ERP)	163
Base de données Brain Invaders 2015b (ERP)	165
Base de données EPFL P300 dataset (ERP)	167
Base de données 001-2014 (MI)	169
Base de données Cho2017 (MI)	171
Base de données Schirrneister2017 (MI)	173
Base de données Weibo 2014 (MI)	175
Base de données Zhou 2016 (MI)	177
Base de données Lee2019_SSVEP (SSVEP)	179
Base de données SSVEP Exoskeleton (SSVEP)	180
Base de données SSVEP MAMEM3 (SSVEP)	181
Base de données SSVEP Nakanishi (SSVEP)	182
Base de données SSVEP Wang (SSVEP)	183

A.1 Résultats détaillés du Chapitre 2

Database 003-2015 (ERP)

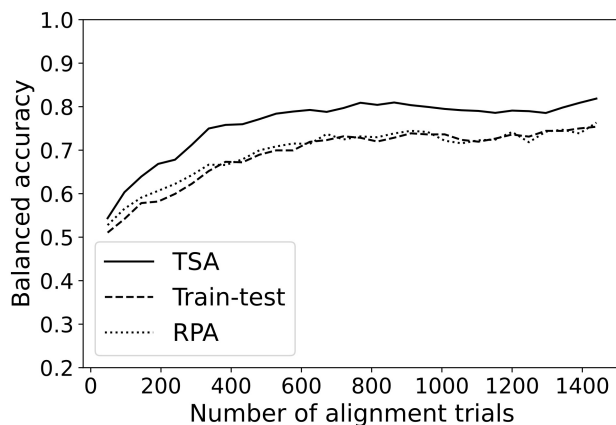


FIGURE A.1 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données 003-2015.

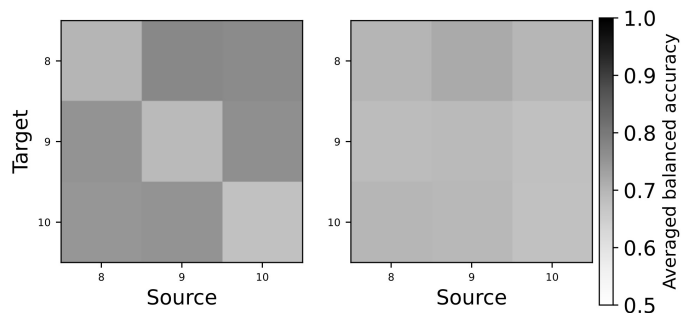


FIGURE A.2 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données 003-2015.

TABLE A.1 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données 003-2015.

	Train-test acc	TSA acc	RPA acc
Subject 8	69.99 ± 1.06	77.14 ± 0.56	70.83 ± 0.56
Subject 9	69.30 ± 1.70	75.85 ± 1.46	68.44 ± 1.14
Subject 10	67.84 ± 0.99	75.16 ± 0.70	69.63 ± 0.73

TABLE A.2 – p-value par sujet pour la base de données 003-2015.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 8	0.047	0.979	0.213
Subject 9	0.011*	0.957	0.764
Subject 10	0.022	0.986	0.791

Database 008-2014 (ERP)

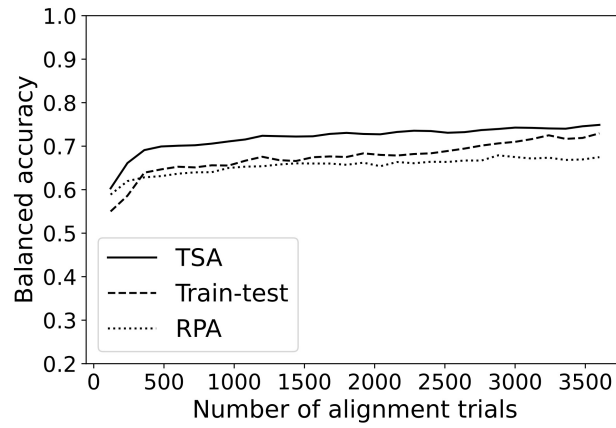


FIGURE A.3 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données 008-2014.

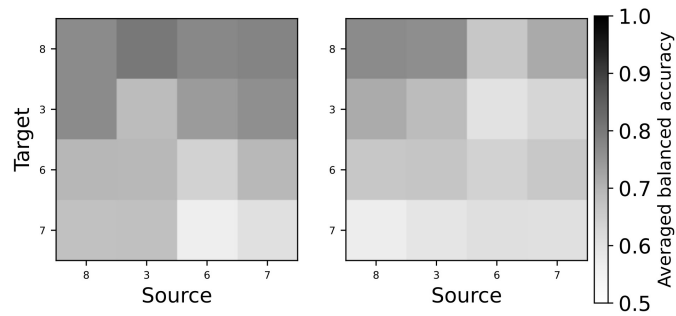


FIGURE A.4 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données 008-2014.

TABLE A.3 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données 008-2014.

	Train-test acc	TSA acc	RPA acc
Subject 3	68.87 \pm 0.82	75.70 \pm 0.50	64.96 \pm 0.61
Subject 6	64.02 \pm 0.58	69.52 \pm 0.28	66.56 \pm 0.21
Subject 7	60.37 \pm 0.60	64.23 \pm 0.58	58.96 \pm 0.31
Subject 8	76.81 \pm 1.12	78.54 \pm 0.49	71.54 \pm 0.56

TABLE A.4 – p-value par sujet pour la base de données 008-2014.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 3	0.060	0.781	0.401
Subject 6	0.169	0.758	0.264
Subject 7	0.060	0.984	0.943
Subject 8	0.054	0.096	0.055

Database 009-2014 (ERP)

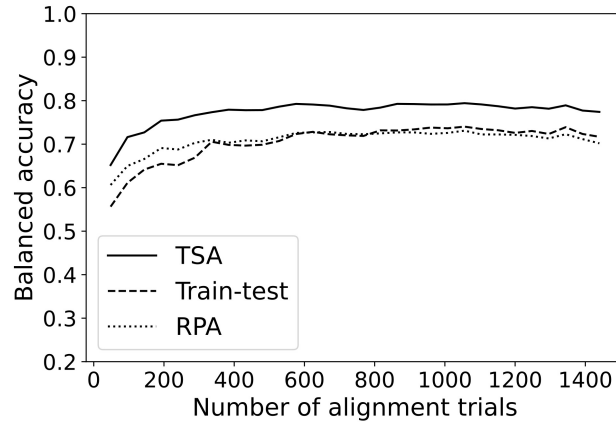


FIGURE A.5 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données 009-2014.

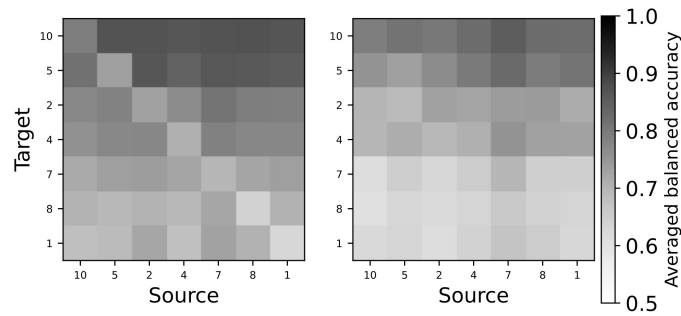


FIGURE A.6 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données 009-2014.

TABLE A.5 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données 009-2014.

	Train-test acc	TSA acc	RPA acc
Subject 1	62.86 \pm 0.75	70.19 \pm 0.42	63.91 \pm 0.33
Subject 2	73.24 \pm 1.48	78.63 \pm 0.44	71.90 \pm 0.46
Subject 4	70.71 \pm 1.04	77.45 \pm 0.29	72.00 \pm 0.33
Subject 5	73.57 \pm 0.57	85.05 \pm 0.16	79.23 \pm 0.27
Subject 7	69.83 \pm 0.85	72.94 \pm 0.31	63.98 \pm 0.28
Subject 8	64.01 \pm 0.63	70.33 \pm 0.27	62.99 \pm 0.27
Subject 10	79.13 \pm 0.95	87.10 \pm 0.07	82.31 \pm 0.22

TABLE A.6 – p-value par sujet pour la base de données 009-2014.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	<0.001*	0.999	0.985
Subject 2	<0.001*	0.911	0.114
Subject 4	<0.001*	0.931	0.564
Subject 5	<0.001*	0.775	0.075
Subject 7	0.001*	0.993	0.918
Subject 8	<0.001*	0.998	0.986
Subject 10	<0.001*	0.029*	0.002*

Database Brain Invaders 2013a (ERP)

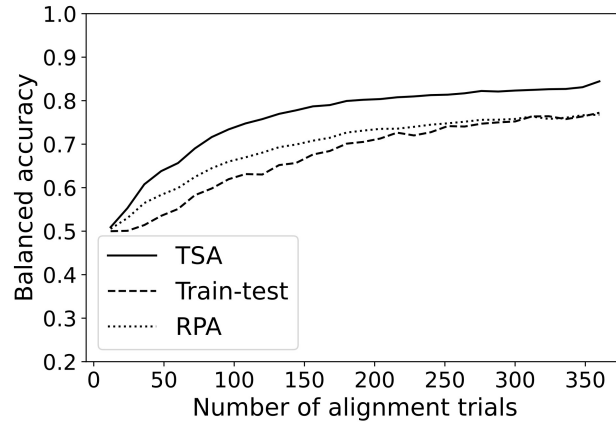


FIGURE A.7 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données Brain Invaders 2013a.

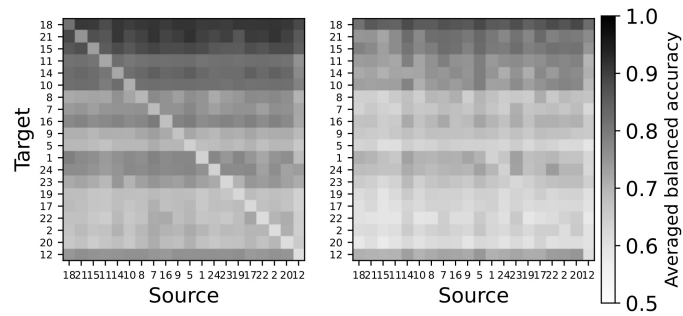


FIGURE A.8 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données Brain Invaders 2013a.

TABLE A.7 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données Brain Invaders 2013a.

	Train-test acc	TSA acc	RPA acc
Subject 1	63.99 \pm 1.01	75.97 \pm 0.35	69.30 \pm 0.35
Subject 2	61.48 \pm 1.42	69.03 \pm 0.39	63.29 \pm 0.31
Subject 5	64.41 \pm 1.86	69.13 \pm 0.46	63.33 \pm 0.36
Subject 7	68.80 \pm 2.04	75.06 \pm 0.38	66.65 \pm 0.34
Subject 8	69.60 \pm 2.06	74.39 \pm 0.33	66.92 \pm 0.30
Subject 9	67.32 \pm 1.79	71.06 \pm 0.41	66.97 \pm 0.38
Subject 10	71.09 \pm 1.57	81.57 \pm 0.38	74.95 \pm 0.35
Subject 11	72.14 \pm 2.63	81.30 \pm 0.56	75.87 \pm 0.57
Subject 12	60.96 \pm 0.92	75.42 \pm 0.29	71.97 \pm 0.33
Subject 14	72.09 \pm 1.82	82.80 \pm 0.45	74.28 \pm 0.40
Subject 15	73.57 \pm 1.91	87.27 \pm 0.47	79.72 \pm 0.45
Subject 16	68.03 \pm 2.06	77.45 \pm 0.40	69.11 \pm 0.36
Subject 17	62.39 \pm 1.54	67.90 \pm 0.30	62.31 \pm 0.26
Subject 18	81.34 \pm 2.41	90.75 \pm 0.44	86.25 \pm 0.47
Subject 19	62.83 \pm 1.97	67.76 \pm 0.36	63.25 \pm 0.32
Subject 20	61.44 \pm 1.24	67.90 \pm 0.35	60.86 \pm 0.24
Subject 21	75.31 \pm 2.09	89.41 \pm 0.37	79.41 \pm 0.39
Subject 22	61.61 \pm 1.34	68.33 \pm 0.34	61.61 \pm 0.28
Subject 23	62.89 \pm 1.41	72.86 \pm 0.43	66.26 \pm 0.35
Subject 24	63.99 \pm 1.22	76.04 \pm 0.38	69.53 \pm 0.34

TABLE A.8 – p-value par sujet pour la base de données Brain Invaders 2013a.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	<0.001*	1.000	0.999
Subject 2	<0.001*	1.000	1.000
Subject 5	<0.001*	1.000	1.000
Subject 7	<0.001*	1.000	0.939
Subject 8	<0.001*	1.000	0.624
Subject 9	<0.001*	1.000	0.913
Subject 10	<0.001*	0.984	0.038
Subject 11	<0.001*	0.882	0.003*
Subject 12	<0.001*	1.000	0.998
Subject 14	<0.001*	0.989	0.357
Subject 15	<0.001*	0.701	<0.001*
Subject 16	<0.001*	1.000	0.938
Subject 17	<0.001*	1.000	1.000
Subject 18	<0.001*	<0.001*	<0.001*
Subject 19	<0.001*	1.000	0.999
Subject 20	<0.001*	1.000	1.000
Subject 21	<0.001*	0.553	<0.001*
Subject 22	<0.001*	1.000	1.000
Subject 23	<0.001*	1.000	1.000
Subject 24	<0.001*	1.000	0.992

Database EPFL P300 (ERP)

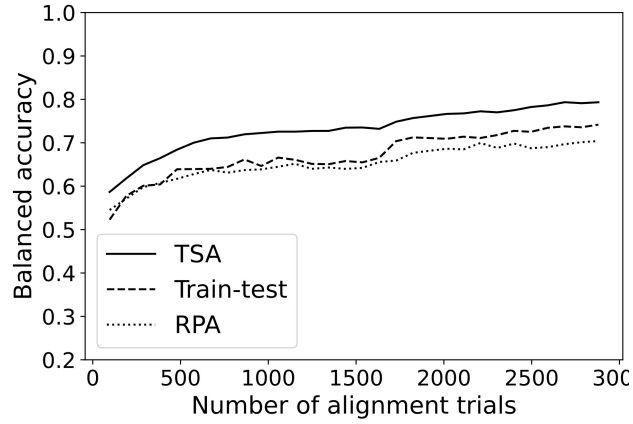


FIGURE A.9 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données EPFL P300.

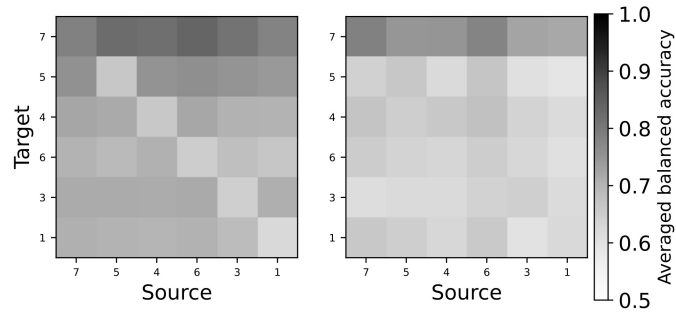


TABLE A.9 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données EPFL P300.

	Train-test acc	TSA acc	RPA acc
Subject 1	62.40 \pm 0.55	70.05 \pm 0.29	64.00 \pm 0.34
Subject 3	64.55 \pm 1.67	71.48 \pm 0.64	62.24 \pm 0.60
Subject 4	66.12 \pm 1.11	71.39 \pm 0.48	65.05 \pm 0.48
Subject 5	66.45 \pm 1.41	75.42 \pm 0.65	62.63 \pm 0.51
Subject 6	65.16 \pm 0.62	69.01 \pm 0.33	63.12 \pm 0.31
Subject 7	78.61 \pm 1.10	81.45 \pm 0.34	74.60 \pm 0.37

TABLE A.10 – p-value par sujet pour la base de données EPFL P300.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.003*	0.996	0.612
Subject 3	0.003*	0.987	0.410
Subject 4	<0.001*	0.987	0.350
Subject 5	<0.001*	0.975	0.447
Subject 6	0.001*	0.992	0.875
Subject 7	0.004*	0.001*	<0.001*

Database 001-2014 (MI)

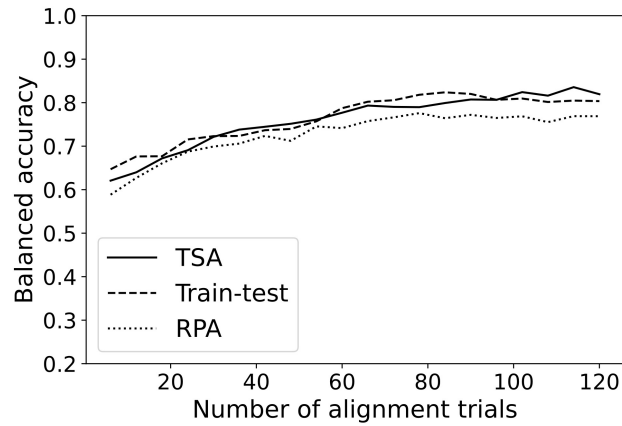


FIGURE A.10 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données 001-2014.

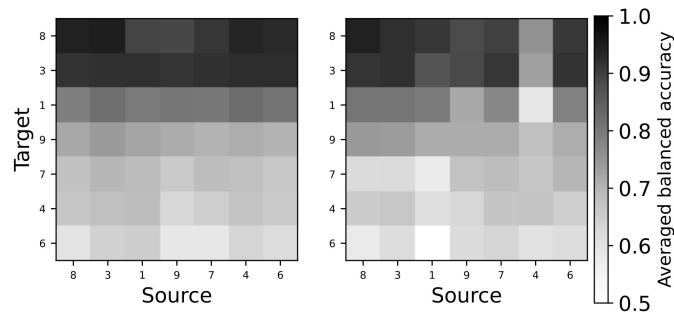


FIGURE A.11 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données 001-2014.

TABLE A.11 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données 001-2014.

	Train-test acc	TSA acc	RPA acc
Subject 1	79.90 \pm 1.11	80.88 \pm 0.40	74.63 \pm 0.94
Subject 3	92.04 \pm 1.54	92.06 \pm 0.55	87.19 \pm 0.79
Subject 4	67.08 \pm 2.24	66.12 \pm 0.96	64.50 \pm 0.77
Subject 6	61.47 \pm 1.67	61.61 \pm 0.74	59.09 \pm 0.75
Subject 7	68.61 \pm 2.74	67.75 \pm 1.08	64.24 \pm 0.83
Subject 8	94.21 \pm 1.06	91.82 \pm 0.63	88.03 \pm 0.87
Subject 9	71.45 \pm 1.69	71.73 \pm 0.63	71.73 \pm 0.60

TABLE A.12 – p-value par sujet pour la base de données 001-2014.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.028	0.227	0.094
Subject 3	0.026*	0.008*	0.006*
Subject 4	0.018	0.960	0.493
Subject 6	0.365	0.989	0.992
Subject 7	0.685	0.879	0.926
Subject 8	0.266	0.003*	0.004*
Subject 9	0.248	0.705	0.652

Database 001-2015 (MI)

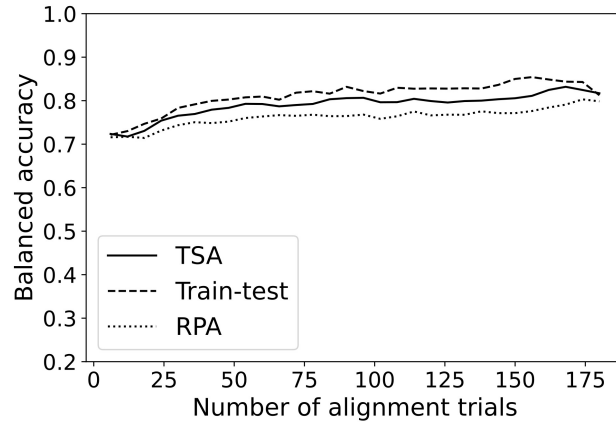


FIGURE A.12 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données 001-2015.

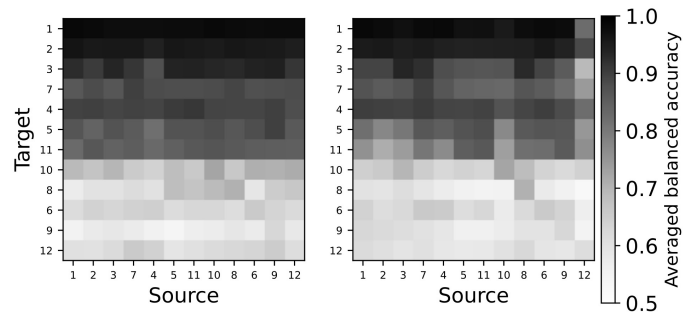


FIGURE A.13 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données 001-2015.

TABLE A.13 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données 001-2015.

	Train-test acc	TSA acc	RPA acc
Subject 1	98.45 \pm 0.24	97.85 \pm 0.06	96.12 \pm 0.29
Subject 2	95.68 \pm 0.20	95.56 \pm 0.11	94.13 \pm 0.16
Subject 3	93.71 \pm 0.50	92.39 \pm 0.21	87.01 \pm 0.38
Subject 4	89.30 \pm 0.34	89.57 \pm 0.10	88.70 \pm 0.18
Subject 5	87.21 \pm 0.53	86.48 \pm 0.19	82.77 \pm 0.34
Subject 6	65.78 \pm 0.91	62.87 \pm 0.28	62.17 \pm 0.32
Subject 7	89.93 \pm 0.41	87.84 \pm 0.18	84.22 \pm 0.32
Subject 8	70.54 \pm 1.45	63.11 \pm 0.53	57.20 \pm 0.43
Subject 9	62.86 \pm 1.34	57.05 \pm 0.31	58.99 \pm 0.33
Subject 10	72.50 \pm 1.46	68.12 \pm 0.34	64.91 \pm 0.33
Subject 11	86.22 \pm 1.16	84.96 \pm 0.31	78.47 \pm 0.39
Subject 12	61.35 \pm 0.99	62.33 \pm 0.37	59.86 \pm 0.34

TABLE A.14 – p-value par sujet pour la base de données 001-2015.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.350	<0.001*	<0.001*
Subject 2	0.085	0.001*	<0.001*
Subject 3	0.114	0.003*	0.001*
Subject 4	0.208	0.009*	0.008*
Subject 5	0.024*	0.051	0.021
Subject 6	0.025	0.996	0.990
Subject 7	0.259	0.014*	0.013*
Subject 8	0.228	0.974	0.976
Subject 9	0.003*	1.000	0.996
Subject 10	0.002*	0.954	0.720
Subject 11	0.014*	0.077	0.033
Subject 12	<0.001*	1.000	0.985

Database 002-2014 (MI)

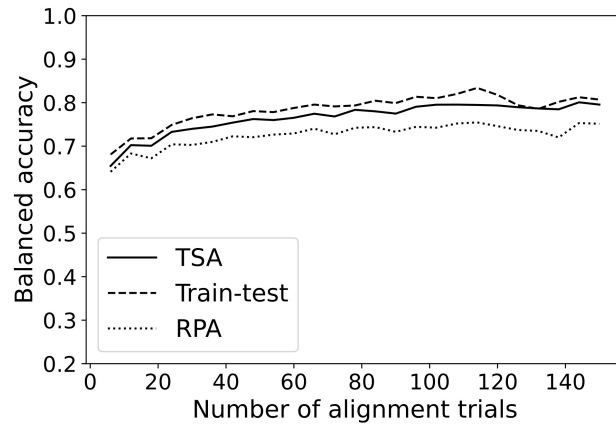


FIGURE A.14 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données 002-2014.

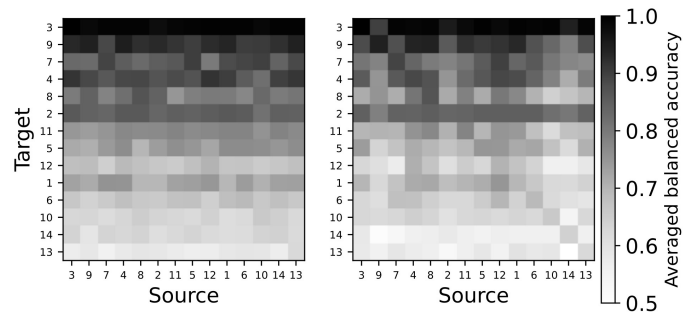


FIGURE A.15 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données 002-2014.

TABLE A.15 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données 002-2014.

	Train-test acc	TSA acc	RPA acc
Subject 1	69.67 \pm 0.98	72.84 \pm 0.34	67.88 \pm 0.43
Subject 2	83.45 \pm 0.84	84.81 \pm 0.24	83.60 \pm 0.23
Subject 3	99.86 \pm 0.07	98.99 \pm 0.11	97.63 \pm 0.20
Subject 4	88.84 \pm 1.00	88.35 \pm 0.31	82.17 \pm 0.43
Subject 5	74.90 \pm 1.90	74.35 \pm 0.57	69.34 \pm 0.55
Subject 6	69.33 \pm 0.94	66.58 \pm 0.34	63.86 \pm 0.40
Subject 7	89.42 \pm 1.10	85.80 \pm 0.37	81.64 \pm 0.39
Subject 8	86.96 \pm 0.66	80.25 \pm 0.41	73.07 \pm 0.54
Subject 9	94.42 \pm 0.67	92.10 \pm 0.24	88.84 \pm 0.34
Subject 10	65.93 \pm 1.63	63.01 \pm 0.39	62.09 \pm 0.41
Subject 11	78.05 \pm 1.07	76.61 \pm 0.36	70.67 \pm 0.51
Subject 12	70.24 \pm 1.28	67.14 \pm 0.33	62.28 \pm 0.40
Subject 13	62.33 \pm 1.65	57.86 \pm 0.37	56.66 \pm 0.41
Subject 14	64.19 \pm 1.44	62.39 \pm 0.38	55.91 \pm 0.36

TABLE A.16 – p-value par sujet pour la base de données 002-2014.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.010*	0.972	0.932
Subject 2	<0.001*	0.023*	<0.001*
Subject 3	0.001*	<0.001*	<0.001*
Subject 4	0.013*	0.002*	0.001*
Subject 5	0.010*	0.698	0.429
Subject 6	<0.001*	0.981	0.875
Subject 7	0.020*	<0.001*	<0.001*
Subject 8	0.049*	0.004*	0.004*
Subject 9	<0.001*	<0.001*	<0.001*
Subject 10	<0.001*	0.997	0.867
Subject 11	0.010*	0.296	0.134
Subject 12	0.537	0.958	0.954
Subject 13	<0.001*	1.000	0.991
Subject 14	<0.001*	0.999	0.844

Database 004-2014 (MI)

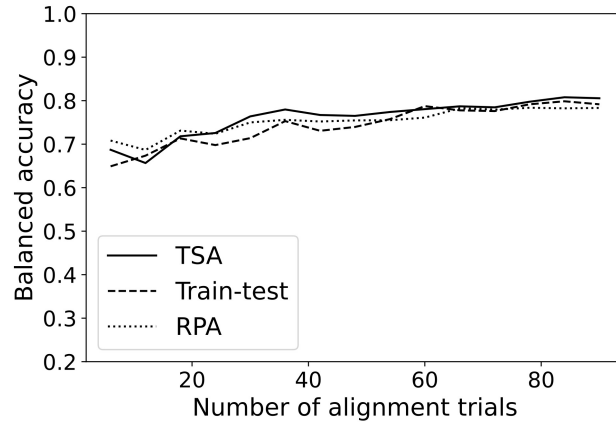


FIGURE A.16 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données 004-2014.

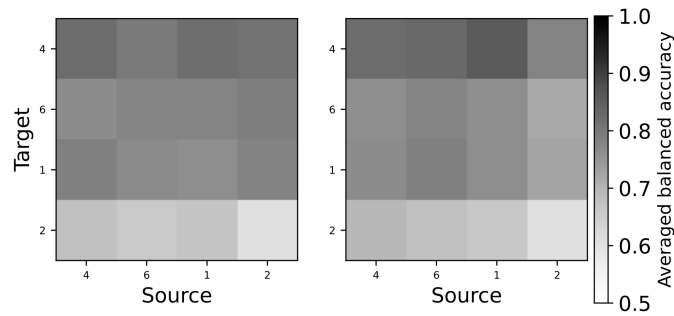


FIGURE A.17 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données 004-2014.

TABLE A.17 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données 004-2014.

	Train-test acc	TSA acc	RPA acc
Subject 1	76.28 \pm 0.99	77.96 \pm 0.45	76.12 \pm 0.48
Subject 2	60.68 \pm 0.55	67.00 \pm 0.85	67.91 \pm 0.84
Subject 4	82.46 \pm 2.12	81.13 \pm 1.02	82.27 \pm 0.84
Subject 6	77.97 \pm 1.91	77.92 \pm 1.02	74.78 \pm 0.80

TABLE A.18 – p-value par sujet pour la base de données 004-2014.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.593	0.453	0.489
Subject 2	0.023	0.999	0.991
Subject 4	0.348	0.066	0.036
Subject 6	0.987	0.240	0.398

Database Alexandre Motor Imagery (MI)

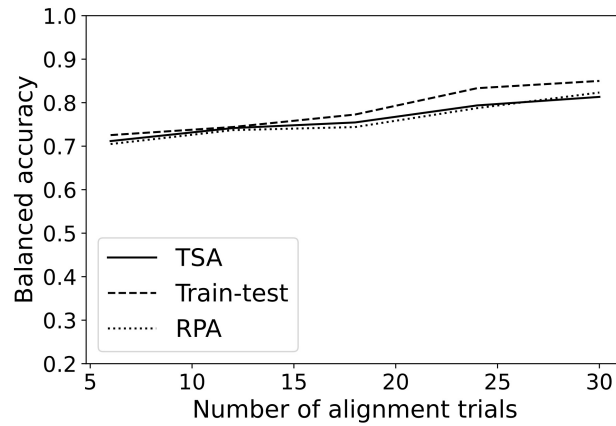


FIGURE A.18 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données Alexandre Motor Imagery.

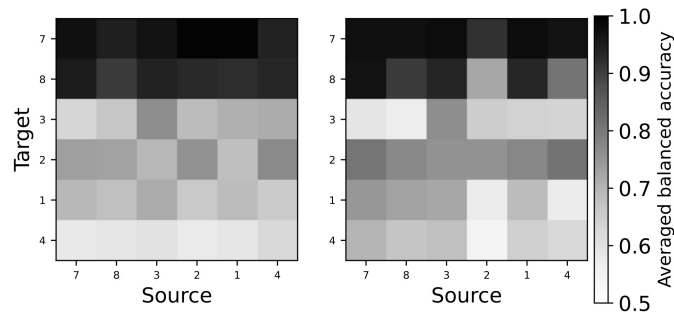


FIGURE A.19 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données Alexandre Motor Imagery.

TABLE A.19 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données Alexandre Motor Imagery.

	Train-test acc	TSA acc	RPA acc
Subject 1	68.81 \pm 3.69	68.08 \pm 1.71	66.88 \pm 2.39
Subject 2	75.73 \pm 2.17	72.25 \pm 1.75	78.52 \pm 1.38
Subject 3	76.36 \pm 7.45	68.19 \pm 2.20	61.77 \pm 2.71
Subject 4	62.33 \pm 6.42	58.79 \pm 2.00	64.77 \pm 1.89
Subject 7	97.09 \pm 1.98	96.78 \pm 0.67	96.25 \pm 0.64
Subject 8	90.74 \pm 3.23	93.72 \pm 1.14	87.41 \pm 2.18

TABLE A.20 – p-value par sujet pour la base de données Alexandre Motor Imagery.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.692	0.851	0.897
Subject 2	0.026	0.559	0.162
Subject 3	0.930	0.604	0.778
Subject 4	0.128	0.982	0.937
Subject 7	0.908	0.009*	0.014*
Subject 8	0.737	0.019	0.034

Database Cho2017 (MI)

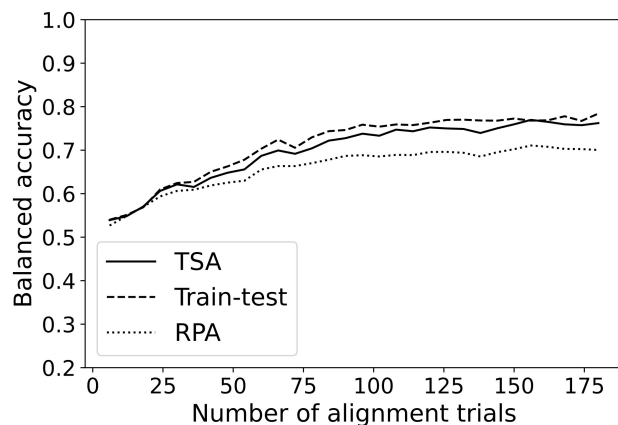


FIGURE A.20 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données Cho2017.

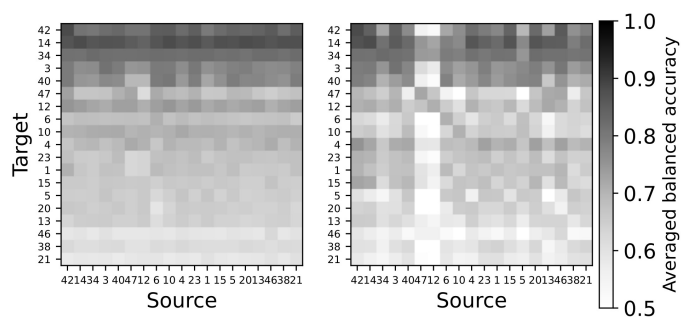


FIGURE A.21 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données Cho2017.

TABLE A.21 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données Cho2017.

	Train-test acc	TSA acc	RPA acc
Subject 1	67.86 \pm 2.00	66.70 \pm 0.43	65.84 \pm 0.43
Subject 3	81.39 \pm 0.81	77.78 \pm 0.18	74.96 \pm 0.32
Subject 4	69.26 \pm 1.49	69.24 \pm 0.40	69.25 \pm 0.50
Subject 5	65.90 \pm 2.12	65.03 \pm 0.50	59.12 \pm 0.55
Subject 6	70.74 \pm 2.11	67.65 \pm 0.38	59.80 \pm 0.53
Subject 10	70.55 \pm 1.91	70.71 \pm 0.45	62.32 \pm 0.43
Subject 12	70.97 \pm 1.61	73.01 \pm 0.39	67.68 \pm 0.29
Subject 13	63.60 \pm 1.14	64.30 \pm 0.25	60.83 \pm 0.25
Subject 14	88.26 \pm 1.52	86.99 \pm 0.37	82.06 \pm 0.45
Subject 15	66.15 \pm 1.64	65.63 \pm 0.38	65.20 \pm 0.37
Subject 20	64.96 \pm 1.13	64.64 \pm 0.26	60.79 \pm 0.33
Subject 21	60.63 \pm 1.50	58.63 \pm 0.24	56.54 \pm 0.30
Subject 23	68.64 \pm 1.62	66.69 \pm 0.37	64.94 \pm 0.41
Subject 34	82.73 \pm 0.37	81.89 \pm 0.09	79.97 \pm 0.15
Subject 38	61.98 \pm 1.27	61.51 \pm 0.27	58.53 \pm 0.35
Subject 40	76.73 \pm 1.89	76.30 \pm 0.44	72.21 \pm 0.47
Subject 42	88.34 \pm 3.32	83.02 \pm 0.73	75.78 \pm 0.77
Subject 46	62.55 \pm 1.79	59.08 \pm 0.37	54.64 \pm 0.26
Subject 47	72.34 \pm 1.66	67.98 \pm 0.36	63.03 \pm 0.42

TABLE A.22 – p-value par sujet pour la base de données Cho2017.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.002*	0.818	0.420
Subject 3	0.033*	<0.001*	<0.001*
Subject 4	0.005*	0.690	0.248
Subject 5	<0.001*	0.980	0.301
Subject 6	0.001*	0.379	0.013*
Subject 10	<0.001*	0.503	0.007*
Subject 12	<0.001*	0.086	<0.001*
Subject 13	0.001*	0.997	0.700
Subject 14	0.443	<0.001*	<0.001*
Subject 15	0.004*	0.964	0.843
Subject 20	0.385	0.991	0.988
Subject 21	<0.001*	1.000	0.999
Subject 23	0.009*	0.836	0.514
Subject 34	<0.001*	<0.001*	<0.001*
Subject 38	<0.001*	1.000	0.980
Subject 40	0.020*	0.002*	<0.001*
Subject 42	0.361	<0.001*	<0.001*
Subject 46	0.126	1.000	0.995
Subject 47	<0.001*	0.038*	<0.001*

Database Grosse-Wentrup 2009 (MI)

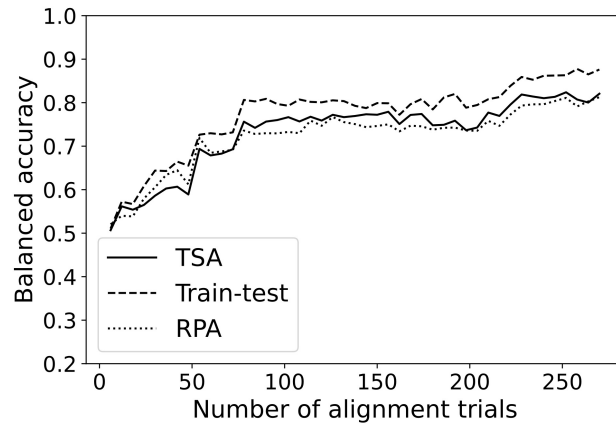


FIGURE A.22 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données Grosse-Wentrup 2009.

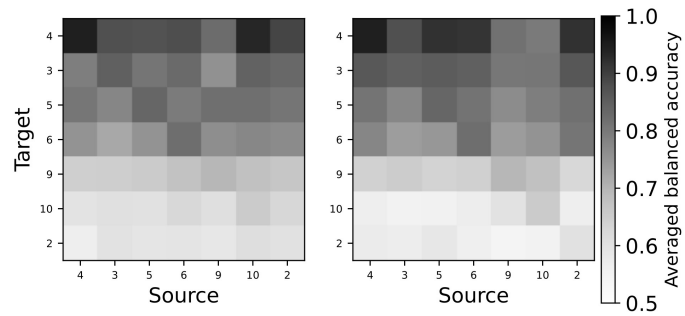


FIGURE A.23 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données Grosse-Wentrup 2009.

TABLE A.23 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données Grosse-Wentrup 2009.

	Train-test acc	TSA acc	RPA acc
Subject 2	60.29 \pm 2.35	59.19 \pm 0.76	56.55 \pm 0.82
Subject 3	84.68 \pm 1.48	81.07 \pm 0.63	83.97 \pm 0.69
Subject 4	94.69 \pm 1.11	88.05 \pm 0.69	87.38 \pm 0.63
Subject 5	83.44 \pm 2.28	80.48 \pm 0.87	79.56 \pm 0.84
Subject 6	82.18 \pm 1.81	75.63 \pm 0.65	76.12 \pm 0.69
Subject 9	69.66 \pm 0.89	66.24 \pm 0.39	64.64 \pm 0.39
Subject 10	65.45 \pm 1.62	61.14 \pm 0.56	56.88 \pm 0.40

TABLE A.24 – p-value par sujet pour la base de données Grosse-Wentrup 2009.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 2	0.503	0.995	0.983
Subject 3	0.173	0.012*	0.015*
Subject 4	0.784	<0.001*	0.003*
Subject 5	0.535	0.026	0.055
Subject 6	0.292	0.063	0.090
Subject 9	0.168	0.744	0.626
Subject 10	0.031	0.975	0.940

Database Physionet Motor Imagery (MI)

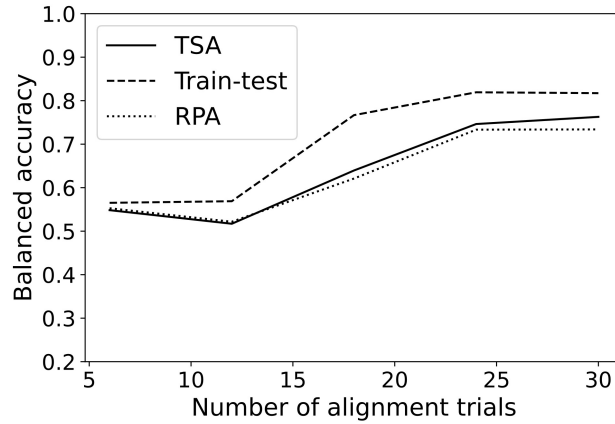


FIGURE A.24 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données Physionet Motor Imagery.

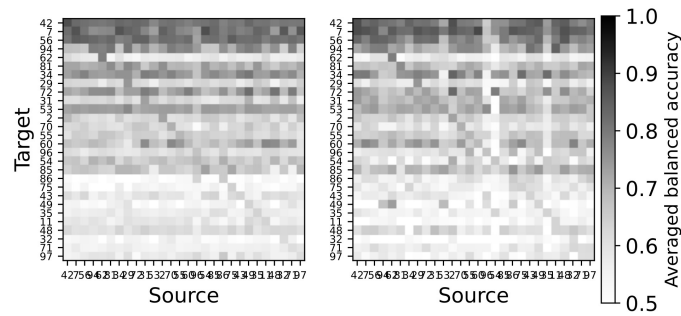


FIGURE A.25 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données Physionet Motor Imagery.

TABLE A.25 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données Physionet Motor Imagery.

	Train-test acc	TSA acc	RPA acc
Subject 2	72.99 \pm 12.35	62.75 \pm 1.47	62.98 \pm 1.49
Subject 7	87.21 \pm 5.96	80.40 \pm 1.66	82.78 \pm 1.26
Subject 11	63.64 \pm 8.32	58.00 \pm 1.39	51.88 \pm 1.19
Subject 29	74.84 \pm 12.64	61.53 \pm 1.78	59.55 \pm 1.72
Subject 31	73.38 \pm 8.95	62.19 \pm 1.42	67.78 \pm 1.13
Subject 32	61.90 \pm 2.85	53.24 \pm 0.65	50.66 \pm 0.71
Subject 34	75.74 \pm 6.48	75.04 \pm 1.41	73.17 \pm 1.31
Subject 35	63.82 \pm 5.31	57.13 \pm 1.25	55.03 \pm 1.00
Subject 42	87.93 \pm 3.38	80.96 \pm 0.64	79.93 \pm 0.73
Subject 43	63.85 \pm 7.08	58.87 \pm 1.46	56.37 \pm 1.43
Subject 48	63.28 \pm 5.30	61.44 \pm 1.11	59.80 \pm 1.28
Subject 49	63.82 \pm 8.37	53.90 \pm 1.46	52.67 \pm 1.16
Subject 53	73.10 \pm 8.17	73.01 \pm 1.30	69.98 \pm 1.33
Subject 54	66.73 \pm 8.37	66.28 \pm 1.18	58.60 \pm 0.95
Subject 55	69.61 \pm 5.71	63.31 \pm 1.23	64.42 \pm 1.30
Subject 56	84.37 \pm 6.63	81.52 \pm 1.21	77.67 \pm 1.20
Subject 60	69.16 \pm 2.76	68.52 \pm 0.79	69.05 \pm 1.11
Subject 62	79.10 \pm 7.90	57.68 \pm 1.79	56.62 \pm 1.76
Subject 70	70.71 \pm 11.28	61.86 \pm 1.28	58.70 \pm 1.16
Subject 71	61.45 \pm 6.85	55.86 \pm 1.13	56.83 \pm 1.22
Subject 72	73.63 \pm 5.44	73.69 \pm 1.27	69.19 \pm 1.26
Subject 75	64.21 \pm 3.86	53.04 \pm 0.59	56.81 \pm 0.80
Subject 81	76.59 \pm 4.04	70.78 \pm 0.83	67.16 \pm 0.87
Subject 85	66.39 \pm 3.84	65.52 \pm 1.18	68.20 \pm 1.21
Subject 86	65.51 \pm 5.34	51.68 \pm 0.62	58.04 \pm 0.95
Subject 94	79.47 \pm 4.35	73.25 \pm 0.83	72.51 \pm 0.85
Subject 96	67.24 \pm 5.79	60.46 \pm 0.56	59.79 \pm 0.85
Subject 97	61.30 \pm 2.95	57.96 \pm 0.63	54.99 \pm 0.58

TABLE A.26 – p-value par sujet pour la base de données Physionet Motor Imagery.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 2	0.519	<0.001*	<0.001*
Subject 7	0.362	<0.001*	<0.001*
Subject 11	0.491	0.793	0.772
Subject 29	0.488	<0.001*	<0.001*
Subject 31	0.064	<0.001*	<0.001*
Subject 32	0.474	0.892	0.863
Subject 34	0.046*	<0.001*	<0.001*
Subject 35	<0.001*	0.741	0.011*
Subject 42	0.471	<0.001*	<0.001*
Subject 43	0.973	0.481	0.865
Subject 48	0.977	0.595	0.919
Subject 49	0.365	0.907	0.893
Subject 53	0.094	<0.001*	<0.001*
Subject 54	<0.001*	0.131	<0.001*
Subject 55	0.690	0.003*	0.006*
Subject 56	0.834	<0.001*	<0.001*
Subject 60	0.078	0.005*	0.002*
Subject 62	0.601	<0.001*	<0.001*
Subject 70	0.013*	0.004*	<0.001*
Subject 71	0.446	0.974	0.981
Subject 72	0.245	<0.001*	<0.001*
Subject 75	0.165	0.578	0.348
Subject 81	0.079	<0.001*	<0.001*
Subject 85	0.050	0.122	0.019
Subject 86	0.576	0.304	0.371
Subject 94	0.308	<0.001*	<0.001*
Subject 96	0.002*	0.004*	<0.001*
Subject 97	0.490	0.897	0.901

Database Weibo 2014 (MI)

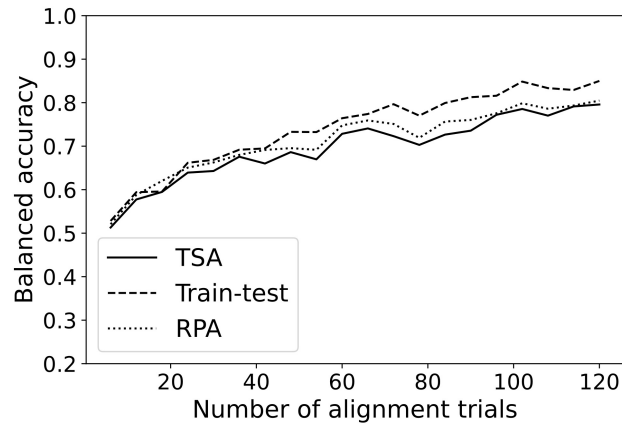


FIGURE A.26 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données Weibo 2014.

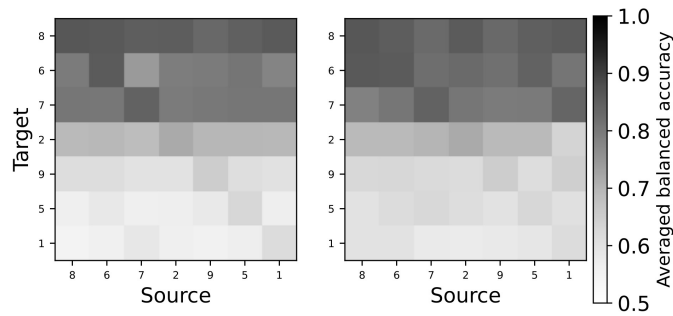


FIGURE A.27 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données Weibo 2014.

TABLE A.27 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données Weibo 2014.

	Train-test acc	TSA acc	RPA acc
Subject 1	61.65 \pm 2.09	56.46 \pm 0.58	58.71 \pm 0.69
Subject 2	71.85 \pm 3.18	69.35 \pm 1.17	68.40 \pm 1.20
Subject 5	62.83 \pm 2.22	57.12 \pm 0.62	61.09 \pm 0.80
Subject 6	85.73 \pm 3.10	78.71 \pm 1.23	83.01 \pm 1.08
Subject 7	84.33 \pm 1.78	80.33 \pm 0.64	80.54 \pm 0.73
Subject 8	86.35 \pm 2.52	85.04 \pm 1.10	84.58 \pm 0.88
Subject 9	65.07 \pm 1.36	60.62 \pm 0.45	62.54 \pm 0.54

TABLE A.28 – p-value par sujet pour la base de données Weibo 2014.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.780	0.957	0.972
Subject 2	0.981	0.349	0.483
Subject 5	0.869	0.949	0.959
Subject 6	0.934	0.009*	0.008*
Subject 7	0.906	0.006*	0.010*
Subject 8	0.943	0.004*	0.006*
Subject 9	0.902	0.863	0.906

Database Zhou 2016 (MI)

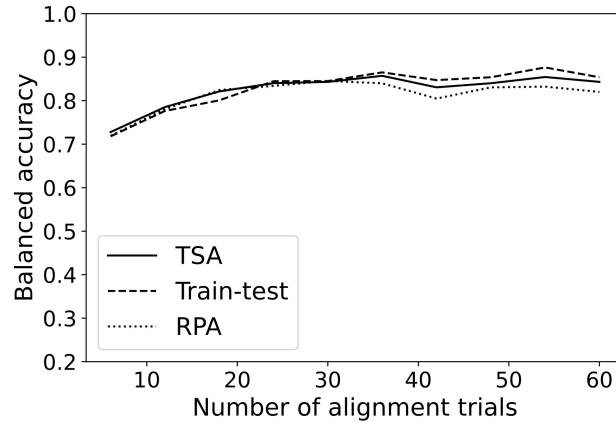


FIGURE A.28 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données Zhou 2016.

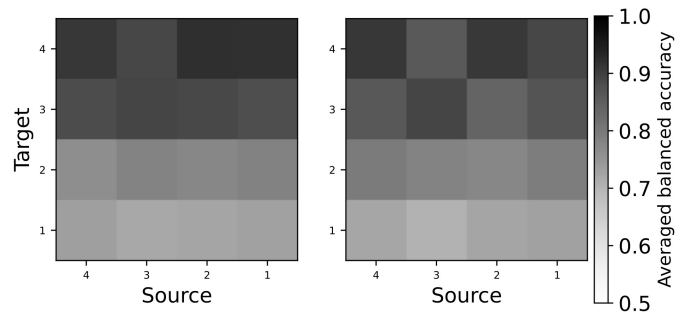


FIGURE A.29 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données Zhou 2016.

TABLE A.29 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données Zhou 2016.

	Train-test acc	TSA acc	RPA acc
Subject 1	73.21 \pm 3.10	72.69 \pm 1.40	71.81 \pm 1.65
Subject 2	77.68 \pm 2.20	77.68 \pm 1.23	79.15 \pm 0.92
Subject 3	89.17 \pm 1.09	88.28 \pm 0.46	85.67 \pm 0.58
Subject 4	91.20 \pm 1.58	91.09 \pm 0.74	88.62 \pm 0.84

TABLE A.30 – p-value par sujet pour la base de données Zhou 2016.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.230	0.958	0.972
Subject 2	0.143	0.811	0.764
Subject 3	0.113	0.099	0.068
Subject 4	0.532	0.060	0.050

Database SSVEP Exoskeleton (SSVEP)

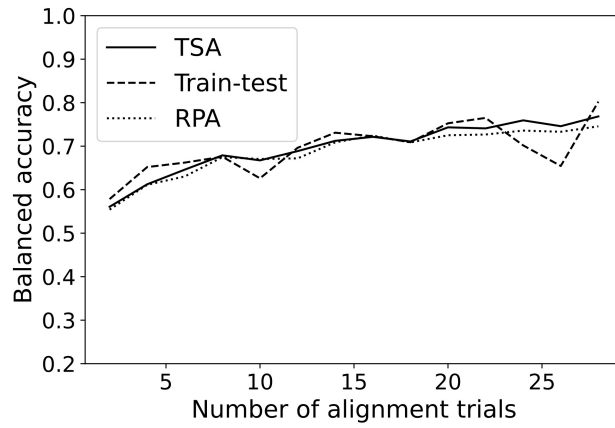


FIGURE A.30 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données SSVEP Exoskeleton.

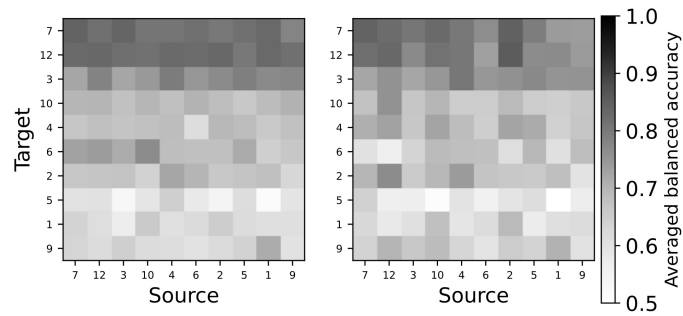


FIGURE A.31 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données SSVEP Exoskeleton.

TABLE A.31 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données SSVEP Exoskeleton.

	Train-test acc	TSA acc	RPA acc
Subject 1	60.89 \pm 2.98	61.96 \pm 1.00	62.03 \pm 1.19
Subject 2	66.10 \pm 3.99	67.23 \pm 1.10	68.47 \pm 1.18
Subject 3	72.46 \pm 2.64	76.67 \pm 1.09	75.76 \pm 1.22
Subject 4	68.69 \pm 3.42	66.95 \pm 1.06	69.26 \pm 1.15
Subject 5	61.51 \pm 3.70	57.86 \pm 1.36	56.68 \pm 1.19
Subject 6	68.04 \pm 1.72	70.49 \pm 1.26	64.11 \pm 1.17
Subject 7	84.13 \pm 2.23	81.30 \pm 0.91	79.36 \pm 0.99
Subject 9	60.11 \pm 3.83	63.34 \pm 1.11	65.68 \pm 1.18
Subject 10	69.81 \pm 2.41	68.89 \pm 0.85	67.32 \pm 0.87
Subject 12	83.26 \pm 2.82	82.22 \pm 0.83	78.58 \pm 0.73

TABLE A.32 – p-value par sujet pour la base de données SSVEP Exoskeleton.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.004*	0.990	0.968
Subject 2	0.888	0.868	0.939
Subject 3	0.178	0.121	0.030
Subject 4	0.132	0.779	0.671
Subject 5	0.031	0.998	0.995
Subject 6	0.041	0.614	0.268
Subject 7	0.373	<0.001*	<0.001*
Subject 9	0.017	0.996	0.993
Subject 10	0.655	0.556	0.640
Subject 12	0.548	<0.001*	0.001*

Database SSVEP Nakanishi (SSVEP)

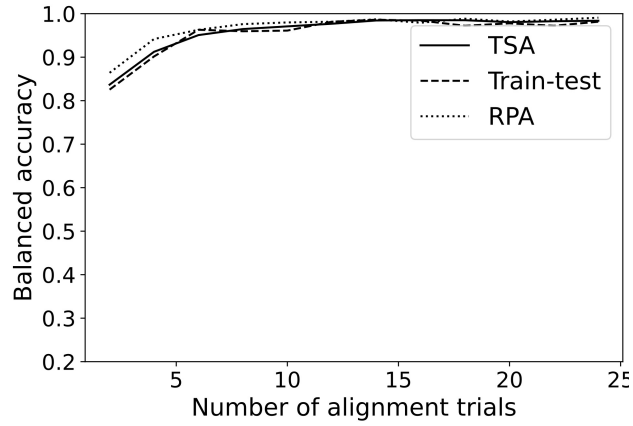


FIGURE A.32 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données SSVEP Nakanishi.

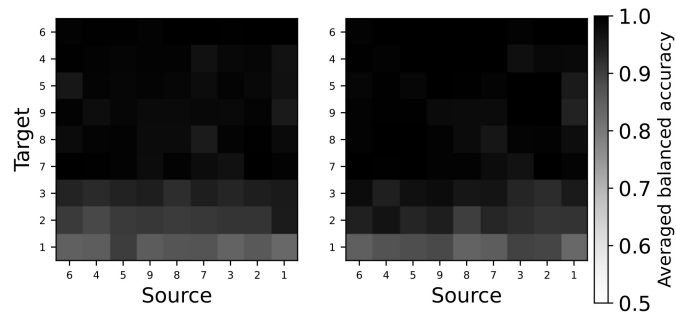


FIGURE A.33 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données SSVEP Nakanishi.

TABLE A.33 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données SSVEP Nakanishi.

	Train-test acc	TSA acc	RPA acc
Subject 1	83.27 \pm 2.52	86.20 \pm 1.13	87.08 \pm 1.01
Subject 2	91.59 \pm 4.18	91.16 \pm 1.08	93.41 \pm 1.15
Subject 3	93.68 \pm 2.06	94.12 \pm 0.74	95.97 \pm 0.76
Subject 4	99.38 \pm 0.42	98.54 \pm 0.49	99.24 \pm 0.33
Subject 5	98.79 \pm 0.92	98.18 \pm 0.54	99.08 \pm 0.51
Subject 6	99.40 \pm 0.60	99.81 \pm 0.10	99.93 \pm 0.07
Subject 7	97.77 \pm 1.51	99.08 \pm 0.44	99.29 \pm 0.43
Subject 8	97.92 \pm 2.08	98.31 \pm 0.65	98.88 \pm 0.55
Subject 9	98.17 \pm 1.30	98.14 \pm 0.55	98.59 \pm 0.58

TABLE A.34 – p-value par sujet pour la base de données SSVEP Nakanishi.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.112	1.000	1.000
Subject 2	0.729	0.993	0.997
Subject 3	0.871	0.858	0.973
Subject 4	0.967	0.039	0.105
Subject 5	0.907	0.071	0.187
Subject 6	0.978	0.033	0.097
Subject 7	0.918	0.063	0.173
Subject 8	0.653	0.124	0.183
Subject 9	0.998	0.096	0.322

Database SSVEP Wang (SSVEP)

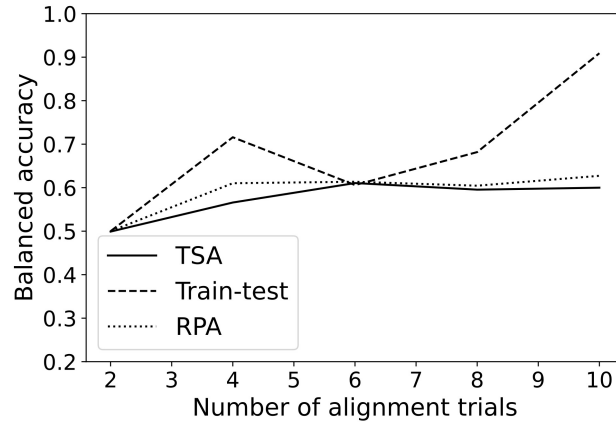


FIGURE A.34 – Précision en fonction du nombre d’essais d’alignement pour les méthodes TSA et RPA pour la base de données SSVEP Wang.

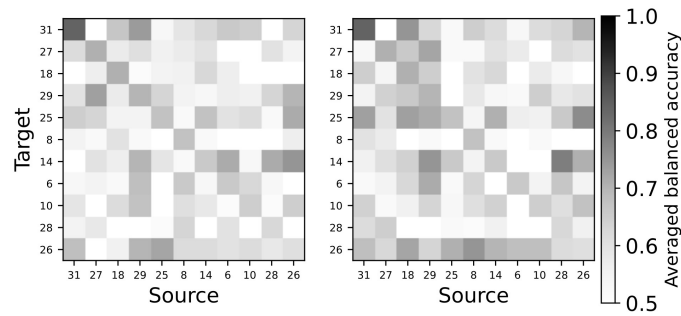


FIGURE A.35 – Diagrammes de sériation représentatifs pour les méthodes TSA (à gauche) et RPA (à droite) pour la base de données SSVEP Wang.

TABLE A.35 – Précision équilibrée \pm erreur type, moyennées sur le nombre d’essais d’alignement et les sources pour le train-test, TSA et RPA pour les bons sujets de la base de données SSVEP Wang.

	Train-test acc	TSA acc	RPA acc
Subject 6	65.83 \pm 9.17	56.42 \pm 1.87	58.42 \pm 2.26
Subject 8	67.50 \pm 10.90	50.08 \pm 2.38	50.67 \pm 2.19
Subject 10	65.00 \pm 10.00	57.58 \pm 2.63	59.25 \pm 2.56
Subject 14	65.83 \pm 9.17	62.00 \pm 2.55	62.33 \pm 2.53
Subject 18	70.83 \pm 8.33	52.88 \pm 2.28	56.08 \pm 2.45
Subject 25	67.50 \pm 10.90	60.42 \pm 2.37	65.83 \pm 2.62
Subject 26	60.83 \pm 4.86	62.00 \pm 2.46	68.08 \pm 2.84
Subject 27	70.83 \pm 8.33	55.58 \pm 2.74	57.83 \pm 2.81
Subject 28	62.50 \pm 9.68	52.83 \pm 2.81	52.00 \pm 2.53
Subject 29	70.00 \pm 9.35	60.83 \pm 2.63	58.25 \pm 2.71
Subject 31	84.17 \pm 9.17	61.00 \pm 2.47	61.50 \pm 2.59

TABLE A.36 – p-value par sujet pour la base de données SSVEP Wang.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 6	0.135	0.005*	<0.001*
Subject 8	0.703	<0.001*	0.003*
Subject 10	0.677	0.001*	0.002*
Subject 14	0.937	0.002*	0.016*
Subject 18	0.900	<0.001*	0.020*
Subject 25	0.273	<0.001*	<0.001*
Subject 26	0.577	0.458	0.510
Subject 27	0.743	<0.001*	<0.001*
Subject 28	0.997	0.012*	0.414
Subject 29	0.704	0.010*	0.048
Subject 31	0.911	<0.001*	<0.001*

A.2 Résultats détaillés du Chapitre 3

Base de données 003-2015 (ERP)

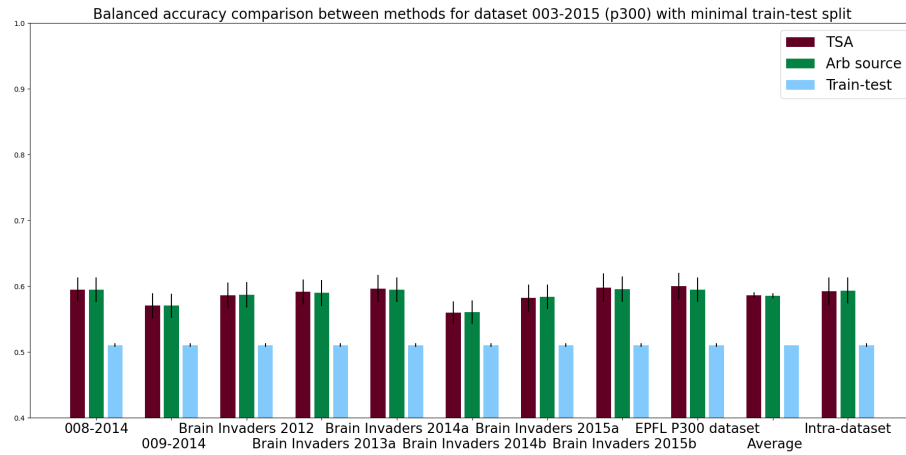


FIGURE A.36 – Précision équilibrée inter-configurations pour la base de donnée 003-2015 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

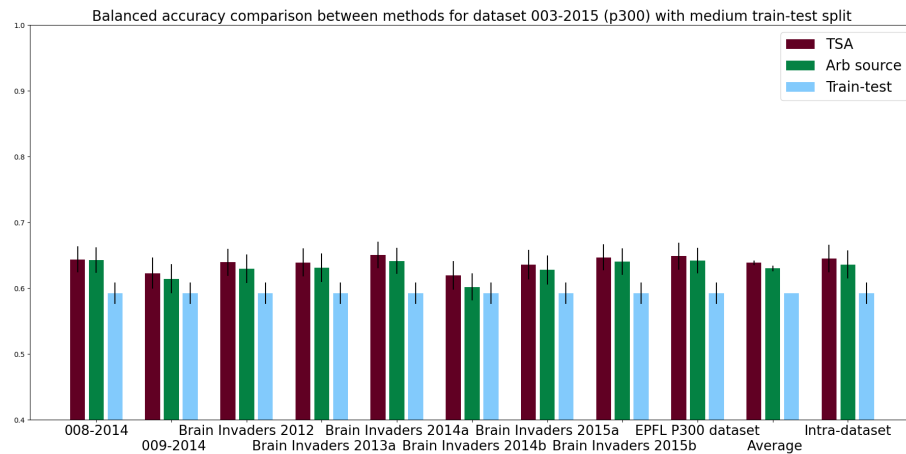


FIGURE A.37 – Précision équilibrée inter-configurations pour la base de donnée 003-2015 avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

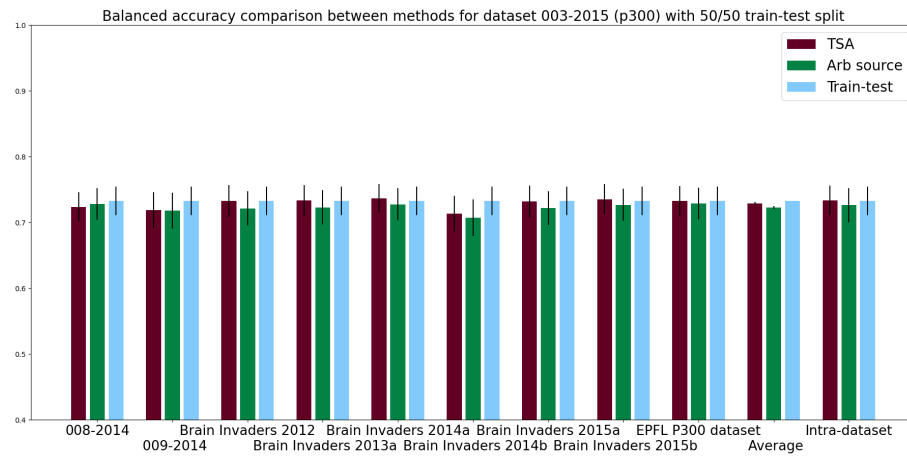


FIGURE A.38 – Précision équilibrée inter-configurations pour la base de donnée 003-2015 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données 008-2014 (ERP)

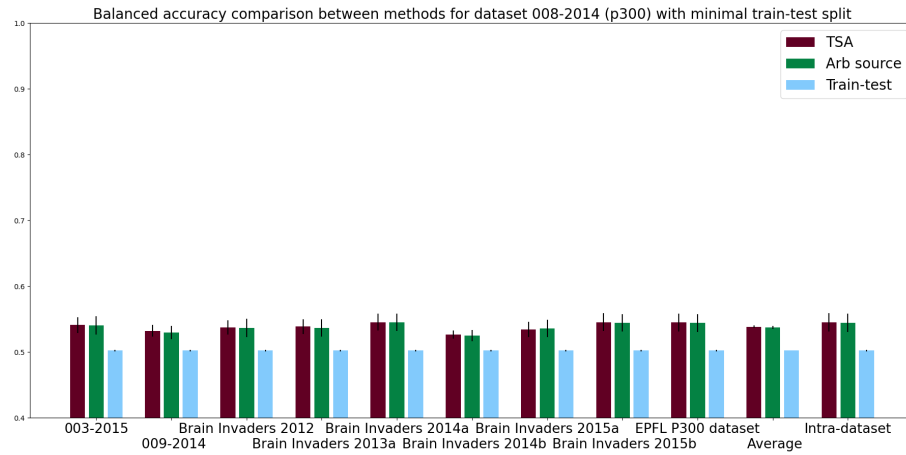


FIGURE A.39 – Précision équilibrée inter-configurations pour la base de donnée 008-2014 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

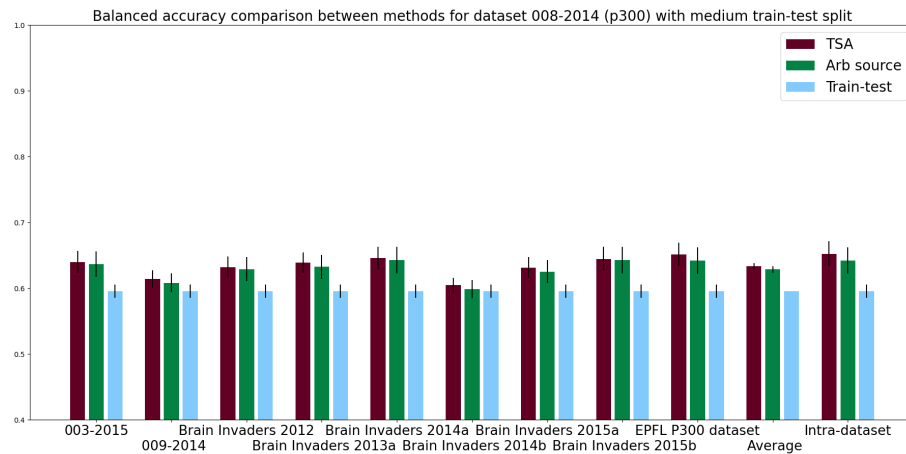


FIGURE A.40 – Précision équilibrée inter-configurations pour la base de donnée 008-2014 avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

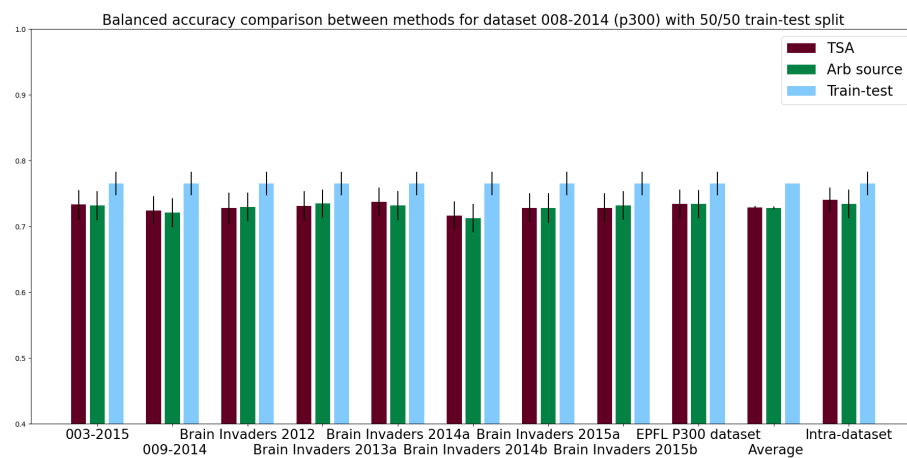


FIGURE A.41 – Précision équilibrée inter-configurations pour la base de donnée 008-2014 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données 009-2014 (ERP)

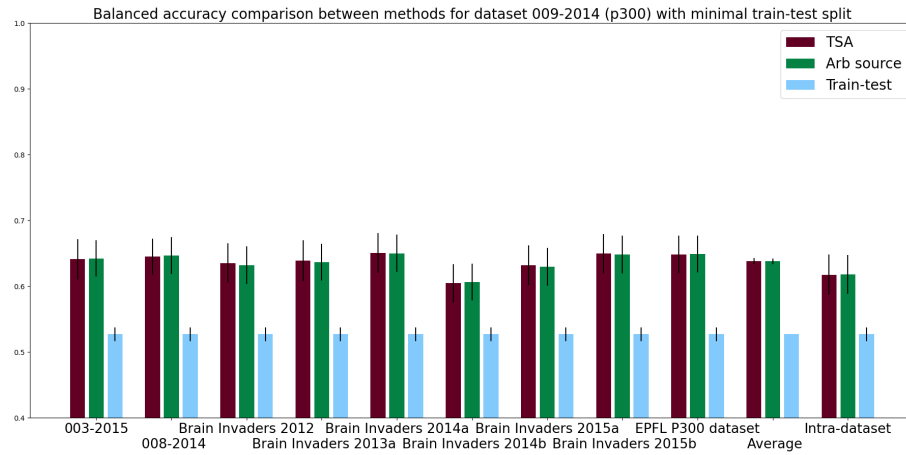


FIGURE A.42 – Précision équilibrée inter-configurations pour la base de donnée 009-2014 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

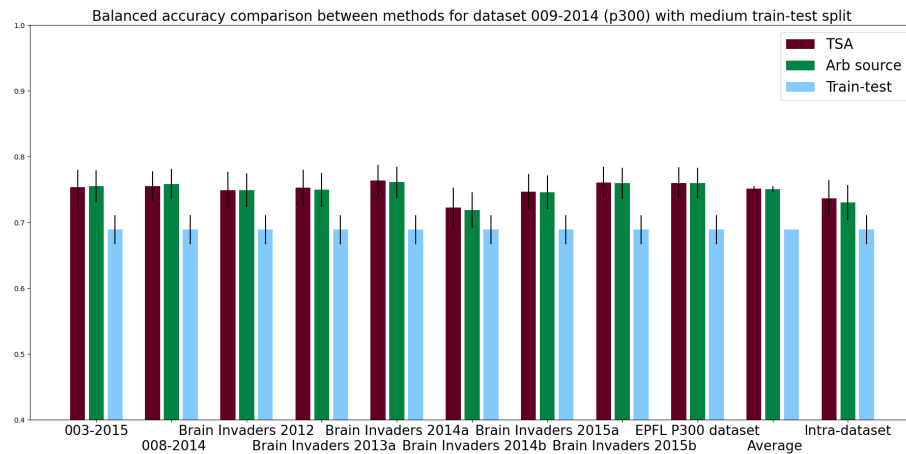


FIGURE A.43 – Précision équilibrée inter-configurations pour la base de donnée 009-2014 avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

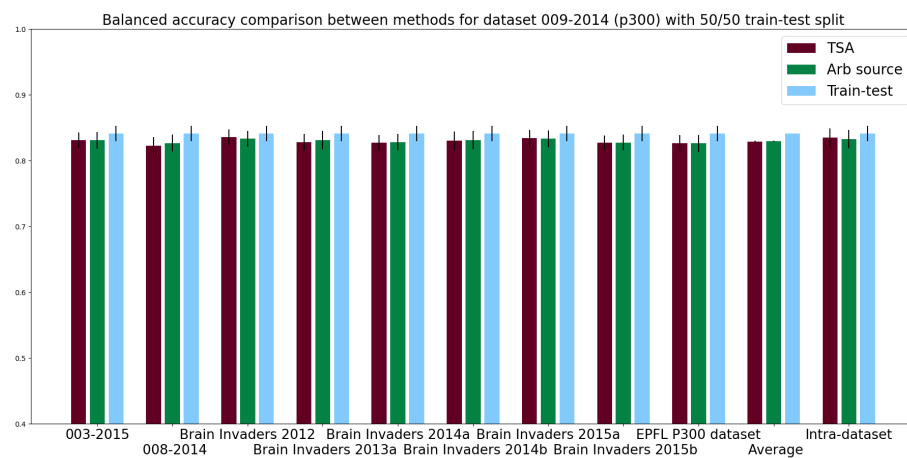


FIGURE A.44 – Précision équilibrée inter-configurations pour la base de donnée 009-2014 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Brain Invaders 2012 (ERP)

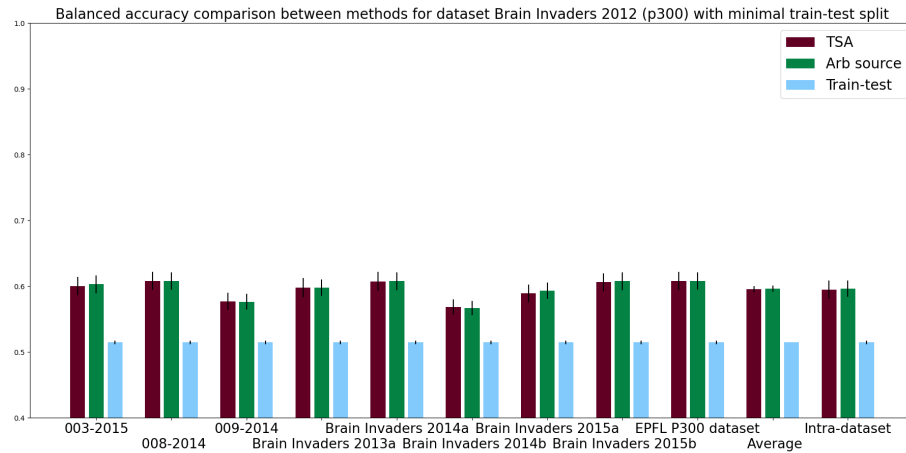


FIGURE A.45 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2012 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

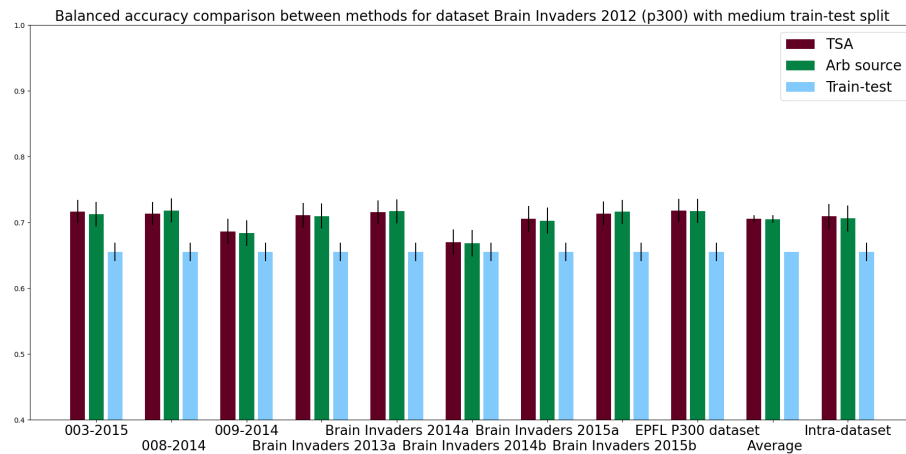


FIGURE A.46 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2012 avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

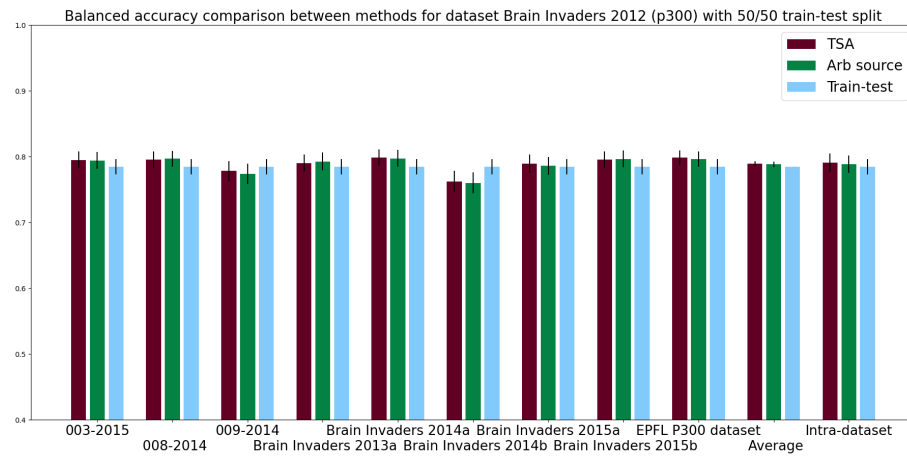


FIGURE A.47 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2012 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Brain Invaders 2013a (ERP)

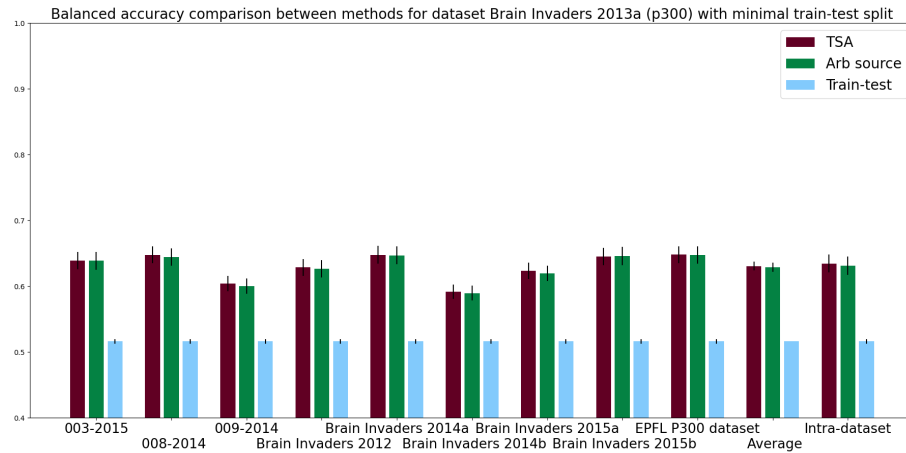


FIGURE A.48 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2013a avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

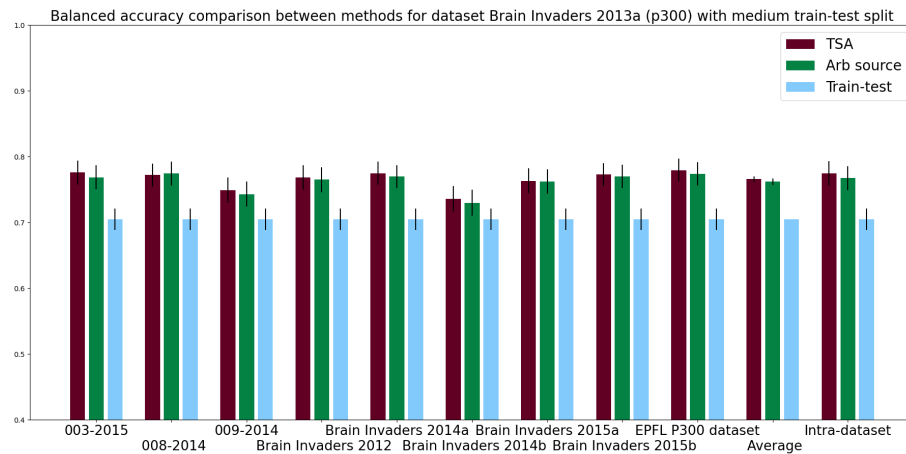


FIGURE A.49 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2013a avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

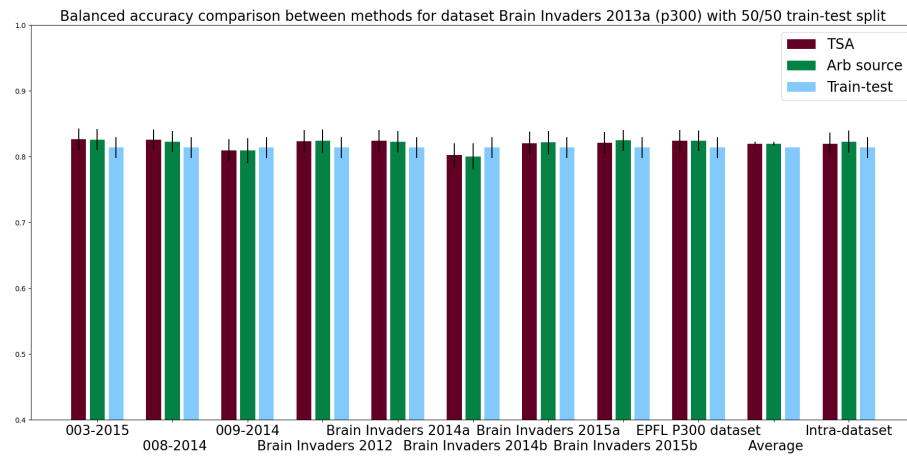


FIGURE A.50 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2013a avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Brain Invaders 2014a (ERP)

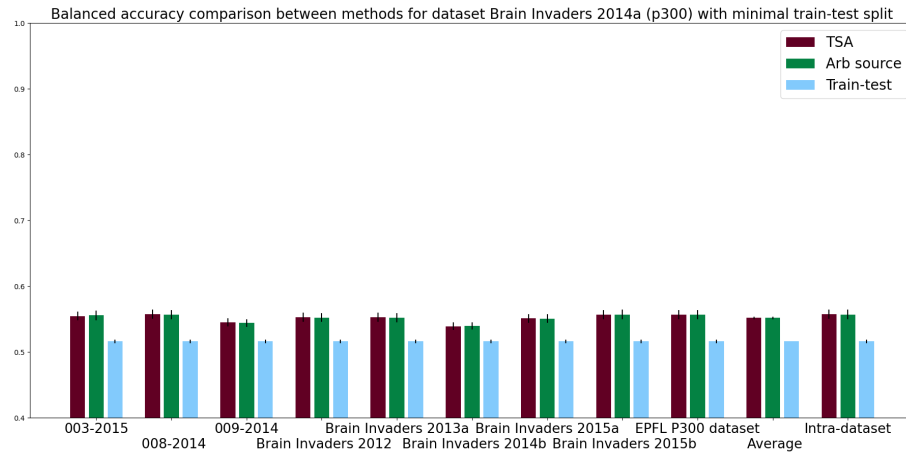


FIGURE A.51 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2014a avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

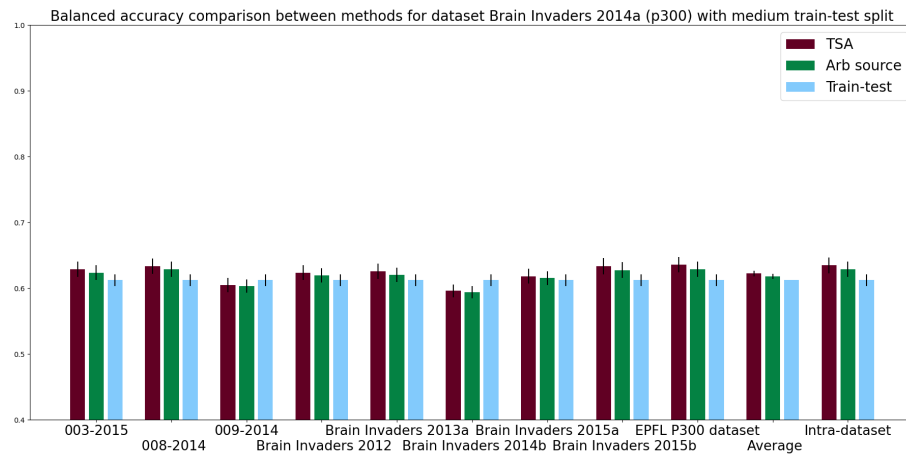


FIGURE A.52 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2014a avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

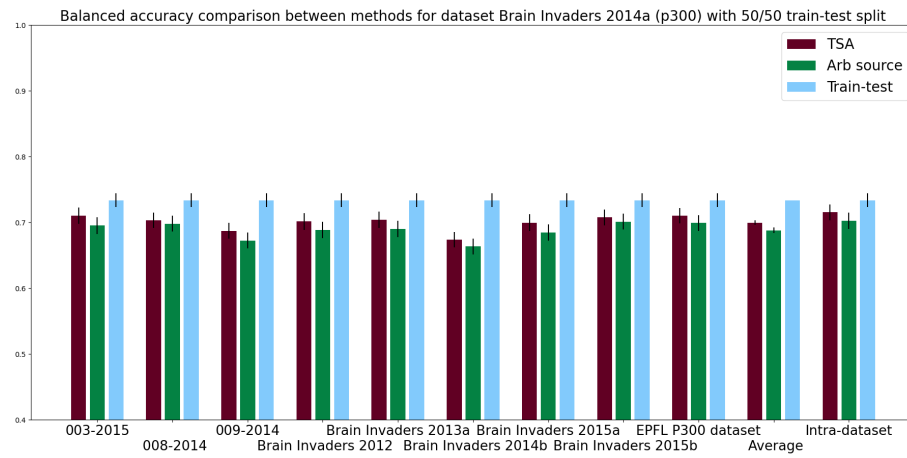


FIGURE A.53 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2014a avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Brain Invaders 2014b (ERP)

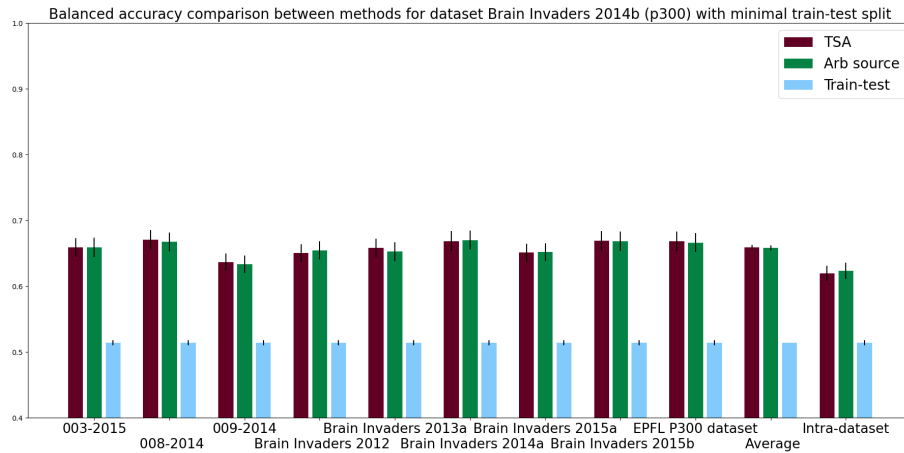


FIGURE A.54 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2014b avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

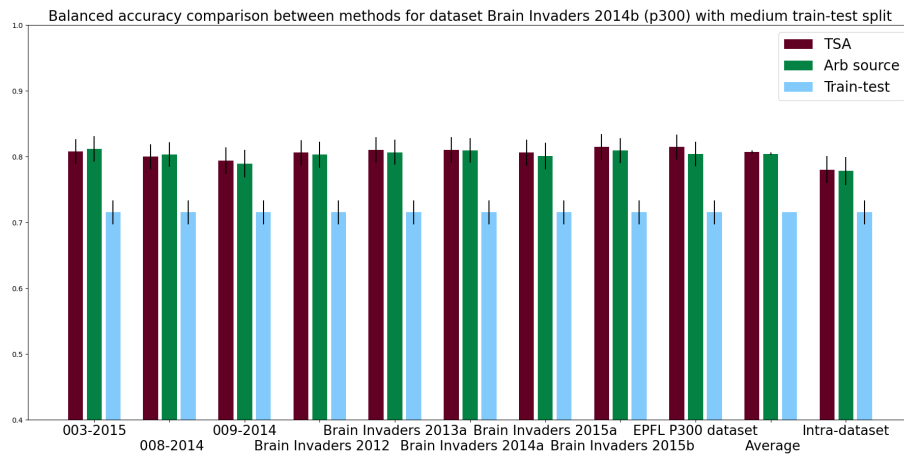


FIGURE A.55 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2014b avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

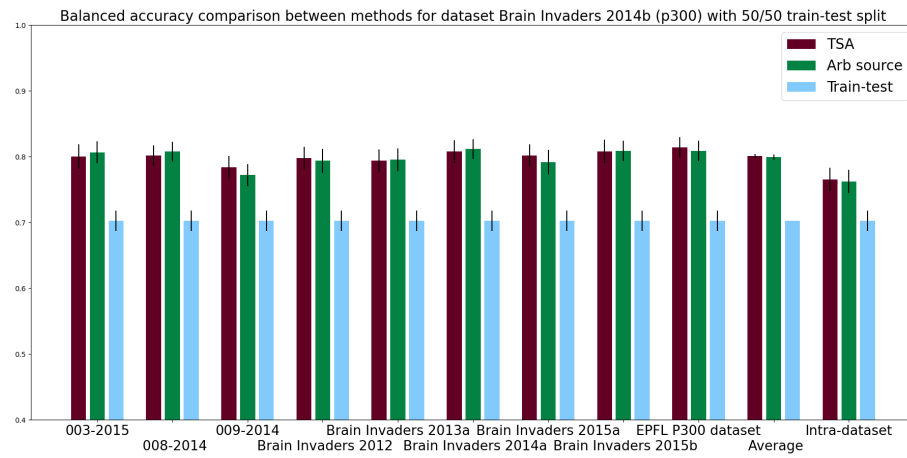


FIGURE A.56 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2014b avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Brain Invaders 2015a (ERP)

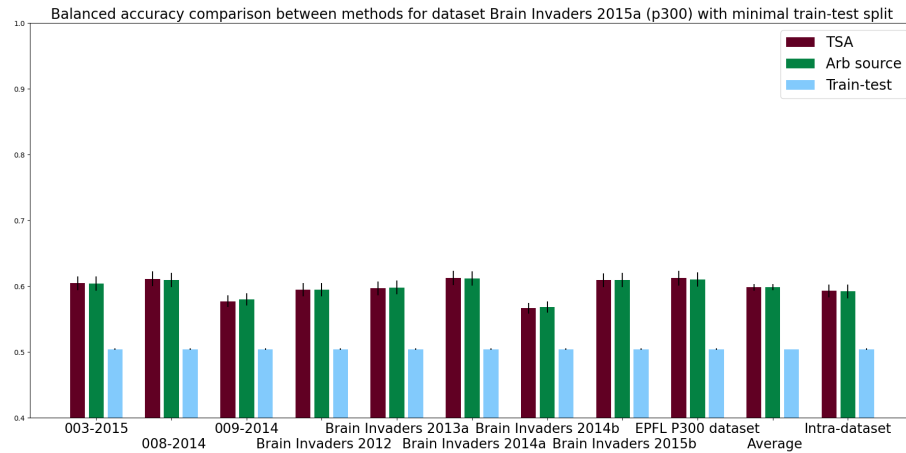


FIGURE A.57 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2015a avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

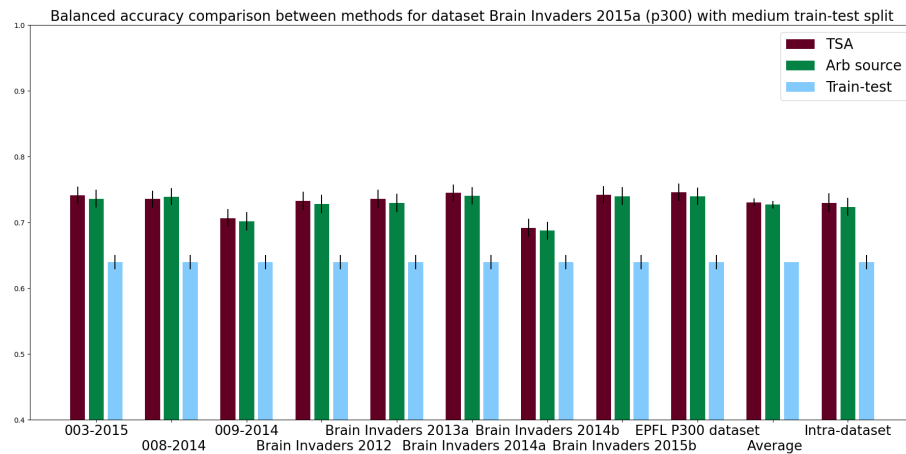


FIGURE A.58 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2015a avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

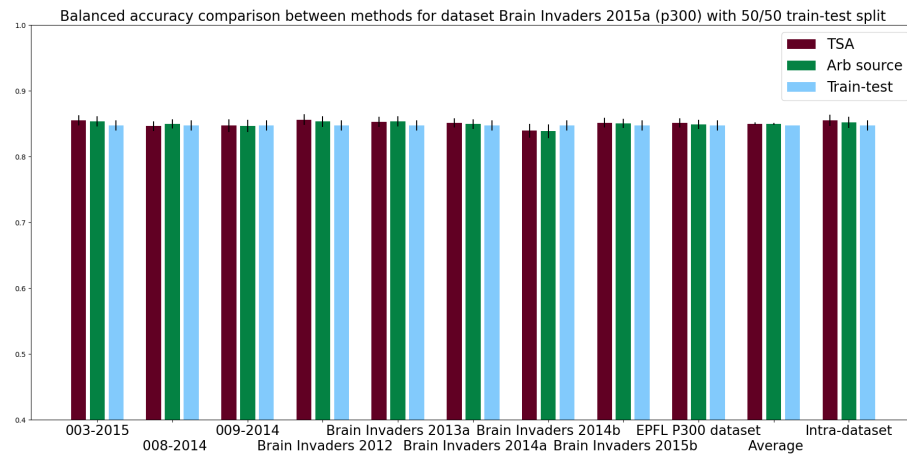


FIGURE A.59 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2015a avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Brain Invaders 2015b (ERP)

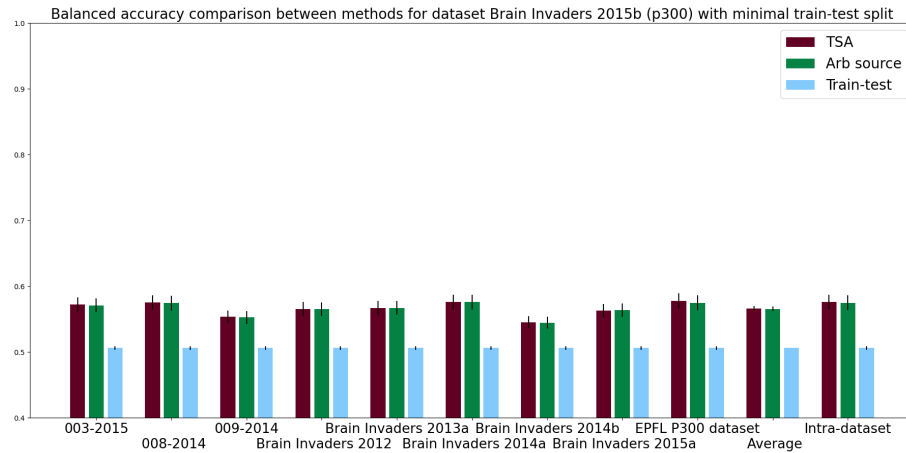


FIGURE A.60 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2015b avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

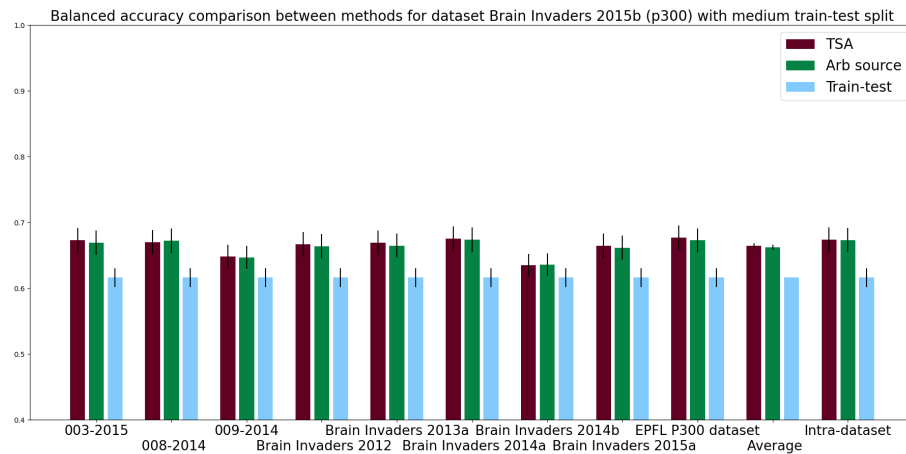


FIGURE A.61 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2015b avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

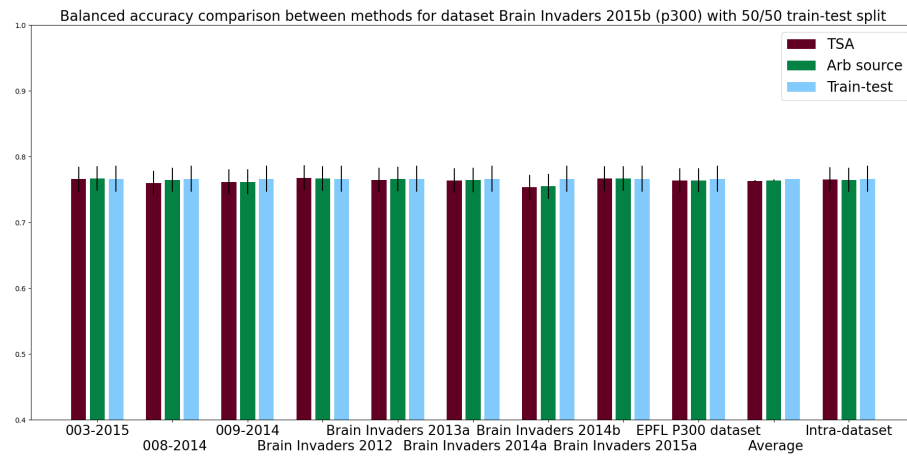


FIGURE A.62 – Précision équilibrée inter-configurations pour la base de donnée Brain Invaders 2015b avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données EPFL P300 dataset (ERP)

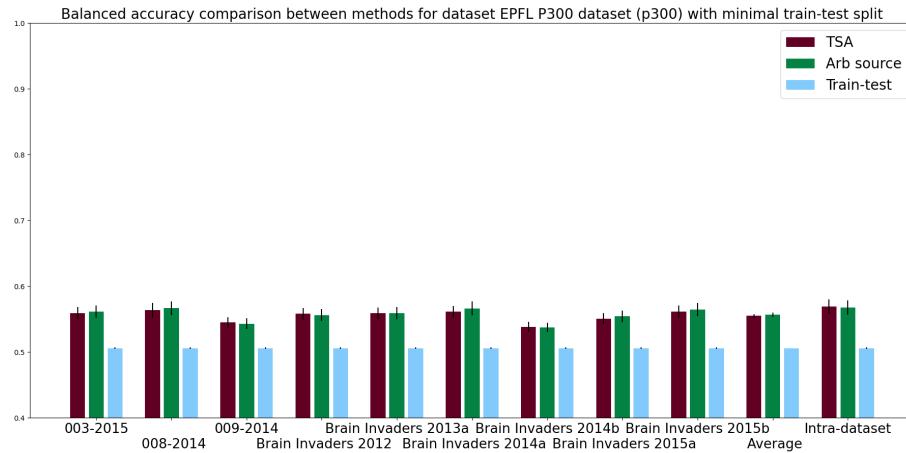


FIGURE A.63 – Précision équilibrée inter-configurations pour la base de donnée EPFL P300 dataset avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

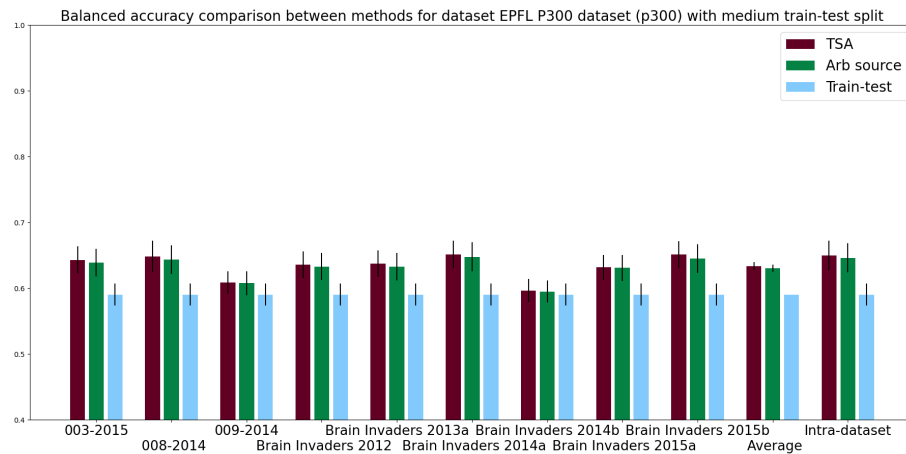


FIGURE A.64 – Précision équilibrée inter-configurations pour la base de donnée EPFL P300 dataset avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

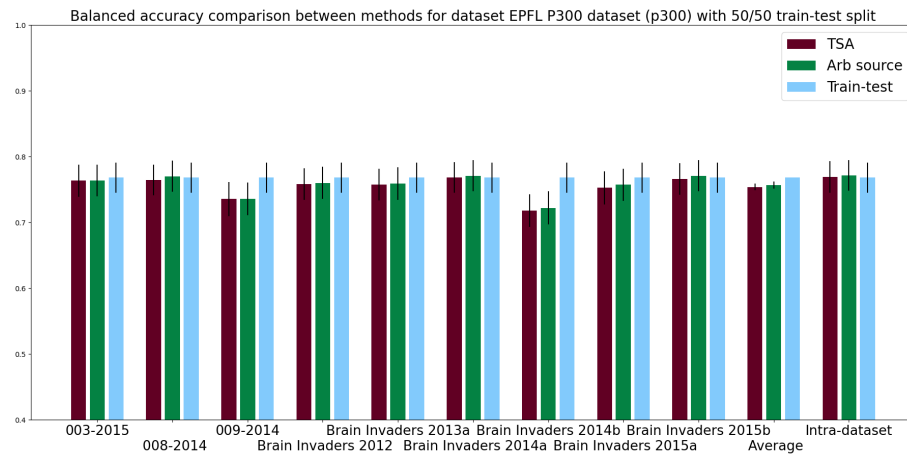


FIGURE A.65 – Précision équilibrée inter-configurations pour la base de donnée EPFL P300 dataset avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données 001-2014 (MI)

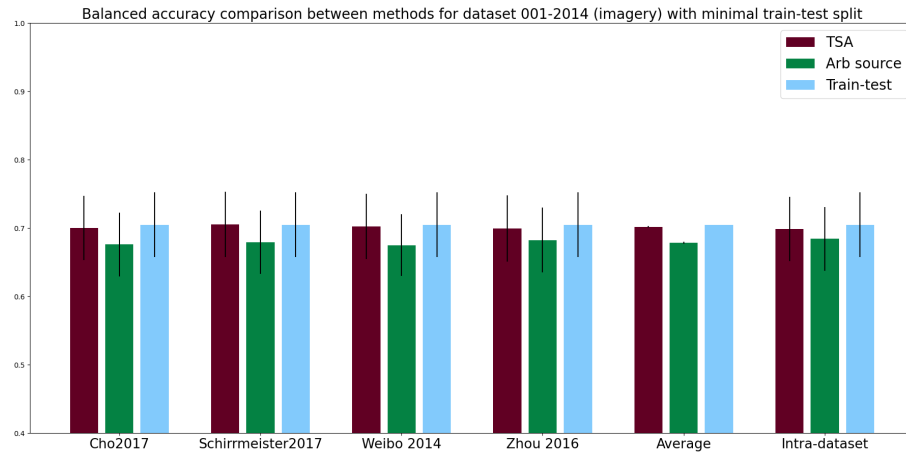


FIGURE A.66 – Précision équilibrée inter-configurations pour la base de donnée 001-2014 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

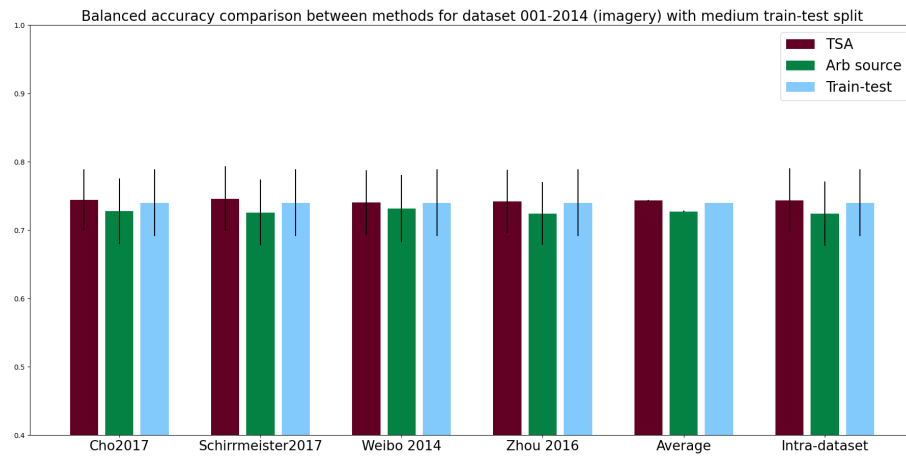


FIGURE A.67 – Précision équilibrée inter-configurations pour la base de donnée 001-2014 avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

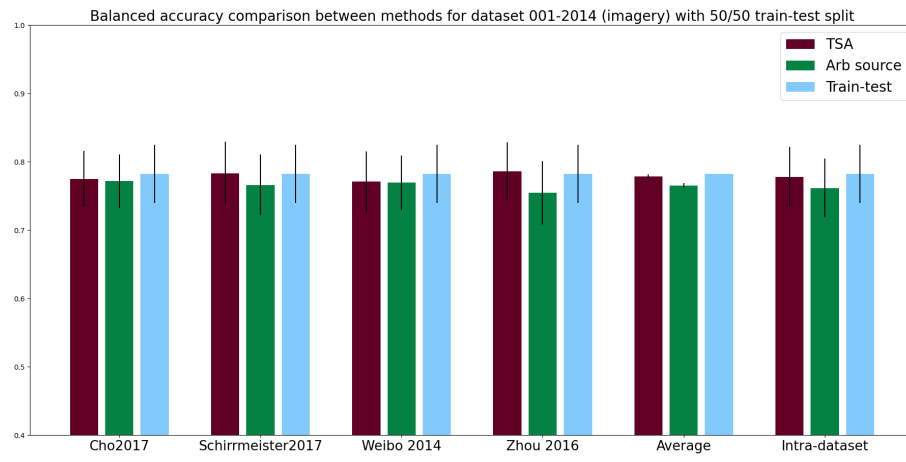


FIGURE A.68 – Précision équilibrée inter-configurations pour la base de donnée 001-2014 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Cho2017 (MI)

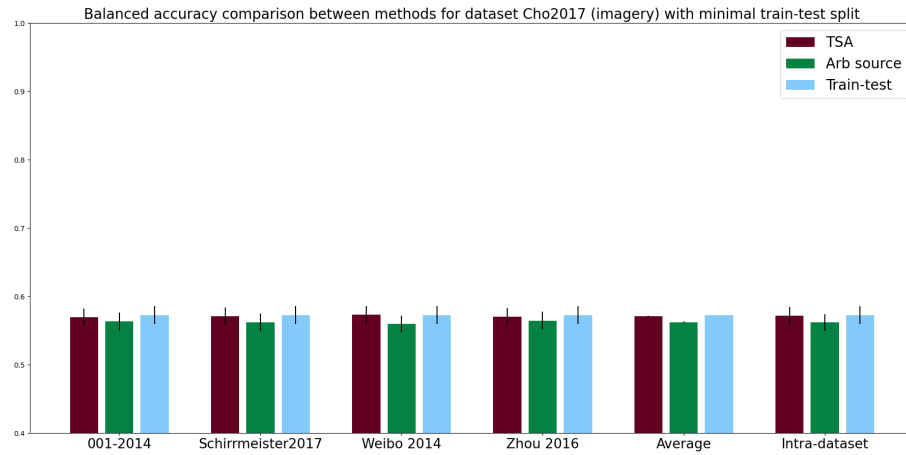


FIGURE A.69 – Précision équilibrée inter-configurations pour la base de donnée Cho2017 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

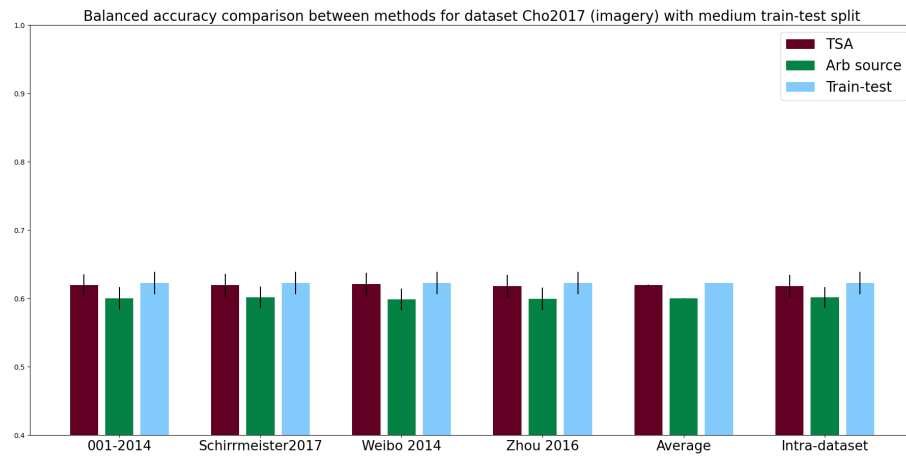


FIGURE A.70 – Précision équilibrée inter-configurations pour la base de donnée Cho2017 avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

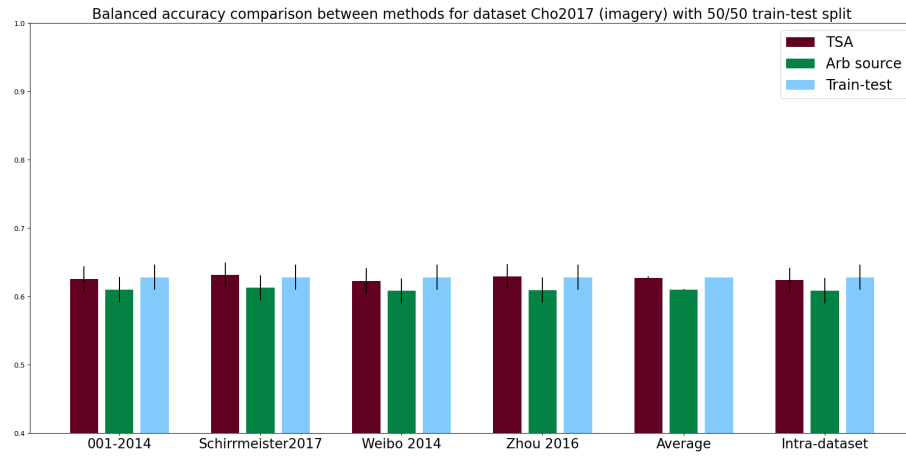


FIGURE A.71 – Précision équilibrée inter-configurations pour la base de donnée Cho2017 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Schirmeister2017 (MI)

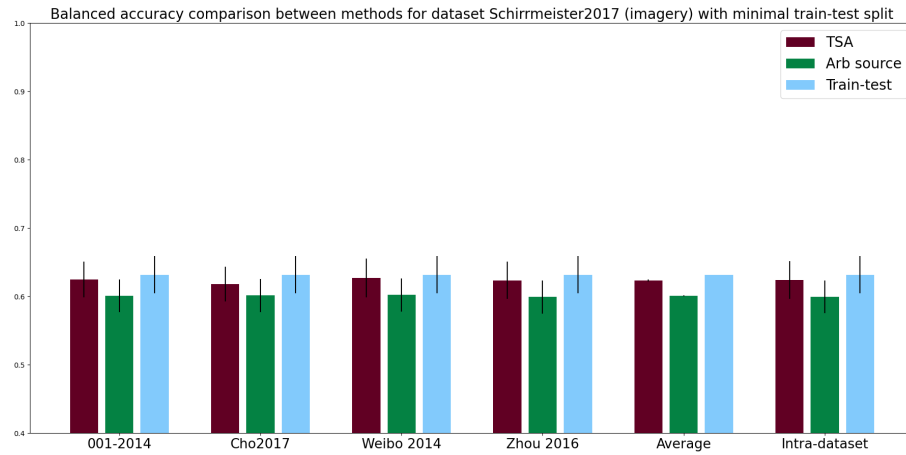


FIGURE A.72 – Précision équilibrée inter-configurations pour la base de donnée Schirmeister2017 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

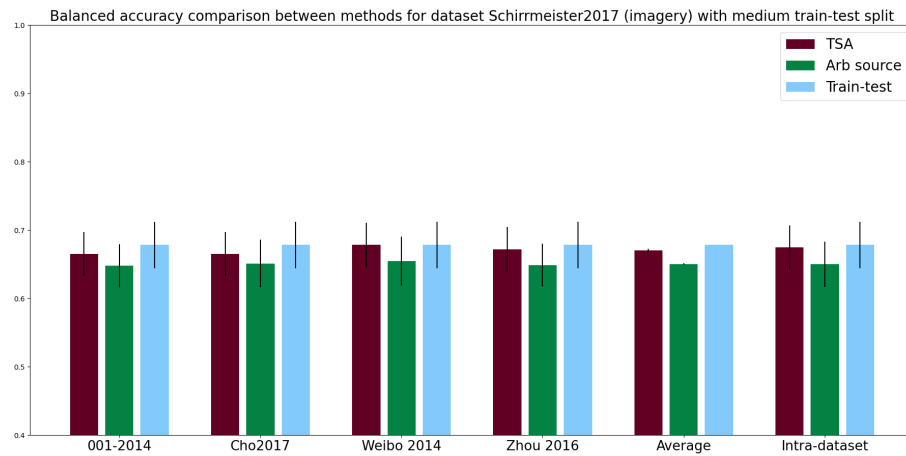


FIGURE A.73 – Précision équilibrée inter-configurations pour la base de donnée Schirmeister2017 avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

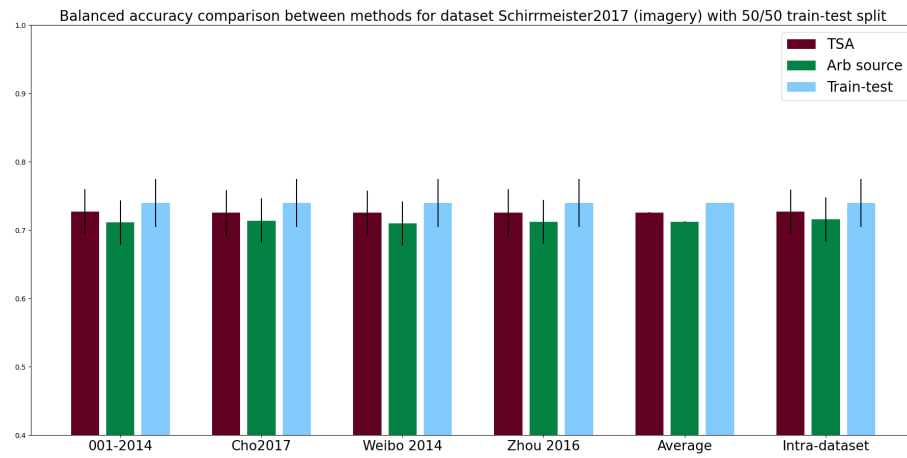


FIGURE A.74 – Précision équilibrée inter-configurations pour la base de donnée Schirmeister2017 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Weibo 2014 (MI)

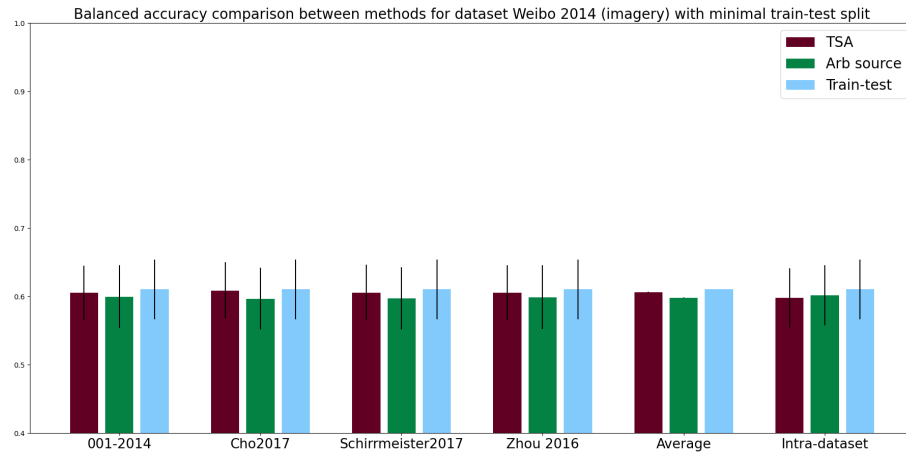


FIGURE A.75 – Précision équilibrée inter-configurations pour la base de donnée Weibo 2014 avec un fractionnement minimal. Les résultats sont présentés comme en figure 3.4.

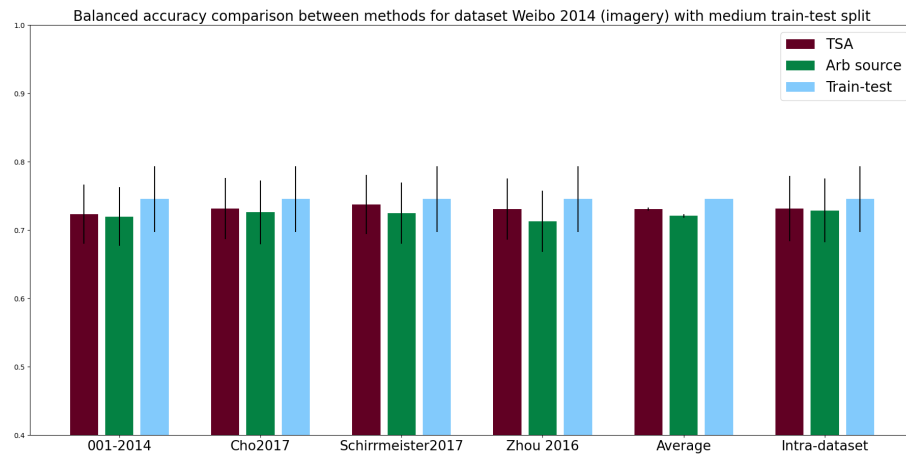


FIGURE A.76 – Précision équilibrée inter-configurations pour la base de donnée Weibo 2014 avec un fractionnement moyen. Les résultats sont présentés comme en figure 3.4.

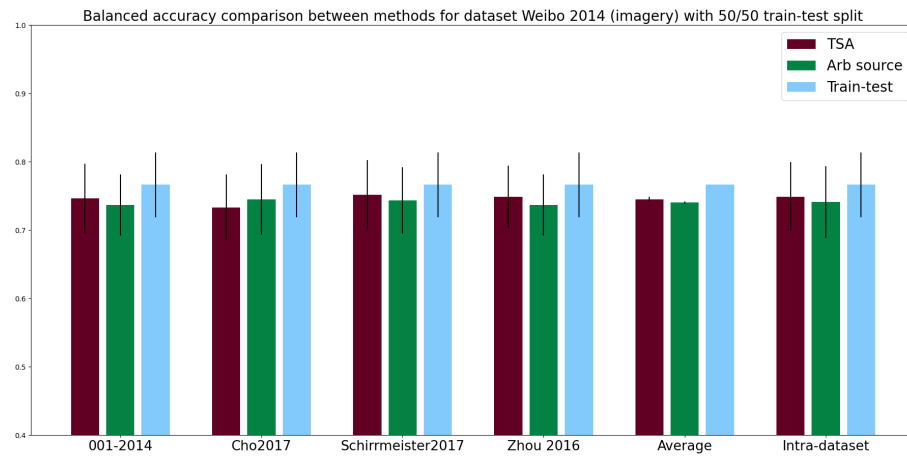


FIGURE A.77 – Précision équilibrée inter-configurations pour la base de donnée Weibo 2014 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Zhou 2016 (MI)

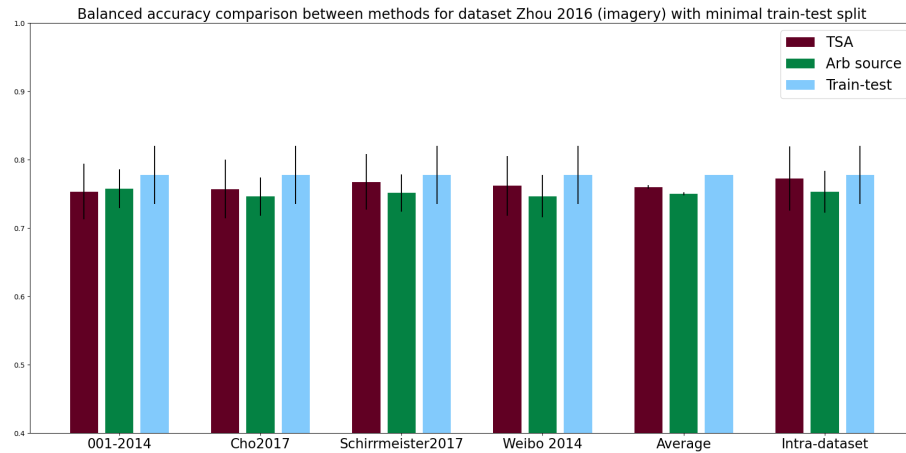


FIGURE A.78 – Précision équilibrée inter-configurations pour la base de donnée Zhou 2016 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

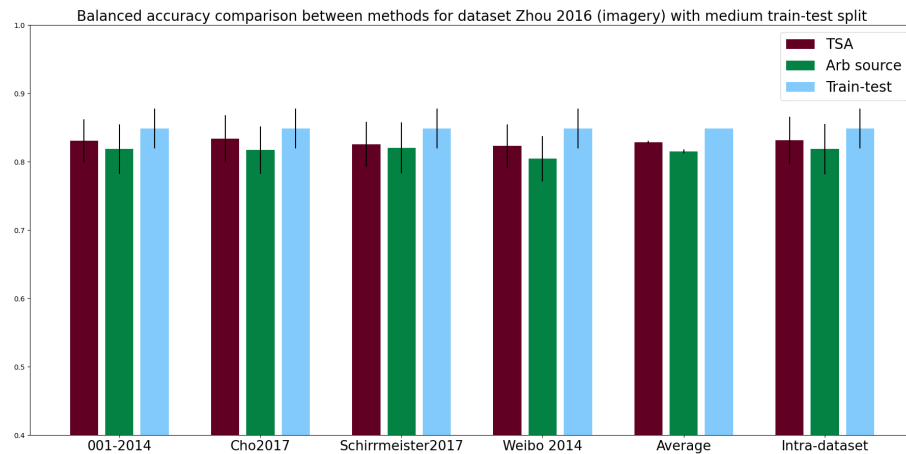


FIGURE A.79 – Précision équilibrée inter-configurations pour la base de donnée Zhou 2016 avec un fractionnement moyen. Les résultats sont présentées comme en figure 3.4.

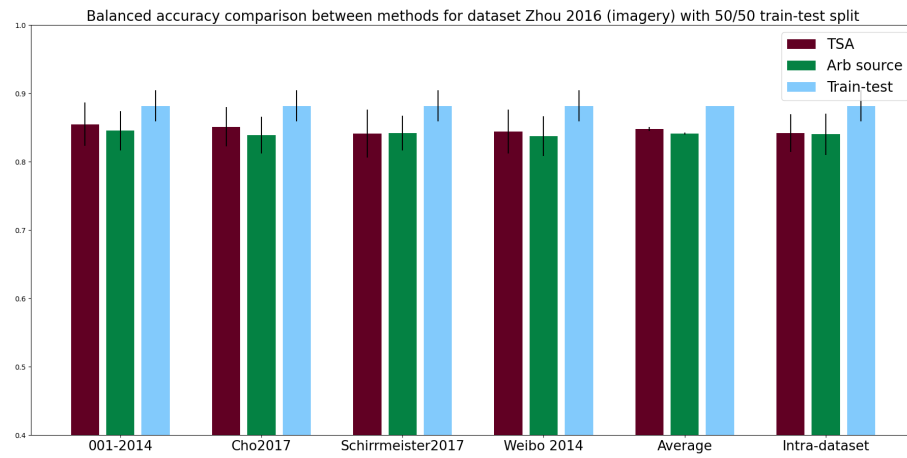


FIGURE A.80 – Précision équilibrée inter-configurations pour la base de donnée Zhou 2016 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données Lee2019_SSVEP (SSVEP)

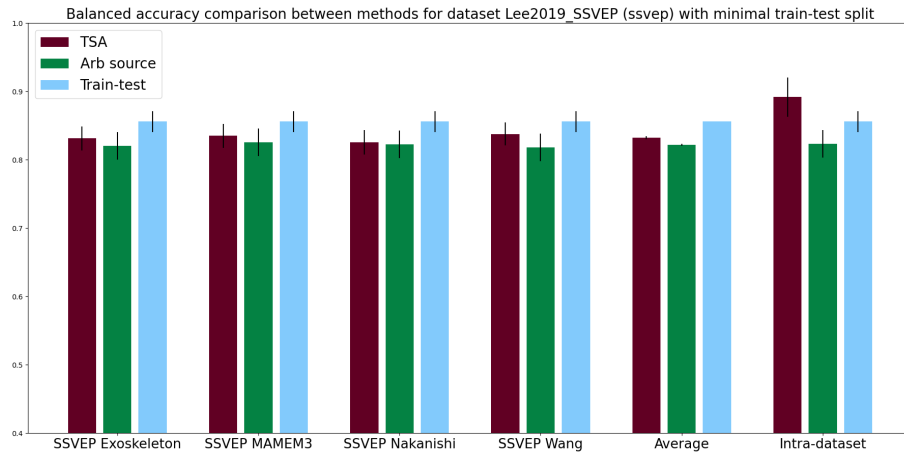


FIGURE A.81 – Précision équilibrée inter-configurations pour la base de donnée Lee2019_SSVEP avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

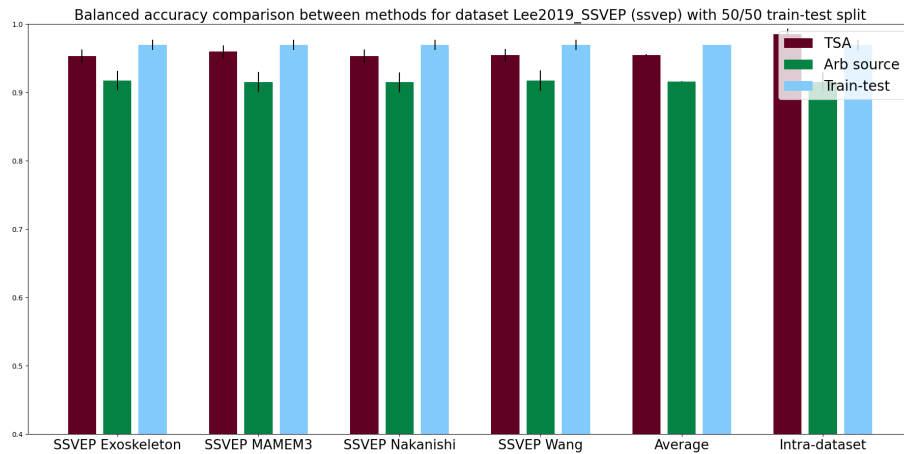


FIGURE A.82 – Précision équilibrée inter-configurations pour la base de donnée Lee2019_SSVEP avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données SSVEP Exoskeleton (SSVEP)

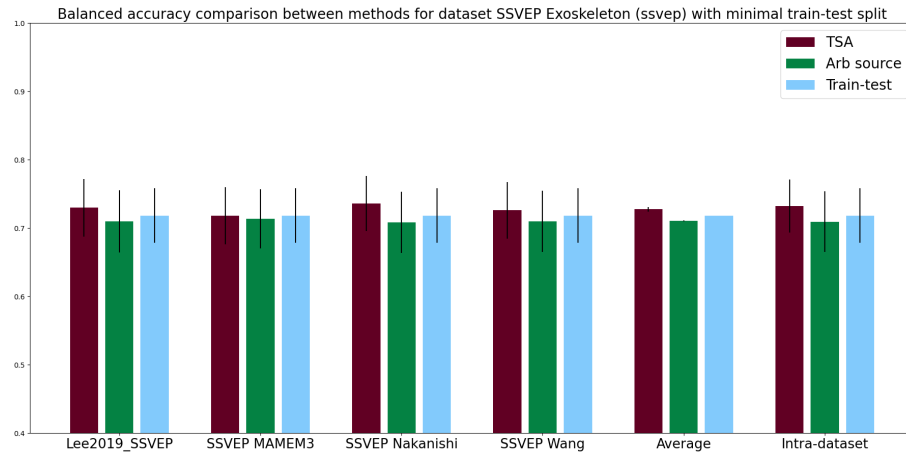


FIGURE A.83 – Précision équilibrée inter-configurations pour la base de donnée SSVEP Exoskeleton avec un fractionnement minimal. Les résultats sont présentés comme en figure 3.4.

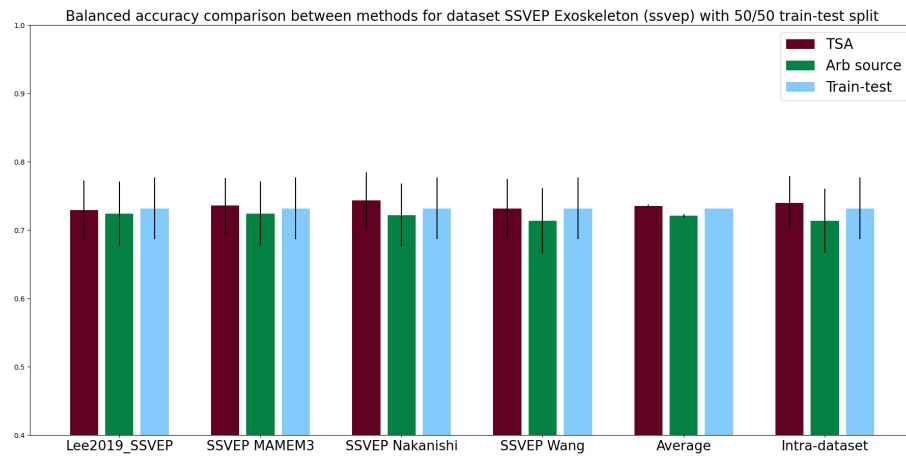


FIGURE A.84 – Précision équilibrée inter-configurations pour la base de donnée SSVEP Exoskeleton avec un fractionnement 50/50. Les résultats sont présentés comme en figure 3.4.

Base de données SSVEP MAMEM3 (SSVEP)

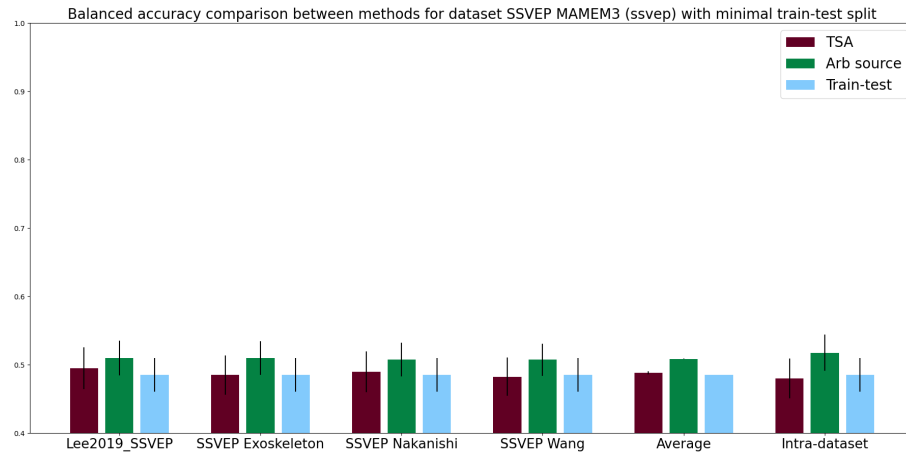


FIGURE A.85 – Précision équilibrée inter-configurations pour la base de donnée SSVEP MAMEM3 avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

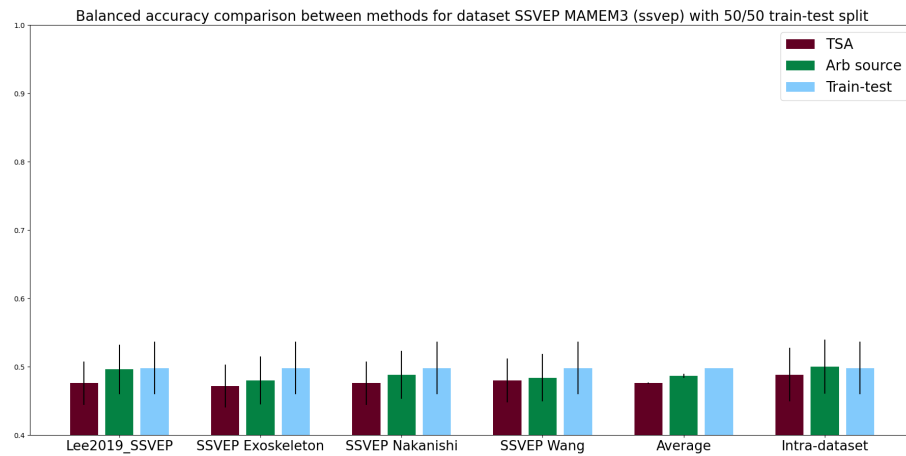


FIGURE A.86 – Précision équilibrée inter-configurations pour la base de donnée SSVEP MAMEM3 avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données SSVEP Nakanishi (SSVEP)

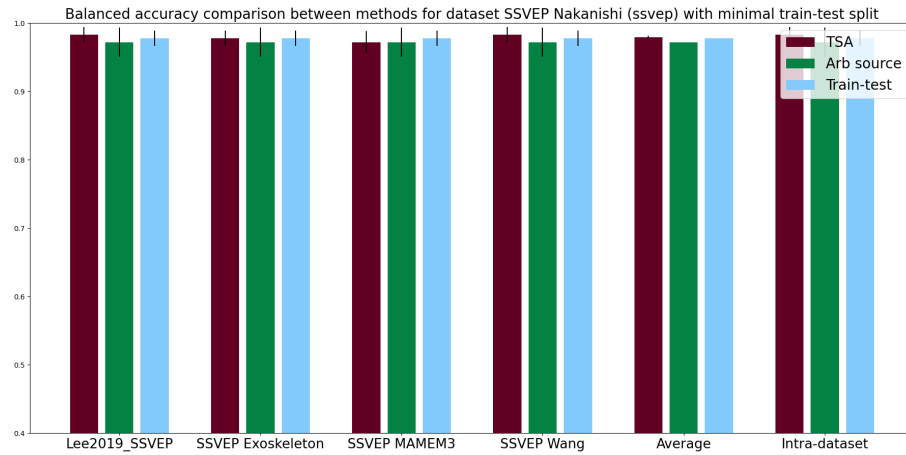


FIGURE A.87 – Précision équilibrée inter-configurations pour la base de donnée SSVEP Nakanishi avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

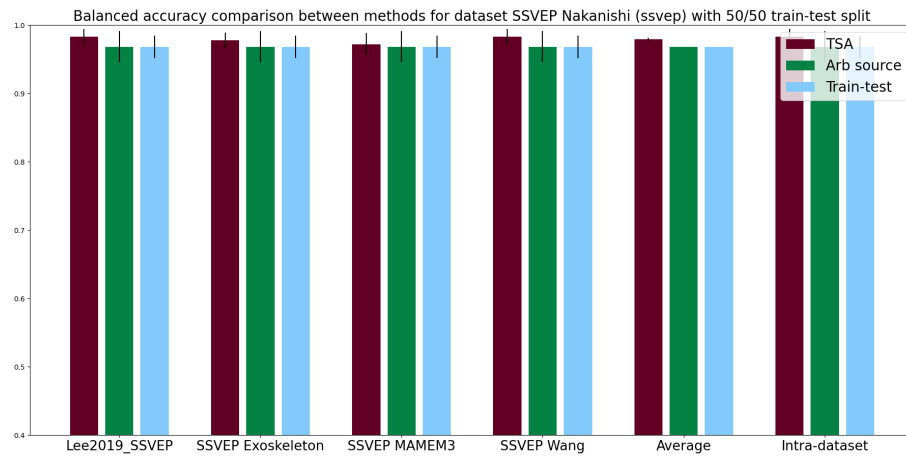


FIGURE A.88 – Précision équilibrée inter-configurations pour la base de donnée SSVEP Nakanishi avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Base de données SSVEP Wang (SSVEP)

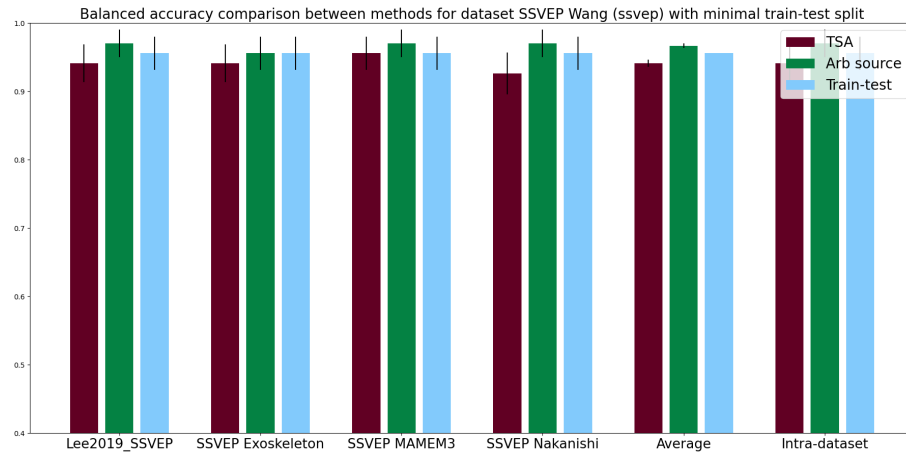


FIGURE A.89 – Précision équilibrée inter-configurations pour la base de donnée SSVEP Wang avec un fractionnement minimal. Les résultats sont présentées comme en figure 3.4.

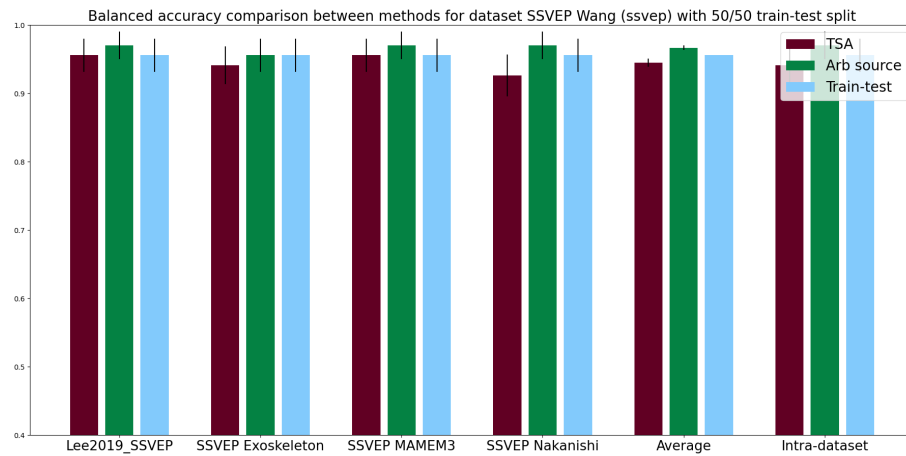


FIGURE A.90 – Précision équilibrée inter-configurations pour la base de donnée SSVEP Wang avec un fractionnement 50/50. Les résultats sont présentées comme en figure 3.4.

Bibliographie

- [1] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley Interdisciplinary Reviews : Computational Statistics*, pages 1–47, 2010.
- [2] Pierre Ablin, Jean Francois Cardoso, and Alexandre Gramfort. Faster independent component analysis by preconditioning with Hessian approximations. *IEEE Transactions on Signal Processing*, 66(15) :4040–4049, 2018.
- [3] Rafeed Alkawadri. Brain–Computer Interface (BCI) Applications in Mapping of Epileptic Brain Networks Based on Intracranial-EEG : An Update. *Frontiers in Neuroscience*, 13(March) :1–13, 2019.
- [4] P Aricò, F Aloise, F Schettini, S Salinari, D Mattia, and F Cincotti. Influence of P300 latency jitter on event related potential-based brain-computer interface performance. *Journal of Neural Engineering*, 11(3), 2014.
- [5] Alexandre Barachant. Commande robuste d’un effecteur par une interface cerveau machine EEG asynchrone. *These. Université de Grenoble*, 2012.
- [6] Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Riemannian geometry applied to BCI classification. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6365 LNCS :629–636, 2010.
- [7] Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Multiclass brain-computer interface classification by Riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4) :920–928, 2012.
- [8] 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, 2013.
- [9] Alexandre Barachant and Marco Congedo. A Plug&Play P300 BCI Using Information Geometry. *arXiv preprint arXiv :1409.0107*, pages 1–9, 2014.
- [10] Alim Louis Benabid, Thomas Costecalde, Andrey Eliseyev, Guillaume Charvet, Alexandre Verney, Serpil Karakas, Michael Foerster, Aurélien Lambert, Boris Morinière, Neil Abroug, Marie Caroline Schaeffer, Alexandre Moly, Fabien Sauter-Starace, David Ratel, Cecile Moro, Napoleon Torres-Martinez, Lilia Langar, Manuela Oddoux, Mircea Polosan,

- Stephane Pezzani, Vincent Auboiron, Tetiana Aksenova, Corinne Mes-tais, and Stephan Chabardes. An exoskeleton controlled by an epidural wireless brain-machine interface in a tetraplegic patient : a proof-of-concept demonstration. *The Lancet Neurology*, 18(12) :1112–1122, 2019.
- [11] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B. Shah. Julia : A fresh approach to numerical computing. *SIAM Review*, 59(1) :65–98, 2017.
- [12] Rajendra Bhatia. *Positive Definite Matrices*. Princeton University Press, Princeton, 2007.
- [13] Niels Birbaumer. Breaking the silence : Brain-computer interfaces (BCI) for communication and motor control. *Psychophysiology*, 43(6) :517–532, 2006.
- [14] Benjamin Blankertz, Ryota Tomioka, Steven Lemm, Motoaki Kawanabe, and Klaus-robert Muller. Optimizing Spatial filters for Robust EEG Single-Trial Analysis. *IEEE Signal Processing Magazine*, 25(1) :41–56, 2008.
- [15] Alexandre Bleuzé, Jérémie Mattout, and Marco Congedo. Tangent space alignment : Transfer learning for Brain-Computer Interface. *Frontiers in Human Neuroscience*, 16, 2022.
- [16] Laurent Bougrain, Sebastien Rimbart, Pedro L.C. Rodrigues, Geoffrey Canron, and Fabien Lotte. Guidelines to use transfer learning for motor imagery detection : An experimental study. *International IEEE/EMBS Conference on Neural Engineering, NER*, 2021-May(3) :5–8, 2021.
- [17] Richard G. Brereton and Gavin R. Lloyd. Support Vector Machines for classification and regression. *Analyst*, 135(2) :230–267, 2010.
- [18] R. Bro, E. Acar, and Tamara G. Kolda. Resolving the sign ambiguity in the singular value decomposition. *Journal of Chemometrics*, 22(2) :135–140, 2008.
- [19] Alexander J Casson, Mohammed Abdulaal, Meera Dulabh, Siddharth Kohli, Sammy Krachunov, and Eleanor Trimble. Electroencephalogram. *Seamless healthcare monitoring : advancements in wearable, attachable, and invisible devices*, pages 45–81, 2018.
- [20] G Charvet, M Foerster, S Filipe, J Porcherot, JF Beche, R Guillemaud, P Audebert, G Regis, B Zongo, S Robinet, et al. WIMAGINE : A wireless, low power, 64-channel ECoG recording platform for implantable BCI applications. In *2011 5th International IEEE/EMBS Conference on Neural Engineering*, pages 356–359. IEEE, 2011.
- [21] 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, 2010.
- [22] Hohyun Cho, Minkyu Ahn, Sangtae Ahn, Moonyoung Kwon, and Sung Chan Jun. EEG datasets for motor imagery brain-computer interface. *GigaScience*, 6(7) :1–8, 2017.

- [23] Pierre Comon. Independent Component Analysis, a new concept ? *Signal processing*, 36(3) :287–314, 1994.
- [24] Marco Congedo. EEG Source Analysis. *HDR. Université de Grenoble*, 2013.
- [25] Marco Congedo, Alexandre Barachant, and Anton Andreev. A New Generation of Brain-Computer Interface Based on Riemannian Geometry. *arXiv preprint arXiv :1310.8115.*, 2013.
- [26] 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.
- [27] Marco Congedo, Matthieu Goyat, Nicolas Tarrin, Gelu Ionescu, Léo Varnet, Bertrand Rivet, Ronald Phlypo, Nisrine Jrad, Michael Acquadro, and Christian Jutten. Brain Invaders : a prototype of an open-source P300-based video game working with the OpenViBE platform. *5th International Brain-Computer Interface Conference*, pages 280–283, 2011.
- [28] Marco Congedo, Pedro Luiz, Coelho Rodrigues, Christian Jutten, M Congedo, P L C Rodrigues, and C Jutten. The Riemannian Minimum Distance to Means Field Classifier. *BCI 2019-8th International Brain-Computer Interface Conference*, 2019.
- [29] Denis Alexander Engemann, Oleh Kozynets, David Sabbagh, Guillaume Lemaitre, Gaël Varoquaux, Franziskus Liem, and Alexandre Gramfort. Combining magnetoencephalography with magnetic resonance imaging enhances learning of surrogate-biomarkers. *eLife*, 9 :1–33, 2020.
- [30] Fatemeh Fahimi, Zhuo Zhang, Wooi Boon Goh, Tih Shi Lee, Kai Keng Ang, and Cuntai Guan. Inter-subject transfer learning with an end-to-end deep convolutional neural network for EEG-based BCI. *Journal of Neural Engineering*, 16(2), 2019.
- [31] Josef Faller, Carmen Vidaurre, Teodoro Solis-Escalante, Christa Neuper, and Reinhold Scherer. Autocalibration and recurrent adaptation : Towards a plug and play online ERD-BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(3) :313–319, 2012.
- [32] Brian Fiani, Taylor Reardon, Benjamin Ayres, David Cline, and Sarah R Sitto. An Examination of Prospective Uses and Future Directions of Neuralink : The Brain-Machine Interface. *Cureus*, 13(3), 2021.
- [33] Pierre Fillard, Vincent Arsigny, Nicholas Ayache, and Xavier Pennec. A riemannian framework for the processing of tensor-valued images. In *International Workshop on Deep Structure, Singularities, and Computer Vision*, pages 112–123. Springer, 2005.
- [34] Keinosuke Fukunaga. *Introduction to statistical pattern recognition*. Elsevier, 1972.
- [35] Moritz Grosse-Wentrup and Martin Buss. Multiclass common spatial patterns and information theoretic feature extraction. *IEEE Transactions on Biomedical Engineering*, 55(8) :1991–2000, 2008.

- [36] Moritz Grosse-Wentrup, Christian Liefhold, Klaus Gramann, and Martin Buss. Beamforming in noninvasive brain-computer interfaces. *IEEE Transactions on Biomedical Engineering*, 56(4) :1209–1219, 2009.
- [37] Moritz Grosse-Wentrup and Bernhard Schölkopf. A review of performance variations in smr-based brain-computer interfaces (bcis). *Brain-Computer Interface Research : A State-of-the-Art Summary*, pages 39–51, 2013.
- [38] Christoph Guger, Shahab Daban, Eric Sellers, Clemens Holzner, Gunther Krausz, Roberta Carabalona, Furio Gramatica, and Guenter Edlinger. How many people are able to control a P300-based brain-computer interface (BCI) ? *Neuroscience Letters*, 462(1) :94–98, 2009.
- [39] Ulrich Hoffmann, Jean Marc Vesin, Touradj Ebrahimi, and Karin Diersens. An efficient P300-based brain-computer interface for disabled subjects. *Journal of Neuroscience Methods*, 167(1) :115–125, 2008.
- [40] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, pages 65–70, 1979.
- [41] Alan Julian Izenman. Linear discriminant analysis. In *Modern multivariate statistical techniques : regression, classification, and manifold learning*, pages 237–280. Springer, 2013.
- [42] Vinay Jayaram, Morteza Alamgir, Yasemin Altun, Bernhard Scholkopf, and Moritz Grosse-Wentrup. Transfer Learning in Brain-Computer Interfaces. *IEEE Computational Intelligence Magazine*, 11(1) :20–31, 2016.
- [43] Vinay Jayaram and Alexandre Barachant. MOABB : Trustworthy algorithm benchmarking for BCIs. *Journal of Neural Engineering*, 15(6), 2018.
- [44] Emmanuel K Kalunga, Sylvain Chevallier, and Quentin Barthelemy. Using Riemannian geometry for SSVEP-based Brain Computer Interface. *arXiv preprint arXiv :1501.03227*, 2015.
- [45] Emmanuel K Kalunga, Sylvain Chevallier, and Quentin Barthélemy. Transfer Learning for SSVEP-based BCI using Riemannian similarities between users. *European Signal Processing Conference*, 2018-Sept :1685–1689, 2018.
- [46] Alexander Y Kaplan, Sergei L Shishkin, Ilya P Ganin, Ivan A Basyul, and Alexander Y Zhigalov. Adapting the P300-based brain-computer interface for gaming : A review. *IEEE Transactions on Computational Intelligence and AI in Games*, 5(2) :141–149, 2013.
- [47] Zoltan J. Koles, Michael S. Lazar, and Steven Z. Zhou. Spatial patterns underlying population differences in the background EEG. *Brain Topography*, 2(4) :275–284, 1990.
- [48] Louis Korczowski, Martine Cederhout, Anton Andreev, Grégoire Cattan, Pedro Luiz Coelho Rodrigues, Violette Gautheret, and Marco Congedo. Brain Invaders calibration-less P300-based BCI with modulation of flash duration Dataset (bi2015a). Technical report, GIPSA-lab, 2019.

- [49] Louis Korczowski, Martine Cederhout, Anton Andreev, Grégoire Cattan, Pedro Luiz Coelho Rodrigues, Violette Gautheret, and Marco Congedo. Brain invaders cooperative versus competitive : Multi-user p300-based brain-computer interface dataset (bi2015b). Technical report, GIPSA-lab, 2019.
- [50] Louis Korczowski, Ekaterina Ostaschenko, Anton Andreev, Grégoire Cattan, Pedro Luiz Coelho Rodrigues, Violette Gautheret, and Marco Congedo. Brain Invaders calibration-less P300-based BCI using dry EEG electrodes Dataset (bi2014a). Technical report, GIPSA-lab, 2019.
- [51] Louis Korczowski, Ekaterina Ostaschenko, Anton Andreev, Grégoire Cattan, Pedro Luiz Coelho Rodrigues, Violette Gautheret, and Marco Congedo. Brain invaders solo versus collaboration : Multi-user p300-based brain-computer interface dataset (bi2014b). Technical report, GIPSA-lab, 2019.
- [52] Michael P. LaValley. Logistic regression. *Circulation*, 117(18) :2395–2399, 2008.
- [53] Huiling Le. Estimation of Riemannian Barycentres. *LMS Journal of Computation and Mathematics*, 7 :193–200, 2004.
- [54] Anatole Lécuyer, Fabien Lotte, Richard B Reilly, Robert Leeb, Michitaka Hirose, and Mel Slater. Brain-computer interfaces, virtual reality, and videogames. *Computer*, 41(10) :66–72, 2008.
- [55] Min Ho Lee, O Yeon Kwon, Yong Jeong Kim, Hong Kyung Kim, Young Eun Lee, John Williamson, Siamac Fazli, and Seong Whan Lee. EEG dataset and OpenBMI toolbox for three BCI paradigms : An investigation into BCI illiteracy. *GigaScience*, 8(5) :1–16, 2019.
- [56] Robert Leeb, Felix Lee, Claudia Keinrath, Reinhold Scherer, Horst Bischof, and Gert Pfurtscheller. Brain-computer communication : Motivation, aim, and impact of exploring a virtual apartment. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(4) :473–482, 2007.
- [57] Feng Li, Yi Xia, Fei Wang, Dengyong Zhang, Xiaoyu Li, and Fan He. Transfer learning algorithm of P300-EEG signal based on XDAWN spatial filter and riemannian geometry classifier. *Applied Sciences (Switzerland)*, 10(5), 2020.
- [58] Jun Ma, Banghua Yang, Wenzheng Qiu, Yunzhe Li, Shouwei Gao, and Xinxing Xia. A large EEG dataset for studying cross-session variability in motor imagery brain-computer interface. *Scientific Data*, 9(1) :1–9, 2022.
- [59] Gal Maman, Or Yair, Danny Eytan, and Ronen Talmon. Domain Adaptation Using Riemannian Geometry of SPD Matrices. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2019-May :4464–4468, 2019.
- [60] Dennis J. McFarland, Laurie A. Miner, Theresa M. Vaughan, and Jonathan R. Wolpaw. Mu and beta rhythm topographies during motor imagery and actual movements. *Brain Topography*, 12(3) :177–186, 2000.

- [61] 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.
- [62] Masaki Nakanishi, Yijun Wang, Yu Te Wang, and Tzyy Ping Jung. A comparison study of canonical correlation analysis based methods for detecting steady-state visual evoked potentials. *PLoS ONE*, 10(10) :1–18, 2015.
- [63] Frank Nielsen and Rajendra Bhatia. *Matrix information geometry*. Springer, 2013.
- [64] Vangelis P Oikonomou, Georgios Liaros, Kostantinos Georgiadis, Elisavet Chatzilari, Katerina Adam, Spiros Nikolopoulos, and Ioannis Kompatsiaris. Comparative evaluation of state-of-the-art algorithms for ssvep-based bcis. *arXiv preprint arXiv :1602.00904*, 2016.
- [65] Serafeim Perdikis, Luca Tonin, Sareh Saeedi, Christoph Schneider, and José del R. Millán. The Cybathlon BCI race : Successful longitudinal mutual learning with two tetraplegic users. *PLoS Biology*, 16(5) :1–17, 2018.
- [66] Gert Pfurtscheller and Christa Neuper. Motor imagery activates primary sensorimotor area in humans. *Neuroscience Letters*, 239(2-3) :65–68, 1997.
- [67] PhysioToolkit PhysioBank. Physionet : components of a new research resource for complex physiologic signals. *Circulation*, 101(23) :e215–e220, 2000.
- [68] Herbert Ramoser, Johannes Müller-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.
- [69] Angela Riccio, Luca Simione, Francesca Schettini, Alessia Pizzimenti, Maurizio Inghilleri, Marta Olivetti Belardinelli, Donatella Mattia, and Febo Cincotti. Attention and P300-based BCI performance in people with amyotrophic lateral sclerosis. *Frontiers in Human Neuroscience*, 7(NOV) :1–9, 2013.
- [70] Pedro Luiz Coelho Rodrigues, Marco Congedo, and Christian Jutten. Dimensionality transcending : a method for merging BCI datasets with different dimensionalities. *IEEE Transactions on Biomedical Engineering*, 68(2) :673–684, 2020.
- [71] Pedro Luiz Coelho Rodrigues, Christian Jutten, and Marco Congedo. Riemannian procrustes analysis : Transfer learning for brain–computer interfaces. *IEEE Transactions on Biomedical Engineering*, 66(8) :2390–2401, 2018.
- [72] Arunabha M. Roy. Adaptive transfer learning-based multiscale feature fused deep convolutional neural network for EEG MI multiclassification in brain–computer interface. *Engineering Applications of Artificial Intelligence*, 116(Mi) :1–52, 2022.

- [73] Simanto Saha and Mathias Baumert. Intra- and Inter-subject Variability in EEG-Based Sensorimotor Brain Computer Interface : A Review. *Frontiers in Computational Neuroscience*, 13, jan 2020.
- [74] G. Schalk, D.J. McFarland, T. Hinterberger, N. Birbaumer, and J.R. Wolpaw. BCI2000 : A General-Purpose Brain-Computer Interface (BCI) System. *IEEE Transactions on Biomedical Engineering*, 51(6) :1034–1043, jun 2004.
- [75] Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggenberger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for EEG decoding and visualization. *Human Brain Mapping*, 38(11) :5391–5420, 2017.
- [76] David Steyrl, Reinhold Scherer, Josef Faller, and Gernot R. Müller-Putz. Random forests in non-invasive sensorimotor rhythm brain-computer interfaces : A practical and convenient non-linear classifier. *Biomedizinische Technik*, 61(1) :77–86, 2016.
- [77] Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. *Proceedings of the AAAI conference on artificial intelligence*, 30(1), 2016.
- [78] Barbara E. Swartz. The advantages of digital over analog recording techniques. *Electroencephalography and Clinical Neurophysiology*, 106(2) :113–117, 1998.
- [79] Michael Tangermann, Klaus Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J Miller, Gernot R Müller-Putz, Guido Nolte, Gert Pfurtscheller, Hubert Preissl, Gerwin Schalk, Alois Schlögl, Carmen Vidaurre, Stephan Waldert, and Benjamin Blankertz. Review of the BCI competition IV. *Frontiers in Neuroscience*, 6(July) :1–31, 2012.
- [80] Ryota Tomioka, Kazuyuki Aihara, and Klaus-Robert Müller. Logistic regression for single trial EEG classification. *Advances in neural information processing systems*, 19, 2006.
- [81] E Vaineu, A Barachant, A Andreev, P Rodrigues, G Cattan, and M Congedo. Brain Invaders Adaptive versus Non-Adaptive P300 Brain-Computer Interface dataset. Technical Report April, GIPSA-lab, 2019.
- [82] Gijbrecht Van Veen, Alexandre Barachant, Anton Andreev, Grégoire Cattan, Pedro Coelho Rodrigues, and Marco Congedo. Building Brain Invaders : EEG data of an experimental validation. *arXiv preprint arXiv :1905.05182*, 2019.
- [83] Mariska J. Vansteensel, Elmar G.M. Pels, Martin G. Bleichner, Mariana P. Branco, Timothy Denison, Zachary V. Freudenberg, Peter Gosselaar, Sacha Leinders, Thomas H. Ottens, Max A. Van Den Boom, Peter C. Van Rijen, Erik J. Aarnoutse, and Nick F. Ramsey. Fully Implanted Brain-Computer Interface in a Locked-In Patient with ALS. *New England Journal of Medicine*, 375(21) :2060–2066, 2016.

- [84] J. J. Vidal. Toward direct brain-computer communication. *Annual review of biophysics and bioengineering*, 2 :157–180, 1973.
- [85] Zitong Wan, Rui Yang, Mengjie Huang, Nianyin Zeng, and Xiaohui Liu. A review on transfer learning in EEG signal analysis. *Neurocomputing*, 421 :1–14, 2021.
- [86] Peitao Wang, Jun Lu, Bin Zhang, and Zeng Tang. A review on transfer learning for brain-computer interface classification. *2015 5th International Conference on Information Science and Technology, ICIST 2015*, pages 315–322, 2015.
- [87] Yijun Wang, Xiaogang Chen, Xiaorong Gao, and Shangkai Gao. A benchmark dataset for SSVEP-based brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10) :1746–1752, 2016.
- [88] Xiaoxi Wei, A. Aldo Faisal, Moritz Grosse-Wentrup, Alexandre Gramfort, Sylvain Chevallier, Vinay Jayaram, Camille Jeunet, Stylianos Bakas, Siegfried Ludwig, Konstantinos Barmpas, Mehdi Bahri, Yannis Pagnakis, Nikolaos Laskaris, Dimitrios A. Adamos, Stefanos Zafeiriou, William C. Duong, Stephen M. Gordon, Vernon J. Lawhern, Maciej Śliwowski, Vincent Rouanne, and Piotr Tempczyk. 2021 BEETL Competition : Advancing Transfer Learning for Subject Independence & Heterogenous EEG Data Sets. *Proceedings of Machine Learning Research*, 176 :205–219, 2022.
- [89] Jonathan R Wolpaw, Niels Birbaumer, Dennis J McFarland, Gert Pfurtscheller, and Theresa M Vaughan. Brain-computer interfaces for communication and control. *Clinical neurophysiology*, 113(6) :767–791, 2002.
- [90] Kyungho Won, Moonyoung Kwon, Minkyu Ahn, and Sung Chan Jun. EEG Dataset for RSVP and P300 Speller Brain-Computer Interfaces. *Scientific Data*, 9(1) :1–11, 2022.
- [91] Dongrui Wu, Xue Jiang, and Ruimin Peng. Transfer learning for motor imagery based brain-computer interfaces : A tutorial. *Neural Networks*, 153 :235–253, 2022.
- [92] Dongrui Wu, Vernon J Lawhern, W David Hairston, and Brent J Lance. Switching eeg headsets made easy : Reducing offline calibration effort using active weighted adaptation regularization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(11) :1125–1137, 2016.
- [93] Dongrui Wu, Yifan Xu, and Bao Liang Lu. Transfer Learning for EEG-Based Brain-Computer Interfaces : A Review of Progress Made since 2016. *IEEE Transactions on Cognitive and Developmental Systems*, 14(1) :4–19, 2020.
- [94] Jiachen Xu, Alex Markham, Anja Meunier, Philipp Raggam, and Moritz Grosse-Wentrup. Distance Covariance : A Nonlinear Extension of Riemannian Geometry for EEG-based Brain-Computer Interfacing. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, pages 2000–2005, 2021.

- [95] Lichao Xu, Minpeng Xu, Yufeng Ke, Xingwei An, Shuang Liu, and Dong Ming. Cross-Dataset Variability Problem in EEG Decoding With Deep Learning. *Frontiers in Human Neuroscience*, 14(April) :1–10, 2020.
- [96] Or Yair, Mirela Ben-Chen, and Ronen Talmon. Parallel Transport on the Cone Manifold of SPD Matrices for Domain Adaptation. *IEEE Transactions on Signal Processing*, 67(7) :1797–1811, 2019.
- [97] Or Yair, Felix Dietrich, Ronen Talmon, and Ioannis G Kevrekidis. Domain adaptation with optimal transport on the manifold of spd matrices. *arXiv preprint arXiv :1906.00616*, 2019.
- [98] Florian Yger, Maxime Berar, and Fabien Lotte. Riemannian Approaches in Brain-Computer Interfaces : A Review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10) :1753–1762, 2017.
- [99] Weibo Yi, Shuang Qiu, Kun Wang, Hongzhi Qi, Lixin Zhang, Peng Zhou, Feng He, and Dong Ming. Evaluation of EEG oscillatory patterns and cognitive process during simple and compound limb motor imagery. *PLoS ONE*, 9(12) :1–19, 2014.
- [100] Jiahui Ying, Qingguo Wei, and Xichen Zhou. Riemannian geometry-based transfer learning for reducing training time in c-VEP BCIs. *Scientific Reports*, 12(1) :1–15, 2022.
- [101] Paolo Zanini, Marco Congedo, Christian Jutten, Salem Said, and Yannick Berthoumieu. Transfer Learning : A Riemannian Geometry Framework with Applications to Brain-Computer Interfaces. *IEEE Transactions on Biomedical Engineering*, 65(5) :1107–1116, 2018.
- [102] Dmitri V. Zaykin. Optimally weighted Z-test is a powerful method for combining probabilities in meta-analysis. *Journal of Evolutionary Biology*, 24(8) :1836–1841, 2011.
- [103] Ruilong Zhang, Qun Zong, Liqian Dou, Xinyi Zhao, Yifan Tang, and Zhiyu Li. Hybrid deep neural network using transfer learning for EEG motor imagery decoding. *Biomedical Signal Processing and Control*, 63(June 2020) :102144, 2021.
- [104] Bangyan Zhou, Xiaopei Wu, Zhao Lv, Lei Zhang, and Xiaojin Guo. A fully automated trial selection method for optimization of motor imagery based Brain-Computer interface. *PLoS ONE*, 11(9) :1–20, 2016.
- [105] Rümeyşa İnce, Saliha Seda Adanır, and Fatma Sevmez. The inventor of electroencephalography (EEG) : Hans Berger (1873–1941). *Child’s Nervous System*, 37(9) :2723–2724, 2021.