

THÈSE DE L'UNIVERSITÉ DE LYON

Délivrée par

UNIVERSITÉ CLAUDE BERNARD LYON 1

DIPLOÔME DE DOCTORAT

Ma spécialité

DÉCODAGE DES INTENTIONS ET DES REPRÉSENTATIONS MOTRICES CHEZ L'HOMME : ANALYSE MULTI-ÉCHELLE ET APPLICATION AUX INTERFACES CERVEAU-MACHINE

par

Etienne Combrisson

Thèse soutenue le 09/2016 devant le jury composé de :

M ^{me}	ERIKA RATÉ	Université à la Menthe	(Rapporteur)
M.	JACQUES OUILLE	Université à la Fraise	(Rapporteur)
M.	HENRI ZOTO	Laboratoire laborieux	(Rapporteur)
M.	JEAN FILE	Indienne	(Directeur)
	etc.		

*À mes parents, Isabelle et Didier Prévot,
Merci*

Titre Décodage des intentions et des représentations motrices chez l'homme : analyse multi-échelle et application aux interfaces cerveau-machine

Résumé Le résumé en français (\approx 1000 caractères)

Mots-clés Les mots-clés en français

Title Le titre en anglais

Abstract Le résumé en anglais (\approx 1000 caractères)

Keywords Les mots-clés en anglais

REMERCIEMENTS

J' voudrais tout d'abord exprimer mes plus profonds remerciements à...
AHÂÂÂH!

Je conclurai en remerciant de tout cœur (l'être aimé).

Montréal, le 22 juin 2017.

TABLE DES MATIÈRES

TABLE DES MATIÈRES	vi
LISTE DES FIGURES	ix
NOTATIONS	1
I Introduction	3
1 DÉCODAGE DE L'ACTIVITÉ CÉRÉBRALE	5
1.1 LES INTERFACE CERVEAU-MACHINE	5
1.1.1 Définition et objectifs	5
1.1.2 Techniques d'acquisition de l'activité neuronale	12
1.1.3 ICM synchrones/asynchrones et invasives/non-invasives	18
1.1.4 Signaux physiologiques pour le contrôle d'une ICM	19
1.2 Data-mining EN NEUROSCIENCES	27
1.2.1 Exploration des données	27
1.2.2 Outils de validation	27
2 ICM ET NEUROPHYSIOLOGIE	29
2.1 BASES PHYSIOLOGIQUES LIÉES À LA MOTRICITÉ	29
2.1.1 Notions sur le cortex	29
2.1.2 L'activité rythmique et lien avec la motricité	30
2.2 DÉCODAGE DIRECTIONNEL DES MEMBRES SUPÉRIEURS	31
2.2.1 Décodage directionnel	32
2.2.2 Prédiction continu de la cinétique du mouvement	33
3 OBJECTIFS DE LA THÈSE	35
4 MÉTHODOLOGIE	37
4.1 EXTRACTION DES FEATURES	37
4.1.1 Pré-requis	37
4.1.2 Puissance spectrale	39
4.1.3 Phase	40
4.1.4 Phase-amplitude coupling	40
4.2 APPRENTISSAGE SUPERVISÉ	46
4.2.1 Labellisation et apprentissage	47
4.2.2 <i>Training, testing</i> et validation-croisée	47
4.2.3 Classificateurs	49
4.2.4 Évaluation de la performance de décodage	51

4.2.5	Seuil de chance et évaluation statistique de la performance de décodage	52
4.2.6	Du single au multi-features	52
4.2.7	Généralisation temporelle	55
5	DÉVELOPPEMENTS INFORMATIQUES	59
5.1	CHOIX DU LANGAGE : PYTHON	59
5.2	PAQUETS DÉVELOPPÉS DURANT CETTE THÈSE	60
5.2.1	ipywks	60
5.2.2	Brainpipe	61
5.2.3	Visbrain	64
6	DONNÉES EXPÉRIMENTALES	69
6.1	DONNÉES INTRACRÂNIENNES	69
6.1.1	Acquisition	69
6.1.2	Avantages et limitations	69
6.1.3	Inspection visuelle	71
6.1.4	Prétraitements	71
6.2	DONNÉES D'ÉTUDE	71
6.2.1	Données <i>Center-out</i>	72
6.2.2	Autres données	74
6.3	DELAYED TASK : PROTOCOLE EXPÉRIMENTAL	74
II	Étude 1 : Niveau de chance et évaluation statistique des résultats de classification par apprentissage supervisé	75
III	Étude 2 : Encodage de l'intention et de l'exécution motrice	89
IV	Étude 3 : Décodage des directions de mouvement pendant et avant l'exécution de mouvement de membres supérieurs	107
V	Étude 4 : Tensorpac, logiciel Python de calcul de Phase-Amplitude Coupling	111
VI	Étude 5 : Visbrain, ensemble d'outils de visualisation de données neuroscientifiques	133
VII	Étude 6 : Sleep, visualisation de données polysomnographiques et scorage	161
CONCLUSION GÉNÉRALE		189
A	ANNEXES	190
A.1	CARTES DES INTERFACES CERVEAU-MACHINE (GRAIMANN ET AL., 2009)	191
A.2	JEUX DE DONNÉES EN LIBRE ACCÈS (<i>BCI competition</i>)	192

A.3 COMPARATIF DE MÉTHODES PAC (TORT ET AL., 2010)	193
A.4 PIPELINE STANDARD DE CLASSIFICATION	194
A.5 COMPARATIF DE CLASSIFIEURS (PEDREGOSA ET AL., 2011)	196
A.6 EXEMPLE DE SCHÉMA D'IMPLANTATION	198
BIBLIOGRAPHIE	199

LISTE DES FIGURES

1.1	Schéma d'une Interface Cerveau-Machine (Pfurtscheller et al., 2008)	9
1.2	Paralysie causée par la SLA et accident vasculaire cérébral (Kübler et al., 2001)	10
1.3	BCI competition et visibilité Tangermann et al. (2012)	12
1.4	ICM utilisant des micro-électrodes (Hochberg et al., 2006, 2012)	14
1.5	Micro et macro-électrodes pour l'Électrocorticographie (Schalk and Leuthardt, 2011, Yanagisawa et al., 2012a)	15
1.6	Production des champs électriques et magnétiques. Recueil EEG et MEG, limitations et complémentarité (Sato et al., 1991)	15
1.7	Exemple de casque EEG	17
1.8	Méthodes d'acquisition de l'activité cérébrale (Waldert et al., 2009)	18
1.9	Spectre de puissance d'un signal EEG contenant des SSVEP (Lalor et al., 2005)	20
1.10	ERP et onde P300 (Kübler et al., 2001)	21
1.11	Comparaison des aires actives lors d'un mouvement imaginé ou exécuté (Hanakawa et al., 2008)	23
1.12	Exemple d'apprentissage pour contrôler les Slow Cortical Potential (Kübler et al., 2001)	24
1.13	Utilisation des ERD et du μ pour contrôler une ICM (Kübler et al., 2001)	25
1.14	Décomposition d'un signal en phase et amplitude	26
2.1	Présentation du cortex et des aires destinées au contrôle moteur	30
2.2	Exemple d'activités rythmiques dans les différentes bandes de fréquences pour un essai unique issue de l'activité sEEG	31
2.3	Directional tuning et décodage directionnel	32
2.4	Décodage continu de la cinématique d'un mouvement 2D	33
4.1	Évaluation statistique à base de permutations	39
4.2	Exemple de représentation temps-fréquence de puissance normalisées z-score (Ossandon et al., 2011)	40
4.3	Densité de probabilité d'une distribution d'amplitudes en fonction de tranches de phases	42
4.4	(A) Exemple de cartes temps-fréquence phase locked sur le β , (B) Exemple de comodulogramme	44
4.5	Exemple de phase préférentielle	46
4.6	Labellisation des données	47
4.7	Exemple d'une cross validation 3-folds	48
4.8	Principe du Linear Discriminant Analysis (Lotte et al., 2007, Naseer and Hong, 2015)	49

4.9	Principe du Support Vector Machine (Lotte et al., 2007, Naseer and Hong, 2015)	50
4.10	Principe du k-Nearest Neighbor (Weinberger et al., 2005)	50
4.11	Entraînement puis test d'un classifieur linéaire	51
4.12	Calcul de l'acuité de décodage	51
4.13	Exemple d'une <i>Forward feature selection</i> appliquée sur six features	54
4.14	Exemple d'une <i>Backward feature elimination</i> appliquée sur six features	55
4.15	Exemple de décodage temporel (Waldert et al., 2008). Ici, l'auteur décode 4-directions de mouvements de la main dans le temps. A chaque instant, un classifieur est créé, entraîné puis testé à ce même instant.	56
4.16	Exemple de généralisation temporelle (King and Dehaene, 2014)	57
5.1	<i>ipywksp</i> : Exemple de workspace pour <i>Jupyter</i>	60
5.2	Exemple de calcul PAC avec brainpipe	62
5.3	Exemple de plot d'un signal et de sa déviation	64
5.4	Exemples des principales fonctionnalités de Brain	65
5.5	Exemples des principales fonctionnalités de Sleep	66
5.6	Exemples des principales fonctionnalités de Ndviz	67
5.7	Exemple de mise en page avec le module Figure	68
6.1	Comparatif de résolution spatiale et temporelle pour différentes techniques d'imagerie (Lachaux et al., 2003)	71
6.2	Détails cliniques des sujets ayant participé à la tâche <i>Center-out</i>	72
6.3	Implantation intracrâniale et couverture corticale de six sujets épileptiques ayant passés la tâche <i>Center-out</i>	73
6.4	Descriptif de la tâche <i>Center-out</i>	73
A.1	Cartes des Interfaces Cerveau-Machine (Graimann et al., 2009)	191
A.2	Jeux de données en libre accès (<i>BCI competition</i>)	192
A.3	Comparatif de méthodes PAC (Tort et al., 2010)	193
A.4	Pipeline standard de classification	194
A.5	Comparatif de classifieurs (Pedregosa et al., 2011)	196
A.6	Exemple de schéma d'implantation	198

NOTATIONS

Général

ICM	Interface Cerveau-Machine
ICO	Interface Cerveau-Ordinateur
BCI	Brain Computer Interface
BMI	Brain Machine Interface

Enregistrements

EEG	Électroencéphalographie
MEG	Magnétoencéphalographie
SUA	Single Unit Activity
MUA	Multi Unit Activity
SEEG	Stéréoélectroencéphalographie
ECoG	Électrocorticographie
IRMf	Imagerie par Résonance Magnétique fonctionnelle
fNIRS	Functional near-infrared spectroscopy

Marqueurs/Motifs/Attributs/Pattern/Features

SCPS	Slow Cortical Potential
SSEP	Steady-State Evoked Potentials
ERS	Event-Related Synchronization
ERD	Event-Related Desynchronization
PAC	Phase-Amplitude Coupling

Classificateurs

LDA	Linear Discriminant Analysis
SVM	Support Vector Machine
RF	Random Forest
KNN	k-Nearest Neighbor
NB	Naive Bayes

Première partie

Introduction

DÉCODAGE DE L'ACTIVITÉ CÉRÉBRALE

1.1 LES INTERFACE CERVEAU-MACHINE

1.1.1 Définition et objectifs

Cette première section a pour but d'introduire et de définir le concept d'Interface Cerveau-Machine . Nous verrons dans quel contexte elles sont apparues, les personnes à qui elles sont destinées ainsi que les principaux éléments qui les composent.

Point de vue lexicale, on utilisera indifféremment *Interface Cerveau-Machine (ICM)*, *Interface Cerveau-Ordinateur (ICO)* ou les termes anglais correspondant à savoir *Brain Computer Interface (BCI)* et *Brain Machine Interface (BMI)*.

1.1.1.1 Contexte d'apparition des ICM

En 1964, Dr. Grey Walter connecte des électrodes directement dans le cortex moteur d'un patient et lui demande de presser un bouton pour faire avancer un rétro-projecteur. En même temps, il enregistre l'activité neuronale de telle sorte que elle aussi, puisse le faire avancer. Là où l'expérience devient remarquable, c'est que le rétro-projecteur avance avant que le patient ne presse le bouton! Tout l'appareil musculaire du sujet est court-circuité et le contrôle se fait sans mouvement. Contrôler par la *pensée*, un sujet de science fiction qui devient une réalité. Cette anecdote décrite par [Graimann et al. \(2009\)](#), permet de placer la naissance de la possibilité d'une Interface Cerveau-Machine (ICM) dans l'histoire. C'est le point d'entrée qui a ensuite conduit une grande diversité de chercheurs à se passionner pour ce sujet. Le terme *Brain Computer Interface* fait son apparition, au début des années 70, dans les publications de Jacques Vidal ([Vidal, 1973, 1977](#)) où il était question de contrôler un curseur sur un écran.

La progression des ICM et de l'intérêt de la communauté scientifique à véritablement commencé dans les années 2000. Trois facteurs sont à l'origine ([Wolpaw et al., 2002](#)) :

1. Une amélioration des connaissances des processus neuro-physiologiques et des techniques d'imagerie.
2. L'arrivée d'ordinateur bon marché et l'amélioration constante de leur performances et des composants électroniques (processeurs, mémoire vive, logiciel...)
3. Une prise de conscience sociétale des besoins de personnes souffrant de problèmes neuro-musculaires.

A noter que, à l'heure actuelle, l'arrivée de cartes graphiques bon marché est entrain de révolutionner l'approche computationnelle des ICM permettant des calculs plus lourds en moins de temps (notamment pour le *Deep Learning*). Outre la continue amélioration des composants et de leur miniaturisation, la prochaine révolution concernera certainement les ordinateurs quantiques, déjà en phase de test dans les domaines de la génétique et de la chimie.

1.1.1.2 Interactions naturelles avec l'environnement

Pour interagir avec son environnement, l'individu se sert des voies de communications naturelles, c'est-à-dire via son système nerveux et musculaire. Le processus de communication débute par une intention qui active certaines régions dans le cerveau. Il en résulte un signal cérébral qui est ensuite envoyé par le système nerveux périphérique en directions des muscles ([Besserve, 2007](#)). C'est ce processus simplifié qui permet à une personne d'interagir avec ce qui l'entoure.

1.1.1.3 Un canal de communication alternatif

Il existe plusieurs maladies ou accidents qui entraînent une dégénérescence des performances motrices. Parmi elles, on peut par exemple citer la Sclérose Latérale Amyotrophique (ou SLA), les accidents vasculaires cérébraux, certaines formes de sclérose, les lésions de la moelle épinière... Toutes ont en commun la possibilité de problème moteur. Dans ce contexte, [Wolpaw et al. \(2002\)](#) introduit trois options pour restaurer ces fonctions :

1. Augmenter les capacités des facultés motrices restantes. Autrement dit, donner un sens nouveau aux mouvements que l'individu est toujours en capacité de faire. A titre d'exemple, le guitariste virtuose Jason Becker, reconnu pour sa vélocité et dont tout le monde s'entendait sur son incroyable talent, fût un jour frappé par la SLA le conduisant au fur et à mesure à l'immobilité totale. Il convenu alors d'un langage basé sur les mouvements oculaires et, avec la complicité de son père, continua de composer.
2. Contourner la lésion. L'auteur donne à titre d'exemple, une lésion de la moelle épinière que l'on peut contourner en utilisant l'activité des muscles situés au dessus de la lésion pour stimuler les muscles paralysés.
3. Enfin, la dernière façon de restaurer des fonctions motrices qui prend tout son sens lorsque les deux précédentes ne sont pas possibles, c'est d'établir un nouveau canal de communication directe entre le cerveau et un ordinateur, et ce, indépendamment de l'activité musculaire. D'où le nom, *Interface Cerveau-Machine*.

Une ICM est un autre système de communication où les voies naturelles sont cout-circuitées. Au lieu de passer par le système nerveux puis musculaire, le signal cérébral est directement intercepté au niveau du cerveau et va ensuite être transformé en commandes. Une ICM est donc un système permettant de traduire une activité neuronale en commande extérieure. Le terme *traduire* est à prendre au sens linguistique c'est-à-dire que les signaux cérébraux forment un langage, composé de règles, de motifs ou *pattern*, que l'on va essayé de décoder (via un ordinateur) pour les transformer en opérations. D'où le terme "Interface Cerveau-Machine".

[Pfurtscheller et al. \(2008\)](#) et ([Graimann et al., 2009](#)) introduisent quatre éléments qui composent une ICM :

1. Enregistrer l'activité directement depuis le cerveau. Cet enregistrement pourra être invasif ou non-invasif (cf. 1.1.2)
2. Générer un retour ou *feedback* pour l'utilisateur
3. L'enregistrement et le *feedback* doivent être en temps réel
4. Enfin, l'interface doit être contrôlable par l'utilisateur, de manière active, via un ensemble d'intentions.

A titre d'exemple et pour illustrer ce dernier point, un utilisateur pourrait par exemple décider de bouger un curseur de souris sur un écran en imaginant des mouvements soit de la main gauche soit de la main droite.

1.1.1.4 Principales composantes d'une ICM

Même si chaque Interface Cerveau-Machine se destine à une utilisation particulière et contient des traitements qui lui sont propres, on peut globalement dire qu'une ICM s'articule autour de cinq grandes étapes ordonnées (Pfurtscheller et al., 2008, Graimann et al., 2009) :

1. L'acquisition de l'activité neuronale
2. Les pré-traitements
3. L'extraction de marqueurs
4. La classification
5. La transformation en commande

Ces étapes sont interdépendantes c'est-à-dire que chacune s'appuie sur les résultats de l'étape précédente. De plus, cette cascade de stades doit se faire en temps réel pour que l'expérience utilisateur soit la plus fluide possible et qu'elle reflète fidèlement ce qui se déroule à chaque instant.

1.1.1.4.1 Acquisition de l'activité neuronale L'acquisition de l'activité neuronale constitue le point d'entrée d'une ICM. Les différentes techniques pour enregistrer sont plus ou moins accessibles (certaines sont portatives, d'autres nécessitent un appareil très lourd...). C'est, entre autre, l'accessibilité qui va influer sur le nombre de sujets d'une étude. Autre point très important que nous décrirons plus bas, la qualité du signal (ou le rapport signal sur bruit (RSB)) qui aura un impact immédiat sur les performances et sur les limitations d'une ICM. Enfin, on parlera d'enregistrements *invasifs* (cf. 1.1.2.1) quand ceux-ci nécessiteront une implantation chirurgicale d'électrodes et *non-invasif* (cf. 1.1.2.2) pour les techniques d'acquisition se faisant en dehors de la boîte crânienne.

1.1.1.4.2 Pré-traitements Les pré-traitements regroupent un ensemble de techniques destinées à nettoyer le signal pour faire ressortir, autant que possible, le signal utile par rapport au signal bruité. Parmi ces traitements, on peut citer le nettoyage d'artefacts oculaires, cardiaques ou musculaires, le référencement (essentiellement pour l'EEG), la bipolarisation (pour la SEEG), le filtrage pour supprimer certaines composantes spectrales... Ces pré-traitements sont propres à chaque technique d'enregistrement. Une description plus détaillée des pré-traitements appliqués dans le cadre de données SEEG est proposée dans la section 6.1.4.

1.1.1.4.3 Extractions de marqueurs En imaginant que l'on répète dix fois le même mouvement, il y aura dans l'activité neuronale une partie similaire permettant de reproduire chacune de ces répétitions. Le signal entier sera très probablement différent à chaque fois, mais, à l'intérieur de ce signal, on pourra trouver un "sous-signal" dont le contenu sera similaire à chacune de ces répétitions.

C'est le but de cette étape d'extraction de marqueurs, la recherche de ce "sous-signal". Une fois l'activité cérébrale nettoyée, on va chercher à extraire des marqueurs qui matérialisent l'état instantané d'un sujet. Par exemple, si celui-ci bouge le bras vers la gauche ou vers la droite, on doit pouvoir extraire une information de ce signal qui encode chacun de ces états. En pratique, on peut distinguer deux types de marqueurs : les marqueurs *locaux*, qui reflètent l'activité d'une "petite" population de neurones prises localement (cf. 1.1.4), et les marqueurs d'*interaction* qui quantifie un degrés de couplage à distance entre deux régions du cerveau.

Cette étape est véritablement au cœur du bon fonctionnement d'une ICM puisque, en fonction de la qualité de ce marqueur, la machine sera plus moins enclue à reconnaître les différentes commandes d'un sujet.

En lieu et place du terme *marqueur*, on pourra utiliser indifféremment *motif*, *pattern*, *feature* ou *attribut*.

1.1.1.4.4 Classification En reprenant l'exemple de la section précédente, supposons que l'on dispose d'un marqueur et on cherche à savoir si celui-ci appartient à une des deux classes de mouvement de bras, vers la gauche ou vers la droite. C'est le problème de classification. A partir d'un motif dont on ignore la provenance, on utilise un algorithme permettant de reconnaître la classe dont est issue ce pattern. Parmi les algorithmes les plus fréquemment rencontrés, on peut citer le Linear Discriminant Analysis , le Support Vector Machine ou le k-Nearest Neighbor .

En pratique, cette reconnaissance de classes est mise en place en deux étapes :

1. L'entraînement (ou *training*) : durant une certaine période, on va apprendre à une machine à reconnaître des événements. Pour cela, on utilise des données dont on connaît la provenance (c'est ce que l'on appelle la *labellisation*)
2. Le test (ou *testing*) : une fois la machine entraînée à partir d'une série de marqueurs labélisés, on teste l'algorithme avec des nouvelles données pour évaluer l'acuité de la machine à identifier ces événements.

Quelque soit l'algorithme de classification, celui-ci doit s'adapter à chaque utilisateur suivant trois niveaux ([Wolpaw et al., 2002](#)) :

1. En premier lieu, et de manière assez évidente, il doit pouvoir s'adapter au marqueur du sujet
2. Ensuite, l'algorithme doit s'adapter en temps-réel aux variations spontanées. En effet, l'expérience utilisateur va varier en fonction d'un certain nombre de paramètres comme le moment de la journée, la fatigue, la maladie, le taux d'hormones, la motivation/concentration/frustration... ([Curran, 2003](#))
3. Enfin, il doit permettre de prendre en compte l'adaptation du sujet. Durant l'expérience, l'utilisateur module son activité et fournit des efforts pour s'adapter au fonctionnement de la machine. En contrepartie, le software doit prendre en compte cette amélioration en fournissant des performances accrues.

Cette étape de classification, décrite plus largement dans la section 4.2, est très fortement dépendante de la qualité des marqueurs extraits en amont. Autrement

dit, plus ces *features* reflètent fidèlement un état, plus le travail de la machine à identifier ces événements sera facilité.

1.1.1.4.5 Transformation en commande Dernière étape du processus d'une ICM, lorsque l'algorithme de classification pense avoir identifié le type de marqueurs, on attribue une commande physique. Par exemple, si la machine reconnaît un mouvement de bras vers la droite, on pourrait attribuer une commande où l'on déplace le curseur d'une souris sur un écran dans la même direction. Ainsi, on procure à l'utilisateur un *feedback* sur la transformation que la machine a réussit à faire à partir de l'activité de son cerveau.

Cette transformation en commande est ensuite physiquement appliquée à un système externe. L'efficacité de l'ensemble de l'ICM est donc évaluée en fonction de son accuité à restituer, avec plus ou moins de fidélité, la commande désirée par l'utilisateur.

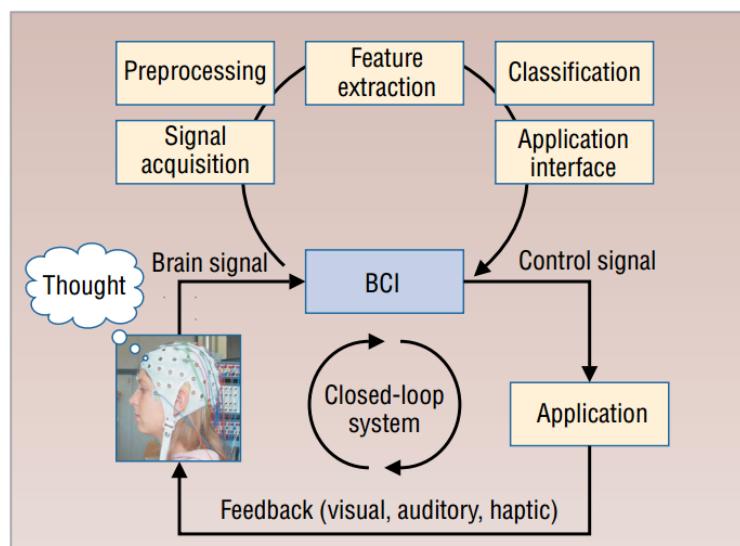


FIGURE 1.1 – L'activité neuronale est enregistrée (*Signal acquisition*) puis nettoyée (*Preprocessing*).

Ensuite, on extrait des motifs ou patterns qui caractérisent la commande que souhaite envoyer le sujet (*Feature extraction*). Enfin, la machine tente de reconnaître ces motifs (*Classification*) et de les transformer en commande (*Application interface*). Cette boucle se termine en donnant un feedback à l'utilisateur sur l'état actuel de la machine. (Pfurtscheller et al., 2008)

Les ICM partagent globalement ces cinq étapes mais se différencient donc le type d'enregistrement de l'activité neuronale, par les pré-traitements associés, par le type de marqueurs étudié, par l'algorithme de classification choisi et surtout, par l'application concrète de cette Interface Cerveau-Machine .

1.1.1.5 Applications des ICM : cliniques et non-cliniques

La réalisation concrète des ICM s'est articulée autour des applications cliniques, c'est-à-dire destinées à essayer d'améliorer les conditions de vie de certaines personnes puis, dans un second temps, pour des applications ludiques et tout public.

1.1.1.5.1 Applications cliniques A l'heure actuelle, il existe de nombreuses maladies soit dont on ignore l'origine, soit qui sont pour le moment incurables. Parmi ces pathologies, certaines évoluent en enfermant les patients dans des conditions de vies

difficiles. C'est dans ce contexte clinique qu'apparaissent de nombreuses ICM qui ne sont donc pas des traitements, mais bien des solutions *temporaires* pour aider certaines personnes à mieux vivre avec leur handicap.

Pathologies et ICM : Prélever directement l'activité neuronale et donc, *bypasser* les voies naturelles, permet de s'affranchir d'éventuelles limitations physiologiques. C'est pourquoi les ICM représentent un enjeu majeur pour la réhabilitation motrice ou handicap moteur ou encore pour la communication palliative. Les applications concrètes des Interface Cerveau-Machine visent donc les personnes disposant de leurs capacités cognitives mais qui sont privées de facultés motrices.

Un accident vasculaire cérébral (AVC) peut engendrer un état d'enfermement (*locked-in state (LIS)*). Les personnes dans cet état sont pleinement conscientes de leur corps, de l'environnement, ils peuvent ressentir les sensations de toucher et de douleur mais n'ont plus de facultés motrices, hormis peut-être, les mouvements de paupières ou des yeux. Un autre exemple est celui de la sclérose latérale amyotrophique (ou SLA) qui est une dégénérescence des neurones moteur (motoneurones). Progressivement, les personnes atteintes de SLA perdent l'usage des bras, des jambes, de la parole des muscles faciaux et enfin de la déglutition mais la conscience et les facultés cognitives demeurent intactes. L'évolution de la maladie amène à deux possibilités ([Chaudhary et al., 2015](#)) : accepter une dépendance totale (respiration artificielle et nutrition) ou un décès par insuffisance respiratoire. Dans le premier cas, la maladie entraîne progressivement les patients en LIS ne leur laissant qu'un minimum de fonctions motrices. Lorsque le patient perd tout contrôle musculaire, ce qui finit souvent par les muscles des yeux, il rentre dans un état d'enfermement complet (*completely locked-in state (CLIS)*).

SLA et *locked-in syndrome* ne sont que deux exemples expliquant l'intérêt social du développement des ICM. Redonner un peu de contrôle ou établir un canal de communication avec les familles des patients expliquent l'engouement qui existe depuis maintenant plus de 40 ans pour les ICM.

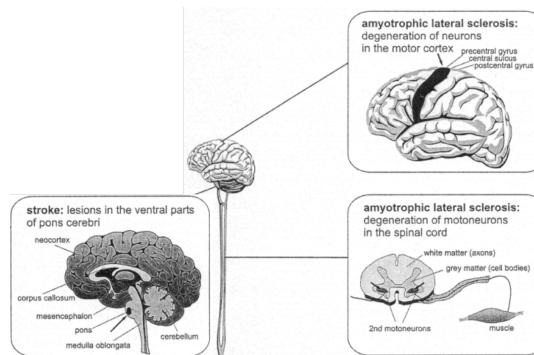


FIGURE 1.2 – Paralysie causée par la SLA et accident vasculaire cérébral ([Kübler et al., 2001](#))

Exemples d'applications : les applications cliniques peuvent être divisées en deux grandes familles : les ICM destinées à rétablir un lien de communication (*communication palliative*) et les ICM pour redonner de la mobilité.

Communication palliative : ces ICM présentent en général des lettres ou groupe de lettres qui doivent pouvoir être sélectionnés volontairement par l'utilisateur. Le *P300-Speller* ([Farwell and Donchin, 1988](#), [Donchin et al., 2000](#)), utilisant l'onde P300, est sans aucun doute l'interface la plus connue et la plus exploitée à ce jour pour

la communication palliative. Le *Hex-o-spell* (Blankertz et al., 2007) est une autre interface contrôlable par l'utilisateur via l'imagerie motrice.

Mobilité, robotique et prothèse : ces systèmes sont destinés à aider, améliorer ou remplacer des facultés motrices lésées. Ces applications dont la mobilité est au centre peuvent afficher différents degrés de complexité :

- Curseur 1D, 2D et 3D : certainement la première étape, le but ici est de permettre au sujet de pouvoir contrôler le curseur sur un moniteur. Ce curseur pourra par exemple servir de bouton *yes/no* ou pour le déplacement (Wolppaw et al., 1991, Wu et al., 2003, Trejo et al., 2006, Kayagil et al., 2009, Kim et al., 2011, Vadera et al., 2013) ou tout simplement pour de la navigation comme le *web-browsing* (Mugler et al., 2010). Ces catégories de déplacement de curseur diffèrent par le nombre de degrés de liberté qu'elles offrent.
- Contrôle d'un fauteuil roulant : comme le nom l'indique, l'idée ici est de permettre à un utilisateur de contrôler son fauteuil roulant via son activité neuronale seulement. En effet, si de nombreuses personnes handicapées peuvent encore se servir de leur membres supérieurs, d'autres sont complètement dépendantes. Ce type d'applications permettra donc à terme de redonner une liberté de mouvement à ces personnes (Tanaka et al., 2005, Leeb et al., 2007, Galán et al., 2008a, Philips et al., 2007, Vanacker et al., 2007, Pires et al., 2008, Rebsamen, 2009, Lin et al., 2010, Diez et al., 2013).
- Prothèse : enfin, dernière application liée à la mobilité, le contrôle de prothèse est un large défi. A titre d'exemple, le contrôle d'un bras robotisé doit permettre le contrôle de celui-ci dans l'espace ainsi que des mouvements de main, de coude... C'est un problème à haute dimensionnalité et donc complexe. Toutefois, modulo un certain degrés de réussite, certaine équipe ont proposé de tel systèmes (Fetz and others, 1999, Hochberg et al., 2012, Yanagisawa et al., 2012a, Sunny et al., 2016)

1.1.1.5.2 Applications non-cliniques et récréatives

Les ICM ont également été exploitées à d'autres fins que des applications cliniques :

Jeux : les ICM dédiées aux jeux utilisent l'activité neuronale pour contrôler un vaisseau spatial, un objet ou un personnage dans un environnement virtuel (Lalor et al., 2004, Nijholt, 2008, Oude Bos and Reuderink, 2008, Coyle et al., 2011). Outre le caractère récréatif de ce type de dispositif, les ICM basées sur le jeux pourraient très bien servir pour entraîner la machine d'une manière plus ludique et moins fatigante.

Art : pour finir, les ICM peuvent avoir des applications artistiques comme pour la composition et la pratique musicale (Miranda et al., 2003, Miranda, 2006, Hamadicharef et al., 2010) ou pour peindre (Münßinger et al., 2010, Zickler et al., 2013)

1.1.1.6 BCI competition et open-data

Les *BCI competitions* sont proposées par l'équipe du *Berlin Brain-Computer Interface* (BBCI). L'idée de ces compétitions est, sur un même jeu de données découpées en *training* et *testing* (cf. 4.2.2) mettre les équipes de recherche en ICM en compétitions puis élire celle qui arrivera au meilleur décodage. A l'heure où cette thèse est écrite, quatre de ces compétitions ont eu lieu et chacune est associée à un article

de présentation (*BCI competition I* : [Sajda et al. \(2003\)](#), *II* : [Blankertz et al. \(2004\)](#), *III* : [Blankertz et al. \(2006\)](#), *IV* : [Tangermann et al. \(2012\)](#)). La liste des *datasets* en accès libres est disponible en annexe (cf. A.2. Remarque : il semble que les données de la première compétition ne soient plus accessibles, c'est la raison pour laquelle elles n'apparaissent pas dans le tableau)

Ces compétitions présentent de nombreux avantages :

1. Visibilité des équipes : ces compétitions sont de plus en plus "virales" et sont relayées par la suite par de nombreux articles (voir figure ci-dessous).
2. Permet aux équipes de disposant pas de systèmes d'acquisition de profiter de jeux de données pour développer des méthodes. C'est également un moyen d'initier les étudiants à un ensemble de techniques.
3. Autre avantage, tout le monde se retrouve sur un pied d'égalité en travaillant sur les mêmes données. Ce qui représente par la suite, une méthode rigoureuse pour sélectionner les meilleurs algorithmes.

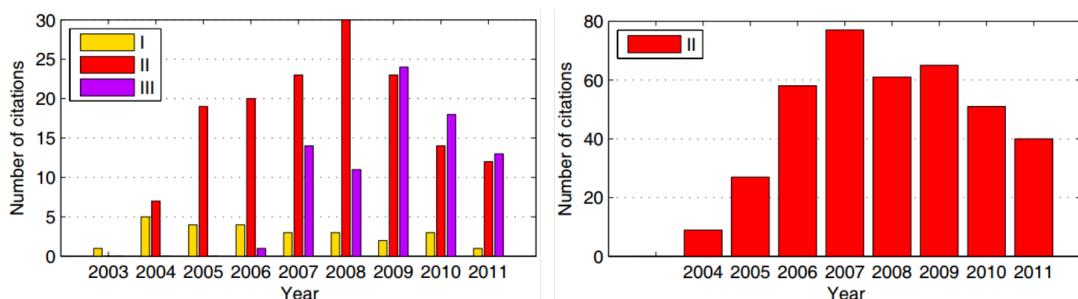


FIGURE 1.3 – (A gauche) Évolution du nombre de citations par an des articles de présentation pour les compétitions I, II et III, (A droite) Évolution du nombre de citations par an pour l'équipe ayant gagné la deuxième compétitions ([Tangermann et al., 2012](#))

Conclusion sur la présentation des ICM

Les personnes atteintes de problèmes moteurs sont certainement les premiers bénéficiaires de l'avancée des Interface Cerveau-Machine . Ces systèmes leur permettraient de rétablir un canal de communication avec leur environnement. Pour cela, les ICM partent de l'enregistrement de l'activité de leur cerveau puis, recherche dans cette activité des motifs relatant un état. Si cet état est *compris* par la machine, il sera ensuite transformé en commande. De nombreux systèmes se sont développés, principalement autour de la communication palliative ou de la mobilité. L'efficacité de ces systèmes reposent en grande partie sur deux points : l'acuité des algorithmes de classification, pour cette raison, ils peuvent être comparés notamment grâce à des initiatives comme les *BCI competitions*. Le deuxième point concerne l'enregistrement de l'activité neuronale qui va très largement conditionner la qualité du signal et des marqueurs qui en seront extraits.

1.1.2 Techniques d'acquisition de l'activité neuronale

Les types d'acquisition peuvent être classés par leur degrés de pénétration dans le corps. Les enregistrements dits *invasifs* vont nécessiter une intervention chirurgicale, donc avec risque potentiel d'infection, mais donnent accès à des signaux de

grande qualité permettant des contrôles complexes comme une prothèse ou un bras robotisé (Hochberg et al., 2012, Taylor, 2002). Les enregistrements *non-invasifs* sont globalement plus faciles à mettre en place car ne nécessitant aucune chirurgie mais la qualité du signal est moindre car l'activité neuronale est enregistrée en dehors de la boîte crânienne, donc filtrée par l'os et la peau.

Au sein de ces deux catégories, on trouvera un ensemble de méthodes d'acquisition qui se différencient par la taille des populations de neurones qu'elles enregistrent ou par le type de signal (électrique, magnétique, mesure indirecte...).

1.1.2.1 Enregistrements invasifs

On parlera d'intracrânien pour les enregistrements à l'intérieur de la boîte crânienne puis d'intracortical pour des électrodes implantées dans le cortex (Engel et al., 2005, Jerbi et al., 2009b).

1.1.2.1.1 Enregistrements unitaires *Single Unit Activity (SUA)* et *Multi Unit Activity (MUA)* sont des micro-électrodes déposées directement au contact de neurones et enregistrent des décharges neuronales. SUA et MUA se distinguent par la taille des populations enregistrées. Les décharges sont événements très courts dans le temps, donc pour être en mesure de les capter, les systèmes d'acquisition possèdent des fréquences d'échantillonnage très élevées (de l'ordre de 30khz). Le signal obtenu en filtrant en dessous de 300hz, est appelé *Local Field Potential (LFP)* et représente l'activité électrique d'une assemblée de neurones prise dans un petit volume.

Dans le cadre des ICM, les micro-électrodes peuvent être implantées directement dans le cortex moteur primaire et permettre un contrôle de BCI de la plus haute précision et fidélité. Chez l'animal, des rats ont pu contrôler en temps réel un bras robotisé pour obtenir de l'eau (Chapin et al., 1999) et des singes ont également pu contrôler un bras robotisé ainsi qu'une pince au bout (Velliste et al., 2008). Auparavant, en 2006, Hochberg et al. (2006) démontre qu'un patient tétraplégique implanté avec 96 micro-électrodes, peut contrôler un curseur sur un écran d'ordinateur, d'ouvrir et fermer une main artificielle ainsi que d'effectuer des mouvements rudimentaires à l'aide d'un bras robotisé. En complément, Kim et al. (2011) utilise les micro-électrodes pour contrôler et cliquer à l'aide un pointeur sur un écran 2D. Mais c'est en 2012 que le sujet contrôle complètement le bras robotisé, dans l'espace, lui permettant d'attraper et de boire son café (Hochberg et al., 2012). Même si le mouvement n'est pas aussi fluide et rapide qu'un mouvement réel, ce fût une avancée majeure et une preuve du concept chez l'homme. Collinger et al. (2013) décrivent également le contrôle d'un bras robotisé par un patient implanté avec 96-microélectrodes. Avec des taux de décodage élevés, le sujet réussit à contrôler un bras robotisé à 7-dimensions (3 degrés de translation, 3 degrés de rotation et un degrés de saisie).

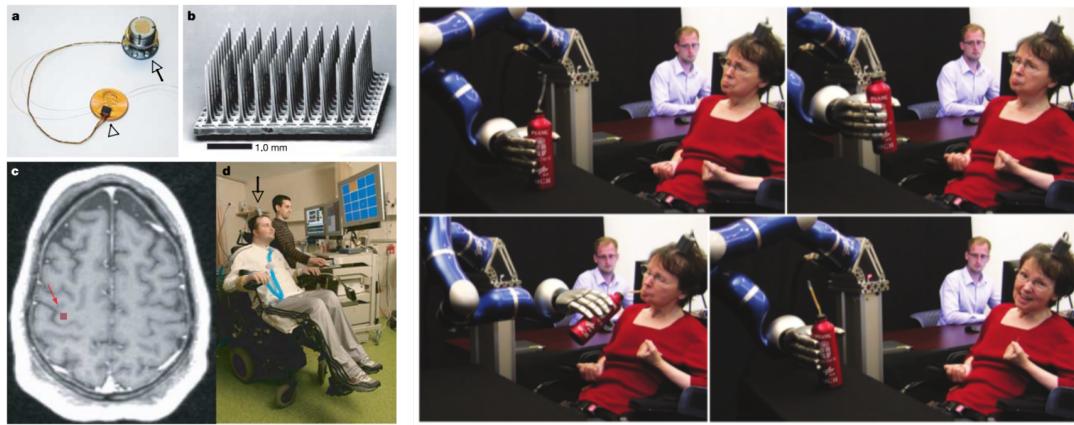


FIGURE 1.4 – (A gauche) Sujet tétraplégique implanté avec des micro-éléctrodes contrôlant un curseur sur un écran (Hochberg et al., 2006), (A droite) (Hochberg et al., 2012)

1.1.2.1.2 Stéréoélectroencéphalographie (SEEG) Macro-électrodes enregistrant des populations plus larges que les micro-électrodes. Contrairement à la SUA ou MUA où l'on peut compter le nombre de fois qu'un ou plusieurs neurones déchargent, la SEEG enregistre des potentiels électriques (μV). La fréquence d'échantillonnage est plus faible que celle des micro-électrodes (de l'ordre de 1kHz) et donne donc accès au signal LFP.

Les données utilisant la Stéréoélectroencéphalographie sont globalement rares dans la littérature. Leur utilisation dans le cadre des ICM n'est pas fréquente mais a quand même été exploré dans le cadre d'ICM dédiée à la communication (Krusienski and Shih, 2011a, Shih and Krusinski, 2012) ou au contrôle de curseur 2D (Vadera et al., 2013)

C'est le type d'enregistrement qui a été le plus exploité durant cette thèse. Une section complète lui est donc accordée (cf. 6.1).

1.1.2.1.3 Électrocorticographie (ECoG) L'ECoG est une grille flexible composée d'une matrice d'électrodes. Celle-ci est ensuite déposée à la surface corticale. Parce que cette méthode n'est pas intracortical elle est dite *semi-invasive*. En comparaison avec l'EEG (voir section suivante), l'ECoG présente de nombreux atouts (Schalk and Leuthardt, 2011). Tout d'abord, la résolution spatiale est meilleure (de l'ordre du millimètre contre quelques centimètres pour l'EEG). Une meilleure amplitude dont le maximum peut-être jusqu'à 5 fois supérieure à celle de l'EEG (amplitude max pour l'EEG est d'environ $10\text{-}20\mu V$). L'ECoG est également moins sensible aux artefacts (mouvements musculaires et oculaires). Enfin, elle permet d'enregistrer des phénomènes plus large bande (entre 0 et 500Hz contre 0-40Hz pour l'EEG).

L'utilisation de l'ECoG dans le cadre des ICM est fréquente (Schalk and Leuthardt, 2011, Shih et al., 2012), que ce soit pour le décodage appliqué aux membres supérieurs comme les mouvements de doigts (Scherer et al., 2009, Acharya et al., 2010), de trajectoire de main (Schalk et al., 2007, Gunduz et al., 2009) ou de bras (Pistohl et al., 2008). Il a également été démontré qu'il est possible de décoder des mouvements fins de saisie avec la main, comme des saisies précises avec le bout des doigts versus des saisies avec la main entière (Pistohl et al., 2012), ou encore des mouvements de saisie, de pincée, d'ouverture de la main, de flexion et d'extension du coude ou de bras robotisé (Muller-Putz and Pfurtscheller, 2008, Yanagisawa et al., 2012a). De plus, l'ECoG a également été exploité pour le contrôle d'un curseur 1 ou 2D (Leuthardt et al., 2004, Wilson et al., 2006, Felton et al., 2007, Milekovic et al.,

2012) ainsi que pour la communication palliative (Brunner et al., 2011, Krusienski and Shih, 2011b).

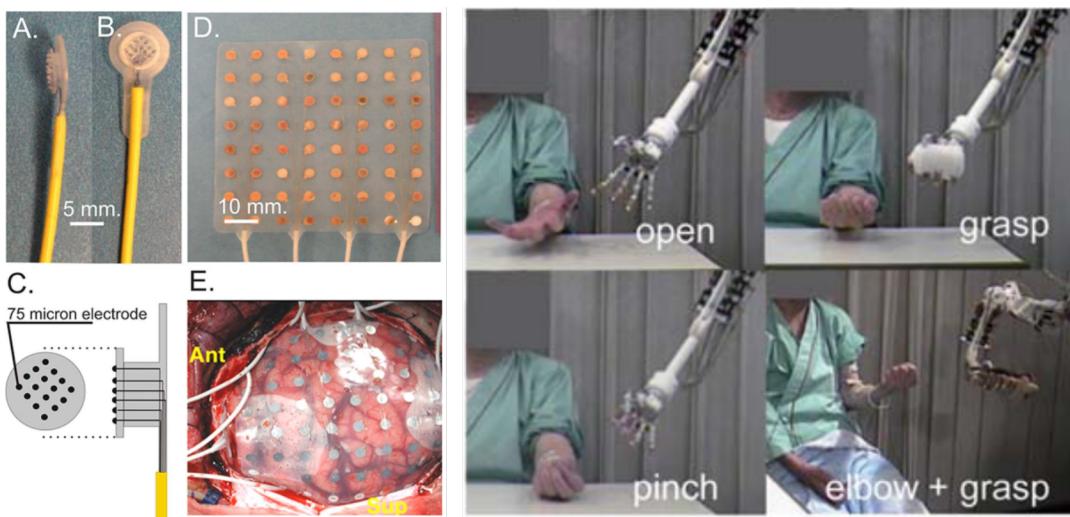


FIGURE 1.5 – A gauche - (A-B) Micro-électrode, (C) Représentation schématique d'une électrode, (D) Grille d'électrodes déposée directement au contact du cortex, (E) Placement chirurgical de la grille d'électrodes (Schalk and Leuthardt, 2011). A Droite - Décodage de mouvements de la main et contrôle d'un bras robotisé (Yanagisawa et al., 2012a)

1.1.2.2 Enregistrements non-invasifs

1.1.2.2.1 Introduction à la production des champs électriques et magnétiques Avant de présenter les différentes techniques d'enregistrement non-invasives, il m'est apparu intéressant de présenter succinctement la façon dont une population de neurones peut produire des champs électriques et magnétiques. Cela permettra d'une part de comprendre un peu mieux les mécanismes sous-jacents à l'EEG et à la MEG, leur limitations et surtout, leur complémentarité.

L'explication des production de champs électromagnétique s'appuiera sur la figure ci-dessous, issue de Sato et al. (1991) ainsi que sur l'article de Garner et al. (1998).

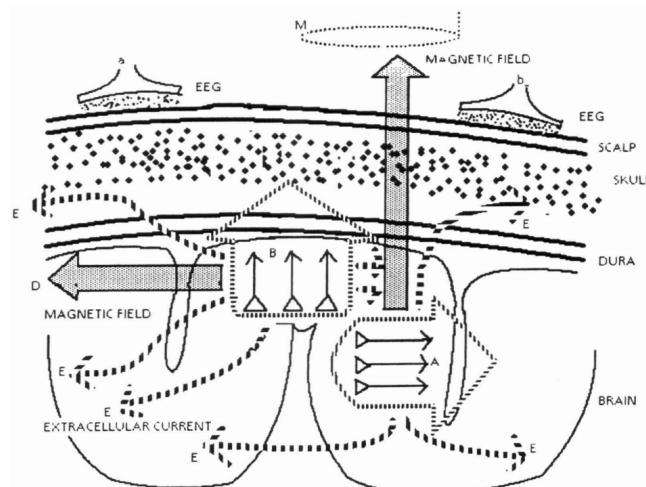


FIGURE 1.6 – Production des champs électriques et magnétiques. Recueil EEG et MEG, limitations et complémentarité (Sato et al., 1991)

a et b sont deux capteurs EEG disposés au contact du scalp. M est un capteur

MEG. **A** et **B** sont deux dipôles matérialisant la somme des courants issus d'une macro-colonne de neurones. Lorsqu'un neurone est excité, il y a libération d'ions au niveau de la membrane des synapses. Ces flux ioniques vont engendrés des courants dits *primaires* ou *sources* et qui seront à l'origine des signaux EEG et MEG. Ces courants primaires vont à leur tour engendrés des courants dits *secondaires* ou *volumiques*, circulant dans tout le volume de la tête (*extracellular current*). Un signal MEG résultera principalement des champs magnétiques produits par les courants sources, contrairement au signaux EEG qui sont majoritairement issus des courants extracellulaires et volumiques. **A** est un dipôle tangentiel (donc parallèle à la surface du crâne) disposé dans un sillon. Le champs magnétique **C** produit par ce dipôle, qui n'est pas filtré par les méninges, l'os et la peau, est bien capté par le capteur MEG. La colonne **B**, qui correspond aux activations situées dans les parties courbes du cortex (gyri) est un dipôle radiale donc perpendiculaire au scalp. Celui-ci produit un champs magnétique **D**, parallèle à la bobine **M**, qui sera donc pas ou peu détecté par la MEG. En revanche, cette source radiale entraîne un fort potentiel électrique. C'est donc un premier point de complémentarité entre la MEG et l'EEG. **A** et **B** engendrent des champs électriques qui seront fortement dispersés, atténus et distordus par la dure-mère, l'os et la peau. Ce phénomène de dispersion explique pourquoi il est beaucoup plus difficile de reconstruire les sources (problème inverse) en EEG plutôt qu'en MEG. Enfin, les propriétés physiques du champs magnétique font qu'il décroît davantage avec la distance que les champs électrique, ce qui limite l'étude des sources profondes mais c'est un deuxième argument justifiant leur complémentarité pour observer l'ensemble des phénomènes.

1.1.2.2.2 Électroencéphalographie (EEG) C'est la technique la plus utilisée dans le domaine des ICM pour son aspect pratique, portatif et peu dispendieux. On dispose à la surface de la boîte crânienne un ensemble d'électrodes qui enregistrent, sous forme de potentiel électrique, l'activité résultant d'une population relativement large de neurones. Une électrode (souvent sur le front ou le nez) est considérée comme référence. Le potentiel de cette électrode de référence est ensuite soustrait à toutes les autres pour obtenir une tension. On pourra utiliser un gel entre l'électrode et la boîte crânienne pour adapter l'impédance. L'EEG dispose d'une excellente résolution temporelle mais sa résolution spatiale est moins bonne (en partie dû au fait que les méninges, l'os puis la peau filtre le signal).

Les premiers enregistrements non-invasifs utilisant l'EEG chez l'homme, ont été rapporté par Berger (1929). Depuis, ils n'ont cessé de s'améliorer et à l'heure actuelle on peut trouver des systèmes sans fils et peu dispendieux permettant le contrôle d'une BMI (Lin et al., 2010, Liao et al., 2012, Liu et al., 2012). La transmission sans fil occasionne une perte de signal supplémentaire, donc la plupart des études utilisant l'EEG continu avec les bons vieux câbles et amplificateur. Il existe plusieurs sous-catégories d'ICM utilisant l'EEG. Celles-ci sont basées sur différents types de marqueurs (cf. 1.1.4).

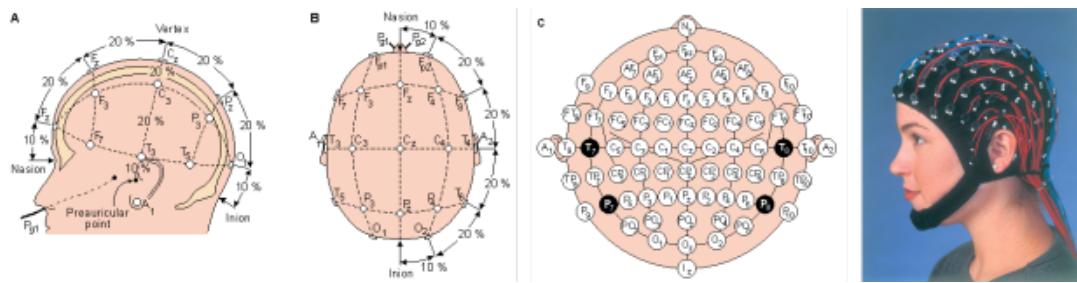


FIGURE 1.7 – Exemple de casque EEG (figure extraite et adaptée de <http://www-psych.nmsu.edu/~jkroger/lab/principles.html>

1.1.2.2.3 Magnétoencéphalographie (MEG) Nettamente moins portable que l'EEG, la MEG enregistre les champs magnétiques résultant d'une production de courants intracellulaires. Ces champs magnétiques peuvent être jusqu'à 10 milliards de fois plus faibles que le champ magnétique terrestre ce qui explique l'utilisation d'un blindage pour limiter les artefacts (un blindage en μ -métal peut atténuer de 10^3 à 10^4 l'influence des champs externes). Cet appareil, qui peut être composé de 100 à 300 capteurs, dispose également d'une excellente résolution temporelle et d'une résolution spatiale supérieure à celle de l'EEG.

L'aspect inamovible de la MEG limite son utilisation pour les ICM mais on trouve quand même quelques études ayant testé son utilisation (Mellinger et al., 2007, Walder et al., 2008).

1.1.2.2.4 Imagerie par Résonance Magnétique fonctionnelle (IRMf) L'IRMf permet de mesurer l'oxygénation d'aires actives grâce à l'apport en sang. Cette technique se base sur des différences de susceptibilité magnétique du fer entre le sang oxygéné (diamagnétique) et désoxygéné (paramagnétique). Le signal BOLD (Blood Oxygenation Level Dependent) mesure les variations locales du temps de relaxation causées par les modifications hémodynamiques. Si la résolution spatiale de l'IRMf est excellente puisqu'elle est de l'ordre du millimètre, la résolution temporelle, quant à elle, est assez faible (de l'ordre de la seconde) ce qui limite la captation de phénomènes temporellement courts.

L'utilisation de l'IRMf pour les ICM a été exploitée Sitaram et al. (2007a). Cette étude, qui propose plusieurs ICM-IRMf existantes, explique qu'une des principales limitations de cette technique d'enregistrement est son coût et la complexité liée à son usage. Enfin, Weiskopf et al. (2004) explique qu'une autre limitation est qu'il peut s'écouler entre 3 et 6 secondes pour observer les changements hémodynamiques.

1.1.2.2.5 Functional near-infrared spectroscopy (fNIRS) Technique d'imagerie relativement récente, puisque la première description du principe fut décrite par (Jobsis, 1977), celle-ci exploite la lumière infra-rouge (longueur d'onde entre 650 et 1000nm) pour mesurer les variations de concentration de l'hémoglobine oxygénée (HbO) et l'hémoglobine désoxygénée (HbR). La principale limitation de cette technique est qu'elle ne permet pas l'étude de structures profondes (profondeur de 3cm maximum à partir du sommet du crâne).

Cette technique est de plus en plus rencontrée dans la littérature ICM dû à son moindre coût et à sa portabilité. La première ICM utilisant la fNIRS a été décrite par Coyle et al. (2004). Dans cette étude, les auteurs utilisent l'imagerie motrice (compression d'une balle en caoutchouc) et déterminent si l'activité du sujet et en

activité ou au repos. Depuis, bien d'autres études ont suivi, dont beaucoup se focalisent sur l'imagerie motrice (Sitaram et al., 2007b, Nagaoka et al., 2010, Fazli et al., 2012, Mihara et al., 2013, Zimmermann et al., 2013, Kaiser et al., 2014). Naseer and Hong (2015) propose une review récente des principales ICM-fNIRS.

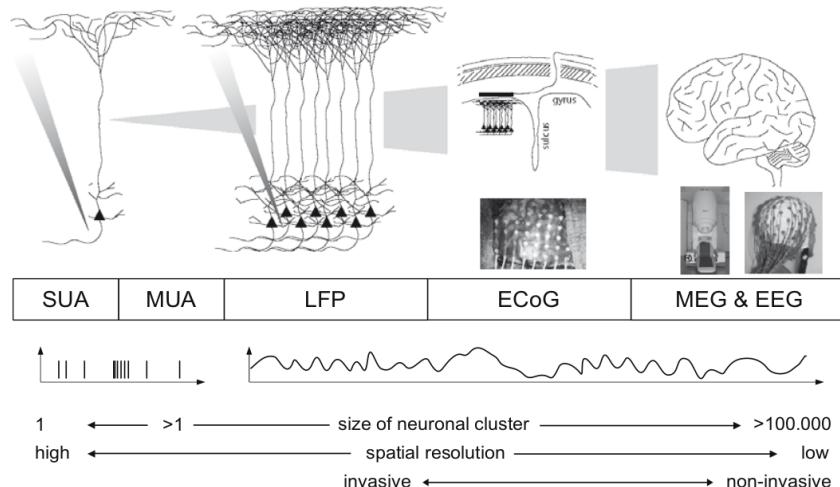


FIGURE 1.8 – Méthodes d’acquisition de l’activité cérébrale (Waldert et al., 2009) classées par invasivité. La figure indique également la taille des populations de neurones enregistrés ainsi que la résolution spatiale intimement liée à l’invasivité.

Conclusion sur les techniques d’enregistrements

Les méthodes non-invasives ne nécessitent aucune intervention chirurgicale et peuvent donc être appliquées sur n’importe quel individu. De plus, elles sont particulièrement accessibles, c’est-à-dire que leur mise en place est relativement simple.

Toutefois, ces méthodes de mesures souffrent encore de compromis. Si l’EEG et la MEG ont une excellente résolution temporelle, la résolution spatiale est peu précise. A contrario, l’IRMf ou la PET offrent une excellente résolution spatiale mais la dimension temporelle est lésée. Les méthodes invasives ne souffrent pas de ces compromis, puisqu’elles offrent à la fois une excellente résolution temporelle et spatiale, au détriment d’une chirurgie invasive. Enfin, les méthodes non-invasives sont beaucoup plus sensibles aux artefacts (musculaires et oculaires) et ont un rapport signal sur bruit (RSB) inférieur aux enregistrements invasifs.

Malgré tout, les techniques d’enregistrement non-invasives jouissent d’un engouement certain de la part de la communauté scientifique, notamment grâce au critère d’accessibilité. L’EEG a très certainement un bel avenir devant lui, en attendant que des chercheurs réussissent à mettre des roulettes à la MEG ou à l’IRM. Sans nul doute, les techniques non-invasives devraient être au cœur des Interfaces Cerveau-Machine du futur.

1.1.3 ICM synchrones/asynchrones et invasives/non-invasives

On peut distinguer différents types d’ICM, sur la base de deux critères (Donoghue, 2002, Lebedev and Nicolelis, 2006, Besserve, 2007, Bekaert et al., 2009) :

- **La synchronisation** : ce critère va définir le fonctionnement interne de l'ICM c'est-à-dire qu'il va fixer la façon dont un utilisateur va pouvoir interagir avec elle. Soit de façon volontaire en modifiant son activité neuronale, ce sont les *ICM asynchrones*, soit de façon imposée par l'utilisation de stimuli externes qui permettront de piloter l'interface, ce sont les *ICM synchrones*.
- **Le type d'enregistrement** : on distinguera les *ICM invasives* des *ICM non-invasives* de part l'utilisation de technique d'enregistrement nécessitant ou non, une chirurgie pour planter des électrodes (cf. 1.1.2).

1.1.3.1 Synchronisation : ICM synchrones et asynchrones

1. *ICM synchrones ou exogènes* : exploitent la réponse du cerveau à des stimuli externe (visuels, auditifs...). Par exemple, un damier composé de cases blanches et noires entraînera une forte variation dans les potentiels visuels. Cette différence de potentiel peut ensuite être détectée puis transformée en commande. Ce type d'ICM présente l'avantage de nécessiter que très peu d'entraînement. En revanche, la réponse étant dépendante du stimulus, le comportement est booléen ce qui limite dans les possibilités pour un contrôle progressif et continu.
2. *ICM asynchrones ou endogène* : ici, grâce à un *feedback*, l'utilisateur change volontairement son activité neuronale pour influer sur le comportement de l'ICM. En pratique, on pourra par exemple se servir de l'imagerie motrice pour avoir un contrôle continu et progressif d'un curseur de souris. Toutefois, les ICM asynchrones nécessitent une longue période d'apprentissage avant de pouvoir reconnaître les *patterns* propres à chaque sujet.

1.1.4 Signaux physiologiques pour le contrôle d'une ICM

Les signaux peuvent classés en deux catégories ([Wolpaw et al., 2002](#), [Pfurtscheller et al., 2008](#)) :

- **Les réponses évoquées ou exogènes** : sont produites, sans que le sujet en ait conscience, suite à un stimuli externe.
- **Les réponses spontanées ou endogènes** : celles-ci peuvent être volontairement modifiées par l'utilisateur.

1.1.4.1 Réponses évoquées

Comme décrit plus haut, les signaux évoqués sont provoqués par un stimuli externe et engendrent une réponse spécifique, c'est-à-dire localement dans le cerveau et à des instants précis. Ces signaux ainsi provoqués prendront des valeurs différentes en fonction de l'intention du sujet. Et c'est grâce à ces variations de valeurs que le sujet pourra contrôler la BMI. Le sujet est donc dépendant des stimulus qu'on lui envoie. La première conséquence, c'est que les ICM utilisant les signaux évoqués ne nécessitent pas d'apprentissage particulier. En revanche, elles peuvent entraîner une grande fatigue pouvant altérer les performances de l'ICM ([Wolpaw et al., 2002](#), [Curran, 2003](#)). Enfin, les stimulus externes peuvent être de nature différents, que ce soit visuel, auditif ou tactile. Ce type peut s'adapter en fonction de la condition physique du sujet.

Ces réponses sont donc liées à un événement (*event-related*). La plus connue et la plus ancienne est le *Potentiel Évoqué* (PE) recueillis en Électroencéphalographie (et

puisque l'il est lié à l'apparition d'un événement, on parlera d'*Event-Related Potential* (ERP)) et *Champs magnétique Évoqué* en Magnétoencéphalographie . Ces ondes sont obtenues en moyennant un grand nombre d'essais par rapport à l'apparition du stimuli, afin d'écraser les variabilité inter-signal (bruits) et renforcer l'émergence des phénomènes communs aux essais. Parmi ces phénomènes, deux sont particulièrement exploités dans le cadre des ICM, les *Steady-State Evoked Potentials* et l'onde *P300*.

- 1.1.4.1.1 Steady-State Evoked Potentials (SSEP)** Les SSEP sont une réponse naturelle du cerveau à des stimulus envoyés à une certaine fréquence. Bien souvent, le stimuli est de nature visuelle, on parlera donc de *Steady-State Visual Evoked Potentials* (SSVEP). Généralement, les stimulus visuels sont envoyés à des fréquences comprises entre 3.5hz et 75hz et génèrent, dans le cortex visuel, des réponses aux mêmes fréquences (et harmoniques) ([Wolpaw et al., 2002](#), [Beverina et al., 2003](#)).

[Sutter \(1992\)](#) décrit une BCI basée sur les SSVEP (SSVEP-BCI) où l'utilisateur fait face à un écran composé de lettres et de symboles disposés dans une matrice 8x8. Chaque groupe de lettre est flashé à des vitesses différentes pour établir le profil standard du sujet. On demande ensuite à l'individu de choisir une lettre. En flashant de nouveau les éléments de la matrices, les réponses dans le cortex visuel vont différencier du profil standard et c'est ça qui permettra d'établir le choix de l'utilisateur. Depuis, les SSVEP ont été utilisées dans de nombreux autres systèmes, comme la sélection binaire ([Allison et al., 2008](#)), pour épeler ([Cecotti, 2010](#)), le contrôle continu d'un curseur 1D ou 2D ([Trejo et al., 2006](#)), les prothèses ([Muller-Putz and Pfurtscheller, 2008](#)) ou encore pour le jeux ([Lalor et al., 2005](#)).

Ces SSVEP-BCI nécessitent que l'individu ai, d'une part, un restant de contrôle oculo-moteur pour orienter son regard et d'autre part, un système visuelle fonctionnelle. Enfin, puisque des images sont flashées à des vitesses assez importantes, ces ICM ne correspondent pas aux personnes épileptiques.

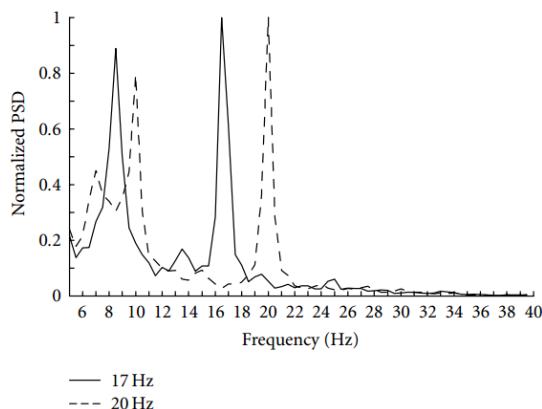


FIGURE 1.9 – Spectre de puissance d'un signal EEG contenant des SSVEP à 17Hz (ligne pleine) et à 20hz (ligne pointillées). Dans cette étude, les auteurs utilisent les SSVEP pour une prise de décision binaire permettant un contrôle d'un jeux virtuel en 3D ([Lalor et al., 2005](#))

- 1.1.4.1.2 Onde P300** Comme décrit ci-dessus, les ERP sont une réponse exogène à un stimulus extérieur pouvant être de nature visuelle, auditive ou tactile. Après moyennage à travers les essais, on peut constater l'apparition d'une série de pic de tension à alternance positive (P) et négative (N) dont la localisation temporelle est fixe (ou contenue dans un intervalle) et connue ([Wolpaw et al., 2002](#), [Beverina et al., 2003](#)).

Ces ondes sont étudiées au dessus du cortex pariétal. Par convention, leur nom dérive de l'instant temporel d'apparition. Ainsi, on parlera des ondes N100 (ou N1), de la P200 (ou P2) et enfin de la P300.

Dans le cadre des ICM, on va surtout s'intéresser à cette dernière, la P300 (bien que la N1 est en souvent également prise en compte). A l'instar des SSVEP, des objets sont aléatoirement mis en surbrillance sur un écran et on demande au sujet de choisir un de ces élément et de compter le nombre de fois que celui-ci est flashé. Enfin, lorsqu'une P300 est détectée dans l'activité neuronale, en remontant 300ms plus tôt, il est possible de retrouver l'élément mis en surbrillance à cet instant. Une autre petite beauté de la P300, c'est que son amplitude est d'autant plus forte que le sujet à réussi à dénombrer le nombre d'apparition.

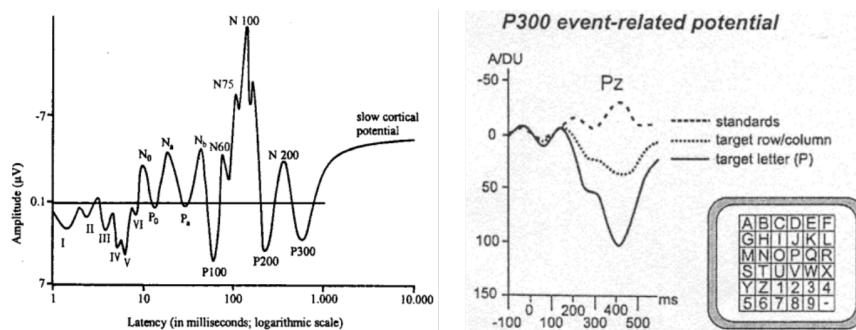


FIGURE 1.10 – (A gauche) ERP générée par un stimuli auditif et apparition des différentes ondes, (A droite) Onde P300. On constate une augmentation d'amplitude lorsque l'utilisateur se focalise sur la lettre P (target letter (P)) et target row/column, comparé au profil standard (standards) (Kübler et al., 2001)

La première ICM utilisant la P300 a été introduite par [Farwell and Donchin \(1988\)](#), [Donchin et al. \(2000\)](#). Ce système appelé le *P300-speller*, permet à l'utilisateur d'épeler des mots. Le sujet est devant un écran formé d'une matrice de lettres aléatoirement flashées. Cette BCI est toujours d'actualité et de nombreuses études continuent de la développer ([Vaughan et al., 2006](#), [Hoffmann et al., 2008](#)). L'utilité de la P300 pour contrôler un fauteuil roulant a également été exploré ([Vanacker et al., 2007](#), [Pires et al., 2008](#)). Enfin, [Mugler et al. \(2010\)](#) décrivent une ICM testée sur des sujets sains et atteints de SLA, permettant de contrôler un navigateur internet.

Les systèmes présentés utilisent des stimulus visuels. Toutefois, d'autres ICM reposent sur des stimulus auditifs ([Sellers et al., 2006](#), [Sellers and Donchin, 2006](#), [Furdea et al., 2009](#), [Schreuder et al., 2010](#)) ou tactiles ([Muller Putz et al., 2006](#), [Brouwer and Van Erp, 2010](#)).

1.1.4.2 Signaux spontanés

Les signaux spontanés correspondent à un ensemble de signaux cérébraux que l'utilisateur peut apprendre à moduler. Cet apprentissage, qui peut s'avérer assez long, va permettre de contrôler une Interface Cerveau-Machine .

Parmi ces signaux, on rencontre les *potentiels corticaux lents* ainsi que les *rythmes sensorimoteurs*. Ces derniers sont largement plus présents dans la littérature ICM. Avant tout, pour apprendre à moduler sont activité neuronale, l'utilisateur devra utiliser une stratégie mentale

1.1.4.2.1 Stratégies mentales pour le contrôle d'une ICM Pour assimiler le contrôle des signaux spontanés, il est nécessaire que l'utilisateur passe par une phase d'apprentissage pouvant être soit autonome, soit guidée via l'utilisation de l'imagerie motrice (Curran, 2003).

1. Apprentissage autonome (*operant conditioning, implicit learning ou operant self-control*) : le choix de la stratégie mentale est laissé à l'utilisateur. C'est à lui de trouver celle qui lui convient le mieux. Dans ce cas, le système doit impérativement fournir un feedback au sujet afin que celui-ci comprenne comment la machine.
2. Imagerie motrice : pour cette stratégie mentale, on demandera donc à l'utilisateur d'imaginer des mouvements qui lui seront imposés en amont (cf. 1.1.4.2.2)

Ces stratégies mentales conditionnent le mode d'interaction de l'utilisateur avec la machine. Il est fréquent que, dans les premiers stades de l'entraînement, les sujets soumis à l'apprentissage autonome utilisent naturellement l'imagerie motrice (Wolpaw et al., 1991, Birbaumer et al., 1999).

L'*operant conditioning* demande un long apprentissage (de plusieurs semaines à années) mais les systèmes l'utilisant rapportent de bonnes performances et une grande stabilité (Fetz, 1969, Wolpaw et al., 1991, Birbaumer et al., 1999, Wolpaw and McFarland, 2004a, Birbaumer, 2006, Vaughan et al., 2006, Wolpaw, 2007). Certaines tâches motrices peuvent ne pas convenir aux personnes ayant un déficit moteur de longue date ou depuis la naissance tout comme les tâches visuelles pour les personnes aveugles de naissance (Curran, 2003). Donc, autre avantage de l'apprentissage autonome, il permet de prendre en compte la préférence ou le confort d'utilisation qui pourrait jouer un rôle dans les performances de l'ICM (Pfurtscheller et al., 2000).

1.1.4.2.2 Imagerie motrice L'imagerie motrice (IM) est définie par la représentation d'une action qui n'est pas suivie de son exécution. Dans les sections précédentes, nous avons vu que l'IM permettait de contrôler une Interface Cerveau-Machine. Les substrats neuronaux mis en jeu lors de l'imagination d'un mouvement sont sensiblement les mêmes que lors de l'exécution de ce mouvement, ce qui permet donc aux personnes à mobilité réduite de solliciter les aires motrices sans la possibilité d'accomplir le mouvement.

Cette section a pour objectif de raffiner l'imagerie motrice, c'est-à-dire introduire les différents types d'imagerie, différentes applications et enfin, leur utilisation pour les ICM.

1. **Types d'IM :** les différents types d'imagerie reposent sur l'exploitation de modalités sensorielles, c'est-à-dire sur les sens (Kosslyn et al., 1990). On distingue donc l'imagerie visuelle, tactile, auditive et olfactive. L'imagerie visuelle se décompose en deux sous-divisions : interne (ou à la première personne) et externe (ou à la troisième personne) (Ruby and Decety, 2001, Jackson et al., 2006, Lorey et al., 2009). Dans le premier cas, on est directement acteur de l'action, donc par exemple, on pourra imaginer effectuer un mouvement de bras de manière similaire à une réelle exécution. En imagerie visuelle externe, on est spectateur d'un mouvement pouvant être effectué par une autre personne, ou par soi-même. À ces différents types d'imagerie, s'ajoute l'imagerie kinesthésique basée sur des informations proprioceptives. Bien que à priori proches, imagerie visuelle et kinesthésique sollicitent des substrats neuronaux différents (Solodkin, 2004, Guillot et al., 2009).

Si une personne imagine avoir une balle dans la main, c'est de l'imagerie

visuelle interne. Si elle imagine que cette balle est tenue par une tierce personne, c'est de l'imagerie externe. Enfin, si la personne imagine les sensations que peut procurer les propriétés de cette balle (comme sa texture, sa malléabilité, son poids ou sa taille) c'est de l'imagerie kinesthésique. De nombreuses études ont comparé les différents types d'imagerie permettant de mettre en valeur les réseaux associés (Jiang et al., 2015, Seiler et al., 2015).

2. **Utilisation de l'IM :** dans le paragraphe précédent nous avons vu que l'IM peut être utilisée comme stratégie mentale pour le contrôle d'une ICM (cf. 1.1.4.2.1). L'IM peut également être utilisée pour faciliter et améliorer un apprentissage, notamment pour la pratique sportive de haut niveau (Driskell et al., 1994, Guillot et al., 2008, Schuster et al., 2011, Di Rienzo et al., 2016), pour la réhabilitation motrice (Jackson et al., 2001, Sharma et al., 2006, de Vries et al., 2011, Malouin et al., 2013, Di Rienzo et al., 2014).

Dans le cadre des Interface Cerveau-Machine, Neuper et al. (2005) ont étudié le décodage de différentes modalités d'imagerie en comparaison avec le repos. L'intérêt de cette étude et de renseigner sur le type d'IM a privilégié pour le contrôle d'une ICM. Pour cela, les sujets effectuent différentes tâches avec une balle : soit ils exécutent des mouvements (ME) (pression continue), soit ils observent une main animée (OOM), soit ils imaginent les sensations procurées par cette balle (MIK) (imagerie kinesthésique) soit ils imaginent une main effectuant des mouvements sous forme de film (MIV) (imagerie visuelle). Neuper et al. (2005) montre alors que le décodage avec la condition MIK (67%) est nettement supérieur à celui atteint avec la condition MIV (58%). De plus, le décodage MIK se focalise essentiellement autour des aires sensorimotrices, tout comme pour l'exécution, alors qu'il n'y a pas de pattern réellement émergent pour le MIV. En conclusion, les auteurs conseillent plutôt l'utilisation de l'imagerie kinesthésique à l'imagerie visuelle comme stratégie mentale pour le contrôle d'une ICM.

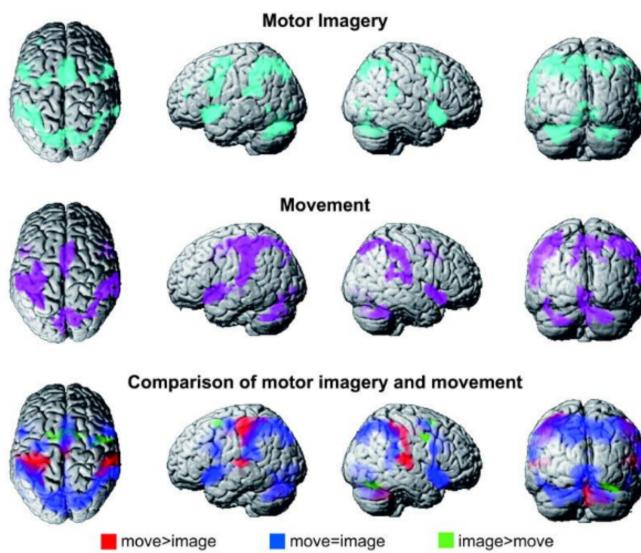


FIGURE 1.11 – Comparaison des aires actives lors d'un mouvement imaginé ou exécuté (Hanakawa et al., 2008)

- 1.1.4.2.3 **Potentiel corticaux lents** Les potentiels corticaux lents (ou *Slow Cortical Potential* (SCP)) sont des variations lentes du potentiel cortical qui ont lieu entre 500ms et 10s (Birbaumer et al., 1990, Birbaumer, 1997, Birbaumer et al., 2003, Birbaumer, 1999,

Kleber and Birbaumer, 2005). Un utilisateur peut apprendre à contrôler l'amplitude de ces signaux, notamment grâce à un bio-feedback où l'utilisateur voit son activité se moduler en temps réel sur un écran. les SCP peuvent prendre des valeurs positives, généralement lors d'un mouvement ou tout autre fonction impliquant une activation corticale, ou négative lors d'une réduction de l'activité corticale (Rockstroh, 1989, Birbaumer, 1997, Wolpaw et al., 2002).

Les ICM basées sur les SCP vont nécessiter un seuil et, en fonction de ce seuil, le sujet module son activité lui permettant un contrôle binaire (Birbaumer et al., 1999). Toutefois, l'apprentissage peut s'avérer extrêmement long. Le dispositif de traduction de pensées (Kübler et al., 1999) (*Thought Translation Device (TTD)*) , est un appareil destiné à entraîné des sujets puis à tester leur apprentissage pour épeler des mots (Perelmouter et al., 1999, Birbaumer et al., 2003). Le software est développés en C++ et utilise des outils de *BCI2000*, plate-forme de développement du groupe central de Wadsworth (Wolpaw et al., 2003, Schalk et al., 2004). Le TTD a été développé pour les sujets complètement paralysés (LIS et CLIS). Tout d'abord, les sujets s'entraînent de manière autonome à moduler leur SCP notamment grâce à un retour visuel et auditif et un renforcement positif (visage souriant et musique lorsque le contrôle est réussi). Après cette phase d'apprentissage machine, les sujets sont ensuite testés pour sélectionner des lettres ou des mots (Birbaumer et al., 2000, 2003).

Les SCP ont également été testées à des fins cliniques. Rockstroh et al. (1993), Kotchoubey et al. (1998) ont montré qu'après un an et demi d'apprentissage autonome à moduler positivement et négativement les SCP, des sujets atteints d'épilepsies pharmacorésistantes ont vu leur crise diminuer de 50% en moyenne (certains sujets n'avaient plus aucune crise tandis que d'autres n'ont eu aucun changement).

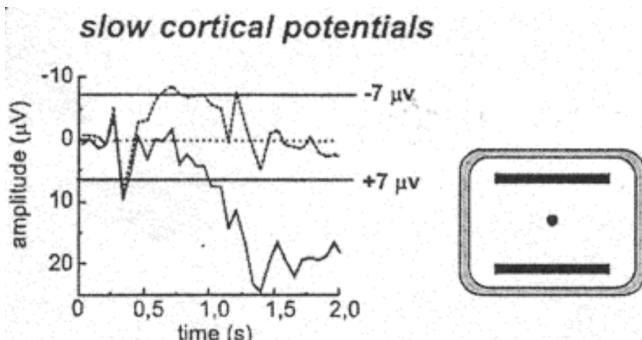


FIGURE 1.12 – Feedback visuel des SCP renvoyées par le TTD - A droite, un curseur peut se déplacer entre les deux objectifs (en haut et en bas). Les négativités corticales bougent le curseur vers le haut (ligne du haut en pointillé) à l'inverse, les positivités corticales permettent de faire descendre le curseur (ligne pleine du bas). Toute modulation de $7\mu V$ est considérée comme une réussite (Kübler et al., 2001)

- 1.1.4.2.4 Le Bereitschaftspotential** Le Bereitschaftspotential (BP), ou *readiness potential*, est un potentiel moteur focalisé principalement dans la pré-aire motrice supplémentaire (pré-SMA) et SMA (Kornhuber and Deecke, 1965, Ball et al., 1999, Brunia and Van Boxtel, 2000). Le BP comprend deux composantes dont la première, la composante précoce (ou *early BP*), commence environ deux secondes avant le début du mouvement qui est ensuite suivie d'une pente négative tardive (ou *late BP*), environ 400ms avant le début du mouvement (Shibasaki and Hallett, 2006). Il a été montré que le BP dépend de paramètres de mouvements tels que l'état de préparation ou encore la répétition et précision de mouvement (Birbaumer et al., 1990). Enfin, le

BP est d'avantage présent lors de mouvements auto-initiés, comparés à des mouvements imaginés ou basés sur un *cue* externe (Deiber et al., 1999, Jenkins et al., 2000, Jankelowitz and Colebatch, 2002).

- 1.1.4.2.5 Rythmes sensorimoteurs (RSM)** Les RSM (ou *Sensorimotor rhythms* (SMR)) correspondent à l'amplitude de signaux, au dessus du cortex sensorimoteur, dans des bandes de fréquences spécifiques. Les plus fréquents sont les rythmes μ 8-13hz et β 13-30hz. Un utilisateur peut apprendre à moduler l'amplitude de son activité cérébrale dans ces bandes pour contrôler une Interface Cerveau-Machine notamment par le biais de l'imagerie motrice (cf. 1.1.4.2.2). On appelle *Event-Related Synchronization* lorsque l'amplitude augmente et *Event-Related Desynchronization* lorsqu'elle diminue (Pfurtscheller and Lopes da Silva, 1999, Pfurtscheller et al., 2008). Ces rythmes ont très largement été exploités pour contrôler un BMI, que ce soit pour déplacer un curseur dans une, deux ou trois dimensions (Wolpaw and McFarland, 2004b, McFarland et al., 2008, Kayagil et al., 2009, McFarland et al., 2010, Doud et al., 2011), pour épeler (Neuper et al., 2006, Vaughan et al., 2006), pour contrôler une prothèse (Pfurtscheller et al., 2000, Müller-Putz et al., 2005, McFarland and Wolpaw, 2008) ou un fauteuil roulant (Tanaka et al., 2005, Galán et al., 2008b)

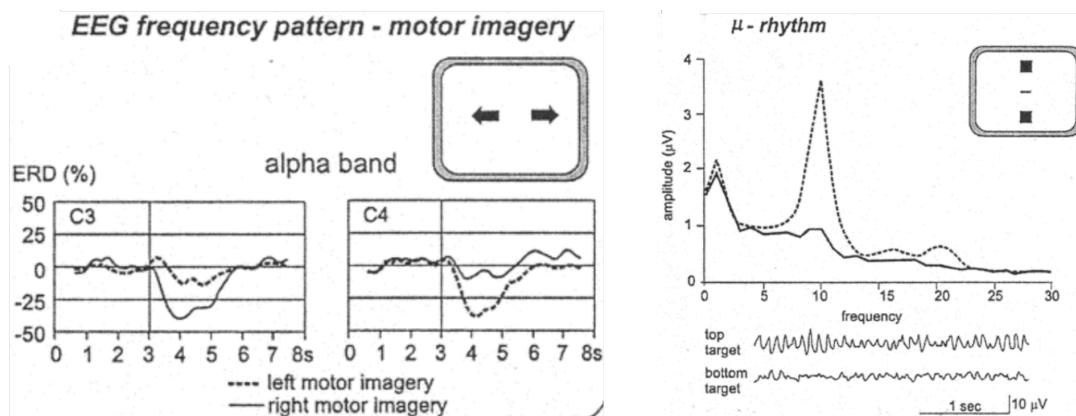


FIGURE 1.13 – (A gauche) Utilisation de l'imagerie motrice pour contrôler la direction d'un curseur et tracé de l'activité neuronale pour les électrodes C3 (hémisphère gauche) et C4 (hémisphère droit) en utilisant l'activité neuronale contenue dans la bande α - L'utilisateur imagine des mouvements des mains droite ou gauche pour déplacer le curseur. L'imagination d'un mouvement de main gauche entraîne une ERD dans l'hémisphère droit (C4) mais l'activité est maintenue sur C3 (A droite) Modulation du μ au dessus de C3-C4 pour contrôler un curseur vers le haut (augmentation du μ , ligne pointillée) ou vers le bas (baisse du μ , ligne pleine) (Kübler et al., 2001)

- 1.1.4.2.6 Autres marqueurs** Les marqueurs présentés ci-dessus sont ceux que l'on retrouve le plus largement à travers la littérature. Toutefois, d'autres marqueurs de l'activité neuronale sont étudiés essentiellement d'un point de vue neuro-scientifique, c'est-à-dire pour améliorer la compréhension des phénomènes physiologiques.

Décomposition phase-amplitude : tout signal temporel réel peut être décomposé en un signal d'amplitude (ou enveloppe car elle va suivre les maxima du signal) et un signal de phase qui indique la situation instantanée d'un cycle (pic, creux, passage à zéro ...). La transformée d'Hilbert permet de décomposer ainsi n'importe quel signal (4.1.1.2).

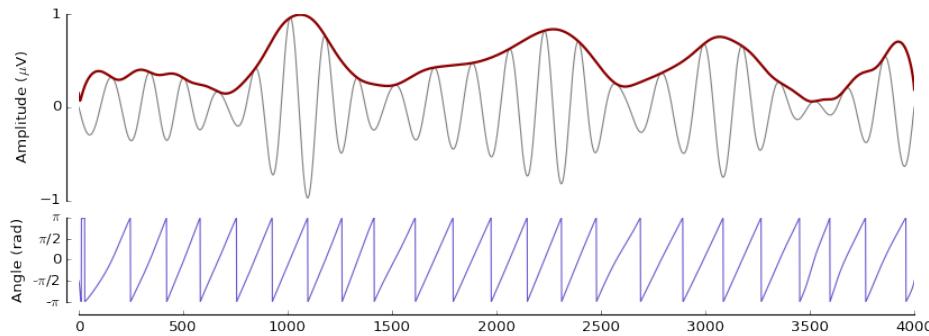


FIGURE 1.14 – (En gris) Un signal original, (En rouge) L'amplitude du signal, (En bleu) la phase instantanée

Phase : si l'amplitude a très largement été étudiée, que ce soit dans le cadre des ICM, du décodage et de son implication physiologique, la phase reste à l'heure actuelle l'objet d'un nombre plus restreint d'études. Toutefois, quelques études ont montré l'implication de la phase basse-fréquence dans l'encodage neuronal de mouvements (Hammer et al., 2013, 2016).

Couplage inter-fréquences locaux ou à distance : ou *Cross-Frequency Coupling* (CFC) regroupe un ensemble de marqueurs qui, contrairement à ceux présentés ci-dessus, étudient une forme de corrélation entre deux signaux pouvant être soit locaux (comme provenant d'une même électrode) ou à distance pour évaluer des synchronisations d'aires cérébrales. Ces *features* sont étudiés dans des bandes de fréquences particulières et vont donc nécessiter une filtrage en amont. Ensuite, on pourra extraire de ces signaux filtrés les informations de phase et d'amplitude afin d'étudier différentes formes de couplage :

Couplage à distance phase-phase : le *Phase-Locking Value* (PLV) (Lachaux et al., 1999, 2000) est un des outils permettant de déterminer la synchronisation de phase. Le couplage inter-fréquence phase-phase semble jouer un rôle dans la communication inter-structures (Fries, 2005, Gregoriou et al., 2009, Siegel et al., 2009, van Elswijk et al., 2010)

Couplage à distance amplitude-amplitude : bien que le rôle physiologique du couplage inter-fréquences amplitude-amplitude soit encore incertain, de tel couplages ont été décrits dans la littérature (Friston, 1997, Shirvarkar et al., 2010, Siegel et al., 2009)

Couplage local ou à distance phase-amplitude : ou *Phase-amplitude coupling* (PAC), fait intervenir des signaux pris dans deux bandes de fréquence et permet d'évaluer la façon dont ces signaux évoluent l'un avec l'autre. Plus précisément, on considère un signal dans les basses fréquences (BF), typiquement dans les bandes delta, thêta ou alpha et on extrait la phase de ce signal. Ensuite, on considère l'amplitude d'un signal haute fréquence (HF), souvent dans la bande gamma. Le PAC renseigne si la phase des BF et l'amplitude des HF évoluent de manière synchrones. A noter ici que il n'est pas question d'inférer une relation de cause/conséquence entre la phase des BF et l'amplitude des HF. Autrement dit, le PAC ne permet pas de conclure que l'un vient moduler l'autre. Plusieurs outils méthodologiques permettant une mesure du PAC ont été proposés (cf. 4.1.4)

Le rôle physiologique du PAC est encore discuté (Canolty and Knight, 2010,

Hyafil et al., 2015) tout comme son implémentation méthodologique (Aru et al., 2015), mais il a été observé dans des tâches variées (Bruns and Eckhorn, 2004, Voytek, 2010, Soto and Jerbi, 2012), dans la maladie de Parkinson (de Hemptinne et al., 2013), dans la prise de risques (Lee and Jeong, 2013) ou dernièrement dans l'encodage mémoriel (Lega et al., 2016). Enfin, Yanagisawa et al. (2012b) ont pu montrer l'existence d'un couplage alpha-gamma dans le cortex sensorimoteur durant une période de repos. Ensuite, ce couplage diminue avec l'exécution de tâches motrice (mouvements de saisie, de pincée et d'ouverture de main). Enfin, cette étude a montré que ce couplage alpha-gamma, dans cette région sensorimotrice, ne permettait pas de décoder ces différents types de mouvements. A noter que généralement l'amplitude est prise dans la bande gamma mais Cohen et al. (2008) ont démontré l'existence d'un couplage entre la phase du delta et thêta avec l'amplitude de l'alpha et du gamma dans la prise de décision.

1.1.4.3 Enregistrement : ICM invasives et non-invasives

1. *ICM invasives ou directes* : basées sur un enregistrement invasif de l'activité neuronal, la qualité du signal étant excellente les possibilités d'exploitations et d'améliorations futures sont grandes. En revanche, cela va avec les risques que comporte la chirurgie.
2. *ICM non-invasives ou indirectes* : de la même manière, ces ICM utilisent des enregistrements non-invasifs. Le terme *indirecte* signifie que l'on enregistre pas directement des décharges de neurones mais un phénomène lié à une population de neurones (consommation d'oxygène pour Imagerie par Résonance Magnétique fonctionnelle , champs électrique et magnétique pour l'EEG et la MEG). Ces ICM sont les plus répandues et représentent un enjeu majeure de part leur accessibilité.

1.2 Data-mining EN NEUROSCIENCES

1.2.1 Exploration des données

1.2.2 Outils de validation

ICM ET NEUROPHYSIOLOGIE

Cette chapitre a pour objectif d'introduire les concepts neurophysiologiques sous-jacents au contrôle et à la définition d'une Interface Cerveau-Machine . Le but sera d'expliquer l'origine et la composition des différents types de signaux cérébraux permettant à un utilisateur de piloter une ICM. Pour cela, nous verrons tout d'abord les bases physiologiques liées à la motricité, puis de là, nous verrons quels types de signaux sont couramment extraits et exploités des principales régions motrices pour le contrôle d'une ICM.

2.1 BASES PHYSIOLOGIQUES LIÉES À LA MOTRICITÉ

Cette section servira avant tout à identifier les principaux acteurs de la planification et de l'action motrice. Pour cela, on introduira les notions nécessaires sur le cortex ainsi que les principales aires sollicitées dans le cadre des ICM. Puis, de ces aires, nous verrons le principe du rythme cérébral qui permettra ensuite de justifier les principaux processus neuronaux liés à la motricité.

2.1.1 Notions sur le cortex

Le cerveau est l'organe le mieux protégé du corps. Tout d'abord, la peau est une première barrière avec l'extérieur. Ensuite, l'os (le crâne) assure une protection mécanique c'est-à-dire qu'il va protéger en cas de coups ou de chocs. Viennent ensuite les méninges (dure-mère, l'arachnoïde et la pie-mère), qui sont des membranes enveloppant le système nerveux central (SNC) et qui vont permettre d'une part d'amortir les chocs et d'autre part, empêcher le cerveau de s'abîmer avec l'intérieur du crâne.

La couche suivante est appelée cortex, mesure entre 1 et 5mm d'épaisseur et correspond au corps cellulaire des neurones. Le cortex se divise en quatre régions :

- Lobe frontal : lié à la planification, la prise de décision et à l'exécution motrice
- Lobe occipital, pour le traitement de la vision, temporal pour l'audition et somatosensorielle pour le touché.
- Lobe pariétal, qui est considéré comme un cortex associatif c'est-à-dire qu'il joue un rôle important dans l'intégration des informations sensorielles.

Le cortex moteur, situé dans le lobe frontal, est un ensemble d'aires destinées à la planification et à l'exécution de mouvements volontaires. Il se compose du *cortex pré-moteur* qui va planifier et organiser les mouvements en fonction des informations sensorielles et qui seront ensuite exécutés par le *cortex moteur primaire*. Enfin, l'*aire motrice supplémentaire* (SMA) est impliquée dans la planification des mouvements complexes et dans la coordination.

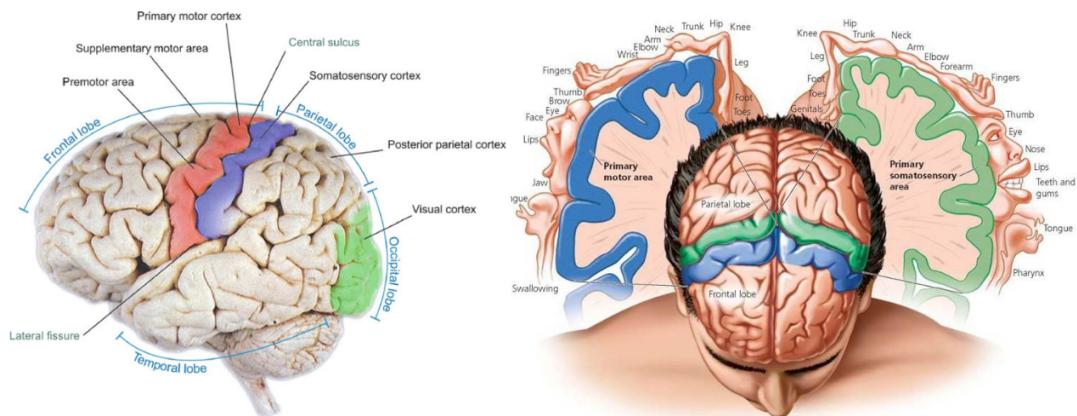


FIGURE 2.1 – (A gauche) - Principaux lobes constitutants le cortex et localisation de la SMA, du cortex pré moteur et moteur primaire (Graimann et al., 2009), (A droite) - Vue en coupe du cortex moteur et sensorimoteur permettant le contrôle de différentes régions (<http://schoolbag.info/biology/humans/9.html>)

2.1.2 L'activité rythmique et lien avec la motricité

L'activité rythmique correspond à l'activité neuronale prise dans des bandes de fréquence précises (Niedermeyer, 2004, Niedermeyer and da Silva, 2005) (bien qu'elle peut différer légèrement d'une publication à l'autre). Leur extraction se fera donc à l'aide de filtrage.

2.1.2.0.1 Définition et rôle fonctionnel des rythmes cérébraux : l'apparition de ces rythmes résulte d'une synchronisation/désynchronisation d'une population large de neurones et vont permettre de caractériser l'état cognitif d'un individu. On peut distinguer six rythmes cérébraux ayant des propriétés spatio-temporelle différentes. La littérature attribue un très grand nombre de rôles fonctionnels à ces bandes. Les descriptions ci-dessous illustrent une partie des rôles qui leur sont affectés :

Delta ($\delta \in [2, 4] \text{Hz}$) : principalement présentes chez les très jeunes enfants, ces rythmes très lents et de forte amplitude sont particulièrement présents durant le sommeil profond (Amzica and Steriade, 1998, Silber et al., 2007).

Theta ($\theta \in [5, 7] \text{Hz}$) : les rythmes thêta sont proéminents pour la mémorisation à long terme (Klimesch, 1999) ou dans les états de somnolence et d'hypnose (Schacter, 1977).

Alpha ($\alpha \in [8, 13] \text{Hz}$) : la puissance dans la bande alpha est réputée pour augmenter lorsque les yeux sont fermés (Berger, 1929). Ce rythme est également important dans la mémorisation (Klimesch, 1999). Markand (1990), Klimesch et al. (2007) décrit plus spécifiquement les rôles supposés de l'alpha.

Mu ($\mu \in [8, 13] \text{Hz}$) : bande de fréquence identique à celle de l'alpha mais le rythme μ correspond en fait à l'alpha pris dans les aires sensorimotrices (Markand, 1990). Un des phénomènes important avec ce rythme qui sera particulièrement exploité par les ICM, c'est la baisse occasionnée lors de tâche motrice ou d'imagerie motrice dans l'hémisphère contralatéral (Salmelin and Hari, 1994, Crone et al., 1998, Pfurtscheller and Lopes da Silva, 1999, Pfurtscheller et al., 2008).

Beta ($\beta \in [13, 30] \text{Hz}$) : tout comme le rythme μ , on observe une baisse dans la bande β lors de tâche d'exécution motrice au dessus du cortex moteur sauf que celle-ci est suivie par une augmentation une à deux secondes après la

fin du mouvement (Pfurtscheller and Berghold, 1989, Cassim et al., 2001). Ce phénomène s'appelle le rebond bêta.

Gamma ($\gamma \geq 40\text{Hz}$) : les oscillations gamma sont présentes en réponse à un stimulus sensoriel, lors de tâche motrices (Crone, 1998, Tallon-Baudry and Bertrand, 1999) ou dans des tâches de recherche visuelle (Tallon-Baudry et al., 1997). La bande haute du γ sera limitée par la fréquence d'échantillonnage (critère de Shannon) et donc, par la technique d'enregistrement (l'ECoG et la SEEG permettent d'étudier des phénomènes plus large bandes que la MEG ou l'EEG par exemple)

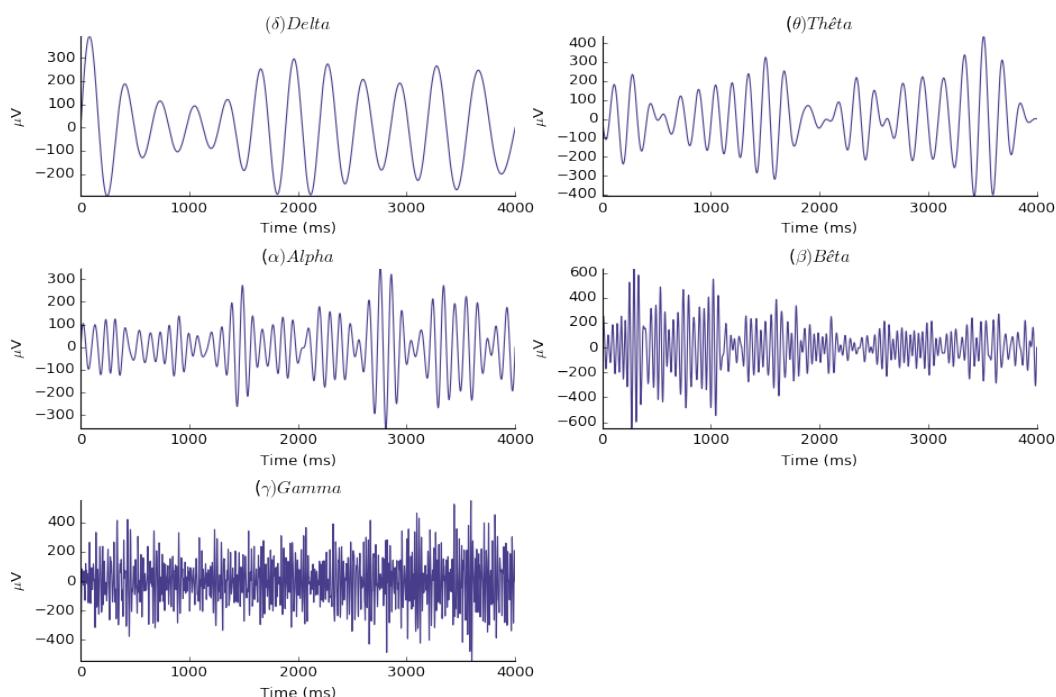


FIGURE 2.2 – Exemple d’activités rythmiques dans les différentes bandes de fréquences pour un essai unique issue de l’activité sEEG

2.2 DÉCODAGE DIRECTIONNEL DES MEMBRES SUPÉRIEURS

Que ce soit pour déplacer un curseur, pour le contrôle d'une prothèse ou pour toute autre application, le décodage de directions est au centre du fonctionnement d'un bon nombre d'Interface Cerveau-Machine . Pour ce faire, les équipes utilisent bien souvent une tâche appelée *center-out* où le sujet bouge son bras du centre vers une direction imposée, que ce soit avec le curseur d'une souris ou avec un joystick. A noter ici que certaines équipes ce sont également intéressées au moment où le sujet prépare le mouvement. Le but sera donc, en *offline*, d'essayer de retrouver la direction effectuée par le sujet que ce soit pendant la période d'exécution ou de préparation en utilisant l'activité neuronale seule.

Le décodage directionnel fait intervenir trois principales composantes :

1. L'aire cérébrale étudiée (cortex moteur primaire, pré moteur, pariétal, préfrontal...). Ces aires peuvent apporter des informations complémentaires à différents instants temporels durant la tâche, c'est pourquoi elles sont bien souvent combinées pour améliorer l'acuité de décodage.

2. L'optimisation des marqueurs : le travail en *offline* permet une exploration plus vaste des marqueurs puisqu'il n'y a pas la contrainte d'application temps-réel.
3. L'optimisation des paramètres de classification : de la même façon, on va pouvoir tester des classificateurs avec des méthodologies plus lourdes.

2.2.1 Décodage directionnel

L'utilisation des micro-électrodes chez le singe ont permis de mettre en valeur le *directional tuning* (Georgopoulos et al., 1982, 1986, Georgopoulos and Carpenter, 2015), c'est-à-dire que le taux de décharges dépend de la direction et donc, un neurone pourra afficher une direction préférentielle dans laquelle il déchargera davantage. Le *directional tuning* a également été étudié avec le signal LFP chez le singe (Mehring et al., 2004, Rickert, 2005). Chez l'homme, l'utilisation du signal filtré basse fréquence en ECoG (Mehring et al., 2004, Ball et al., 2009) et l'amplitude prise dans différentes bandes de fréquences (Leuthardt et al., 2004, Ball et al., 2009) ont aussi permis de mettre en évidence des modulations spécifiques en fonction des directions permettant un décodage. Pour quatre directions, Ball et al. (2009) a obtenu des décodages supérieurs à 80% en ECoG, en combinant les électrodes du cortex moteur. Plus récemment, Gunduz et al. (2016) montre des décodages significatifs en utilisant l'activité gamma en ECoG que ce soit pour décoder l'exécution ou la préparation de mouvements effectués dans 8 directions (avec un maximum de 30% pour décoder la préparation et 50% pour l'exécution dans les quatre directions en combinant jusqu'à 30 électrodes ECoG).

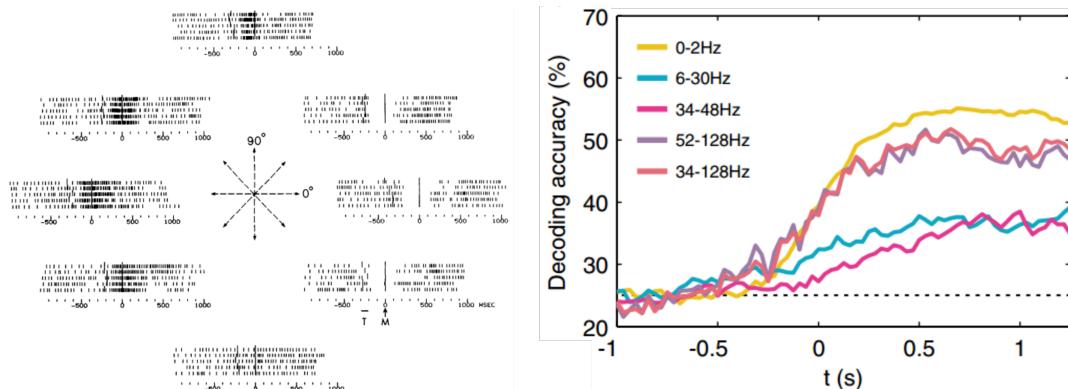


FIGURE 2.3 – (A gauche) Cinq essais montrant le taux de décharge d'un neurone moteur en fonction de directions de mouvements (Georgopoulos et al., 1982). On constate que ce neurone affiche une préférence pour les directions vers la gauche (A droite) Évolution temporelle du décodage de quatre directions utilisant l'amplitude dans différentes bandes de fréquences pour des électrodes prises dans le cortex moteur primaire (Ball et al., 2009)

Le décodage directionnel a également été exploré dans le cadre de données non-invasives, en EEG et MEG (Waldert et al., 2007, 2008). Dans cette étude, l'auteur montre des décodages jusqu'à 67% pour décoder quatre directions de mouvements. Il est intéressant de noter que ces décodages significatifs ont été obtenus en utilisant le signal filtré dans les basses fréquences et que la puissance dans les bandes bêta et gamma ne semblent pas décoder. Enfin, l'EEG et la MEG fournissent des résultats similaires et la combinaison des deux améliore peu les résultats dans cette étude. Hammon et al. (2008) ont également montré des décodages significatifs pour décoder la directionnalité que ce soit pendant l'exécution, ou la préparation motrice.

2.2.2 Prédiction continu de la cinétique du mouvement

Les études ci-dessus tentent de décoder les directions, c'est-à-dire d'essayer de retrouver en *offline* les directions effectuées par le sujets. D'autres articles tentent de décoder la position, la vitesse ou l'accélération du mouvement, notamment via l'utilisation du filtre de Kalman (Wu et al., 2002, 2003, 2004, 2006, Pistohl et al., 2008, Li et al., 2009)

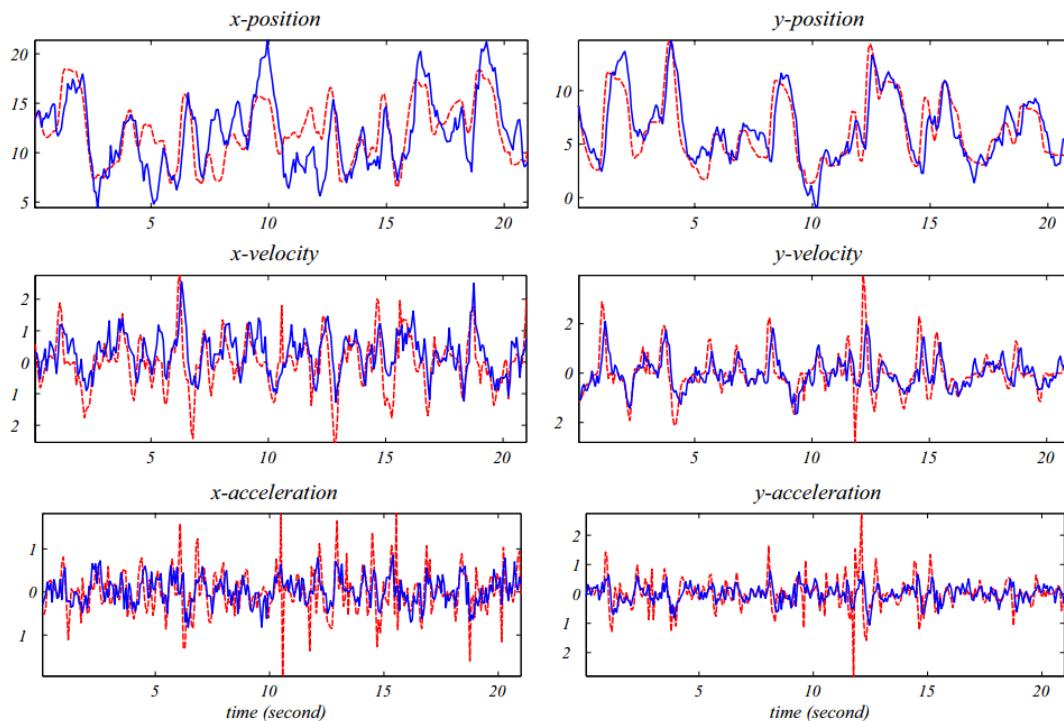


FIGURE 2.4 – Décodage continu de la cinétique d'un mouvement 2D (position/vitesse/accélération) (Wu et al., 2003). Le mouvement réellement effectué (en rouge) est reconstruit en utilisant l'activité spike et le filtre de Kalman (en bleu)

Conclusion du chapitre 2

Les Interface Cerveau-Machine peuvent donc être contrôlées par deux types de signaux cérébraux. D'une part, les *réponses évoquées* (SSEP, P300...), correspondant à des réponses cérébrales suite à un stimulus externe. L'avantage de ce type d'ICM est qu'elles ne nécessitent pas de période d'apprentissage. D'autre part, les *réponses spontanées* (SCP, rythmes sensorimoteurs), correspondant à une activité cérébrale que le sujet peut volontairement modifier. Pour apprendre à modifier l'activité de son cerveau, l'utilisateur pourra soit apprendre de façon autonome ou via les différentes modalités d'imagerie motrice (visuelle, kinesthésique...).

OBJECTIFS DE LA THÈSE

3

Durant cette thèse, nous avons principalement utilisé les données intracrâniennes issues d'une tâche motrice (cf. 6) afin d'étudier les axes suivants :

Utilisation des outils de machine learning : les méthodes d'apprentissage machine ont été utilisées comme un outil de validation pour explorer les différences entre des états moteur.

Comparatif d'états moteur : dans un premier temps, nous avons chercher à raffiner la connaissance des substrats neuronaux propres à un état de repos, de préparation ou d'exécution motrice. Puis, dans un second temps, nous avons étudié le décodage directionnel durant ces phases de préparation ou d'exécution motrice.

Exploration et amélioration des marqueurs de l'activité cérébrale : pour comprendre ce qui caractérise ces états, nous avons extrait de l'activité neuronale une variété relativement large de marqueurs spectraux (principalement la puissance, la phase et le couplage phase-amplitude). De plus, nous avons fait varier de nombreux paramètres liés à ces attributs (taille et emplacement des fenêtres temporelles et fréquentielles, comparatif de différentes méthodologies propres à chaque *feature*...)

Exploration des régions non-motrices : les données intracrâniennes nous ont également permis d'explorer si des régions non-motrices peuvent discriminer certains états ou si leur association avec des régions motrices peut constituer un gain de performances.

Optimisation des paramètres de machine learning : de nombreux paramètres liés à la classification ont été pris en compte afin d'évaluer leur influence sur le décodage (choix et optimisation de l'algorithme de classification et de validation croisée, stratégie de *multi-features*, évaluation statistique...)

Implémentation et mise à disposition d'un ensemble de méthodes : enfin, l'essentiel des méthodes et des outils présentés et utilisés durant cette thèse ont été codé en Python puis mis à la disposition de tous sur un compte Github. Ces outils comprennent l'extraction de marqueurs, leur classification ainsi qu'un ensemble de fonction pour visualiser les résultats.

Tous ces différents paramètres ont permis d'avoir une compréhension assez fine des impacts méthodologiques mais ont aussi forcé chaque étude à être hautement dimensionnelle. Cette thèse permettra peut-être aux lecteurs, et je l'espère, de commencer son exploration avec une nombre plus restreint de dimensions.

MÉTHODOLOGIE

Cette partie méthodologique sera divisée en deux grandes sous parties visant à présenter :

1. **L'extraction des features** : présentation des méthodes utilisées dans le cadre de l'extraction d'attributs issus de l'activité neuronale. De manière générale, nous avons étudiés des attributs spectraux comprenant :
 - Phase et puissance spectrale
 - Attributs de couplage
2. **Le machine learning** : présentation des principaux algorithmes testées dans le cadre du décodage de l'activité neuronale

4.1 EXTRACTION DES FEATURES

Comme nous l'avons décrit précédemment, l'objectif du décodage de l'activité neuronale est d'arriver à extraire des signaux cérébraux une information suffisamment pertinente pour pouvoir discriminer différents types de classes (exemple : mouvement vers la gauche Vs droite).

Tout les attributs testés dans le cadre de cette thèse sont des attributs spectraux, donc issus de bandes de fréquences. La plupart de ces outils partagent donc une partie méthodologique commune à savoir, le filtrage. De plus, la plupart sont extraits en utilisant la transformée d'Hilbert. Pour éviter une redondance à travers les attributs, nous allons tout d'abord introduire quelques pré-requis.

4.1.1 Pré-requis

4.1.1.1 Filtrage

L'intégralité des filtrages dans cette thèse ont été effectués avec la fonction *eegfilt* (qui a ensuite été reproduite pour le passage à python). De plus, afin d'éviter tout phénomène de déphasage, la fonction *filtfilt* a été systématiquement utilisée afin que le filtre soit appliqué dans les deux sens. Si cette dernière fonctionnalité n'est pas forcément indispensable dans le cadre d'un calcul de puissance, elle est absolument nécessaire pour un calcul de couplage phase-amplitude .

L'ordre du filtre présenté au dessus dépend de la fréquence de filtrage. Il a systématiquement été calculé en utilisant la méthode décrite par [Bahramisharif et al. \(2013\)](#) :

$$FiltOrder = N_{cycle} \times f_s / f_{oi} \quad (4.1)$$

où f_s est la fréquence d'échantillonnage, f_{oi} est la fréquence d'intérêt et N_{cycle} est

un nombre de cycles définit par $N_{cycle} = 3$ pour les oscillations lentes et $N_{cycle} = 6$ pour les oscillations rapides.

4.1.1.2 Transformée d'Hilbert

Transformée permettant de passer un signal temporel $x(t)$ du domaine réel au domaine complexe. Le signal peut ensuite s'écrire $x_H(t) = a(t)e^{j\phi(t)}$ où $a(t)$ est l'amplitude et $\phi(t)$, la phase. Cette transformation est particulièrement exploitée car le module de $x_H(t)$ permet de récupérer l'amplitude et la phase est obtenue en prenant l'angle de $x_H(t)$.

4.1.1.3 Transformée en ondelettes

La transformée en ondelettes ([Tallon-Baudry et al., 1997](#), [Worrell et al., 2012](#)) permet de décomposer un signal dans le domaine temps-fréquence. La décomposition en ondelettes d'une fonction f est définie par :

$$f(a, b) = \int_{-\infty}^{\infty} f(x) \bar{\psi}_{a,b} dx \quad (4.2)$$

Où ψ est appelé ondelette mère dont la définition générale est donnée par $\psi_{a,b} = \frac{1}{\sqrt{a}} \Psi(\frac{x-b}{a})$ où a est le facteur de dilatation et b le facteur de translation. Le choix de l'ondelette mère s'est porté sur l'ondelette de Morlet qui est très largement utilisée à travers la littérature et définie par :

$$w(t, f_0) = A e^{-t^2/2\sigma_t^2} e^{2i\pi f_0 t} \quad (4.3)$$

Où $\sigma_f = 1/2\pi\sigma_t$ et $A = (\sigma_t\sqrt{\pi})^{-1/2}$. L'ondelette de Morlet est caractérisée par le ratio constant $r = f_0/\sigma_f$ que nous avons fixé égale à 7 comme suggéré par [Tallon-Baudry et al. \(1997\)](#).

Cette décomposition peut être comparée à la transformée courte de Fourier qui décompose le signal en une somme de combinaisons linéaire de sinus et de cosinus mais part du principe qu'il existe une régularité dans le signal permettant une telle décomposition. La transformée en ondelettes résout plusieurs limitations :

- Elle permet d'obtenir l'énergie d'un signal dans le temps, ce qui permet une bien meilleure exploration des phénomènes.
- Le rapport constant r permet d'obtenir des ondelettes dont la résolution fréquentielle varie en fonction des fréquences et permet une meilleure coïncidence avec la définition des bandes physiologiques ([Bertrand et al., 1994](#))

Tout les attributs qui vont être maintenant présentés, utilisent les méthodes décrites ci-dessus.

4.1.1.4 Évaluation statistique à base de permutations

Pour une distribution de permutations construite à partir de deux sous-ensembles A et B et comportant N observations et pour une valeur p prédéfinie, on pourra conclure que :

- $A > B$ si A est parmi les $N - N \times p$ derniers échantillons ("One-tailed test upper tail")

- $A < B$ si A est parmi les $N \times p$ premiers échantillons ("One-tailed test lower tail")
- $A \neg B$ si A est soit inférieur aux $(N \times p)/2$ premiers échantillons soit supérieur aux $(N - N \times p)/2$

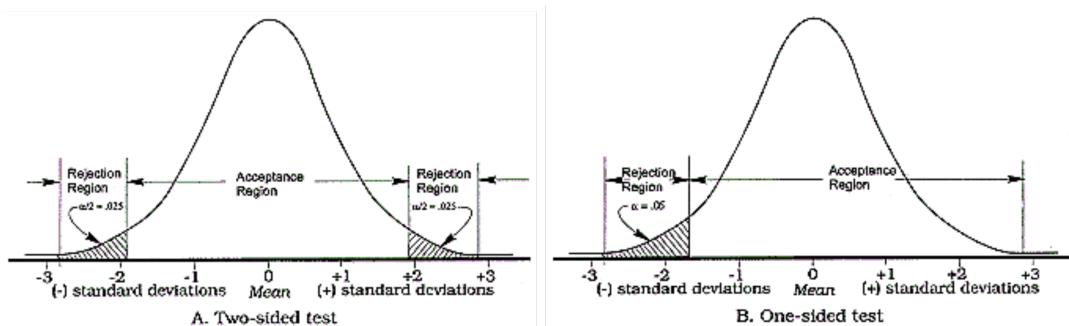


FIGURE 4.1 – Évaluation statistique à base de permutations, (A) "Two-tailed", (B) "One-tailed"

Grâce à cette méthode d'évaluation statistique, nous pourrons par exemple conclure si l'on a une augmentation, une diminution ou une différence statistique entre une valeur de puissance et la puissance contenue dans une période de baseline. Dernière précision, on comprend ainsi que pour obtenir une valeur p il faut que la taille de la distribution N soit au moins de $1/p$.

4.1.1.5 Hyperplan

Un hyperplan est un espace de co-dimension 1. Donc, dans un espace $3D$, l'hyperplan est un plan (dimension $2D + 1$). De manière générale, un espace de dimension N possède un hyperplan de dimension $N - 1$

$$\dim_{\text{ESPACE}} = \dim_{\text{HYPERPLAN}} + 1 \quad (4.4)$$

4.1.2 Puissance spectrale

4.1.2.1 Méthodes explorées

Le calcul de la puissance spectrale a été approché par deux méthodologies et qui ont été utilisés à des fins différentes :

- La transformée d'Hilbert : souvent exploité dans le cadre du décodage ainsi que pour garder une uniformité entre les attributs de phase et couplage phase-amplitude basés eux aussi sur cette transformée.
- La transformée en ondelettes : principalement utilisée pour la visualisation des cartes temps-fréquence à cause de l'adaptation des ondelettes aux bandes physiologiques.

4.1.2.2 Normalisation

On utilise la normalisation pour observer l'émergence d'un phénomène par rapport à une période définie comme baseline. A travers la littérature, quatre grands types de normalisation sont rencontrés :

1. Soustraction par la moyenne de la baseline
2. Division par la moyenne de la baseline
3. Soustraction puis division par la moyenne de la baseline

4. Z-score : soustraction de la moyenne puis division par la déviation de la baseline

La normalisation z-score est certainement la plus fréquemment rencontrée à travers la littérature. Le choix du type de normalisation dépend du type de données utilisées. Dans le cadre de nos données, β . était clairement la plus adaptée pour la visualisation. En revanche, dans le cadre de la classification, nous obtenions systématiquement de meilleurs résultats sans normalisation.

4.1.2.3 Évaluation statistique

La fiabilité statistique de la puissance a été évaluée en comparant chaque valeur de puissance à la puissance contenue dans une période définie comme baseline. Pour ce faire, nous avons testé deux approches :

1. Permutations : les valeurs de puissance et de baseline sont aléatoirement mélangées à travers les essais. Puis, on normalise cette puissance. En répétant cet procédure N fois, on obtient une distribution qui peut ensuite être utilisée pour en déduire la valeur p de la véritable puissance (cf : *pré-requis*)
2. "Wilcoxon signed-rank test" : ordonne les distances entre les paires de puissances (vraie valeur, baseline) ([Demandt et al., 2012](#), [Rickert, 2005](#), [Waldert et al., 2008](#))

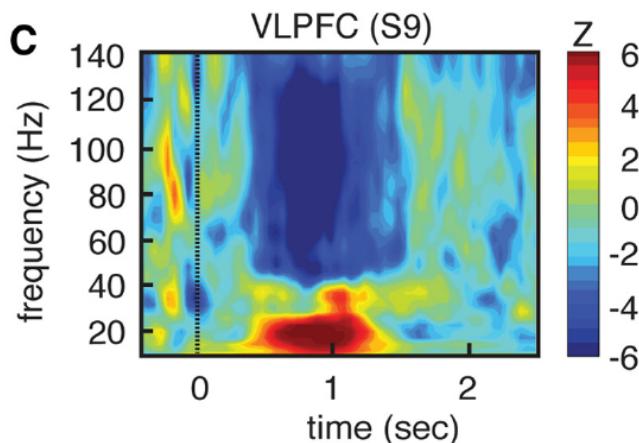


FIGURE 4.2 – Exemple de représentation temps-fréquence de puissance normalisées z-score ([Ossandon et al., 2011](#))

4.1.3 Phase

L'extraction de la phase se fait de la même manière que pour le Couplage phase-amplitude , en prenant l'angle de la transformée d'Hilbert d'un signal filtré. La significativité peut être évaluée en utilisant le test de Rayleigh ([Jervis et al., 1983](#), [Tallon-Baudry et al., 1997](#)). Point de vue pratique, cela correspond à la fonction `circ_rtest` de la toolbox Matlab *CircStat* ([Berens and others, 2009](#))

4.1.4 Phase-amplitude coupling

Le calcul du Phase-amplitude coupling ne se limite pas uniquement à la méthode. En réalité, pour obtenir une estimation fiable sur des données réelles, il est indispensable de suivre les trois étapes suivantes :

1. Estimation de la véritable valeur de PAC. Il existe plusieurs méthodes.
2. Calcul de "surrogates" : on va calculer des PAC déstructurés. Idem, il existe de nombreuses méthodes
3. Correction du véritable PAC par les "surrogates". Cette correction, qui est en fait une normalisation, aura pour but de soustraire à l'estimation du PAC de l'information considérée comme bruitée.

Les sous-parties suivantes présenteront de manières succinctes les principales méthodes rencontrées dans la littérature, ainsi que différents types de corrections applicables.

4.1.4.1 Méthodologie du phase-amplitude coupling

Il existe une large variété de méthodes pour calculer le PAC, ce qui complique son exploration. Toutefois, il n'existe pas de consensus sur une méthode plus polyvalente qu'une autre, chacune possédant ses points forts et limitations. Pour aller un peu plus loin, et présenter quelques méthodes, il est nécessaire d'introduire quelques variables. Soit $x(t)$, une série temporelle de données de taille N. Pour cette série temporelle, on souhaite savoir si la phase extraite dans une bande de fréquence $f_\phi = [f_{\phi_1}, f_{\phi_2}]$ est couplée avec l'amplitude contenue dans $f_A = [f_{A_1}, f_{A_2}]$. Pour cela, on va tout d'abord extraire $x_\phi(t)$ et $x_A(t)$ les signaux filtrés dans ces deux bandes. Enfin, la phase $\phi(t)$ est obtenue en prenant l'angle de la transformée d'Hilbert de $x_\phi(t)$ tandis que l'amplitude $a(t)$ est obtenue en prenant le module de la transformée d'Hilbert de $x_A(t)$.

1. Mean Vector Length-Modulation Index :

Cette méthode a été introduite par [Canolty et al. \(2006\)](#) et consiste à sommer, à travers le temps, le complexe formé de l'amplitude des hautes fréquences avec la phase des basses fréquences. L'équation est donnée par :

$$MVL = \left| \sum_{j=1}^N a(j) \times e^{j\phi(j)} \right| \quad (4.5)$$

2. Kullback-Leibler divergence :

A l'origine, la divergence de Kullback-Leibler (KLD), qui est issue de la théorie de l'information, permet de mesurer les dissimilarités entre deux distributions de probabilités. Ainsi, pour pouvoir utiliser cette mesure dans le cadre du PAC, [Tort et al. \(2010\)](#) propose une solution élégante qui consiste à générer une distribution de densité probabilités de l'amplitude (DPA) en fonction des valeurs de phase et d'ensuite utiliser le KLD pour comparer cette distribution à la densité de probabilité d'une distribution uniforme (DPU). Plus la DPA s'éloigne de la DPU, plus le couplage entre l'amplitude et la phase est consistant.

Pour construire la DPA, l'astuce consiste à couper le cercle trigonométrique en N tranches (dans l'article il est proposé de couper en 18 tranches de 20°). Puis, si on prend l'exemple de la tranche $[0, 20^\circ]$, on va chercher tout les instants temporels où la phase prend des valeurs comprises entre $[0, 20^\circ]$ ($t, \phi(t) \in [0, 20^\circ]$). On prend ensuite la moyenne de l'amplitude pour ces valeurs de t et on répète cette procédure pour chacune des tranches de phase.

On obtient ainsi la densité d'amplitudes en fonction des valeurs de phase. Il ne reste plus qu'à normaliser cette distribution par la somme des amplitudes à travers les tranches et on récupère une distribution de densité de probabilités. La figure 4.3 ([Tort et al., 2010](#)) présente un exemple de DPA en fonction de tranches de phase.

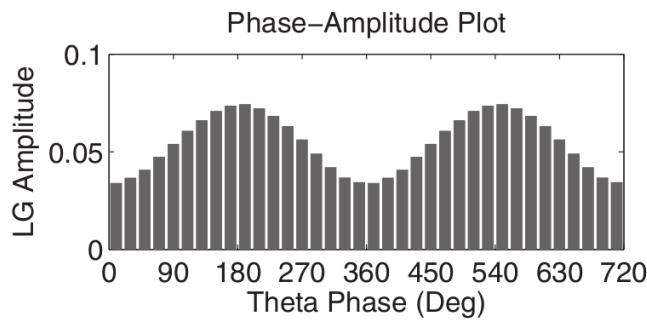


FIGURE 4.3 – *Densité de probabilité d'une distribution d'amplitudes en fonction de tranches de phases*

Le calcul de la divergence de Kullback-Leibler est ensuite appliqué pour mesurer les dissimilarités entre la DPA et la DPU et c'est cette mesure qui servira d'estimation du couplage phase-amplitude :

$$D_{KL}(P, Q) = \sum_{j=1}^N P(j) \times \log \frac{P(j)}{Q(j)} \quad (4.6)$$

où $P(j)$ est la densité de probabilité de $a(t)$ en fonction de $\phi(t)$ et $Q(j)$ est la densité de probabilité d'une distribution uniforme.

3. Height Ratio

La méthode du Height Ratio ([Lakatos, 2005](#)) est extrêmement proche du Kullback-Leibler divergence . En effet, l'amplitude sera binée de la même façon en fonction des tranches de phase. La mesure du PAC est ensuite donnée par :

$$hr = (f_{max} - f_{min}) / f_{max} \quad (4.7)$$

où f_{max} et f_{min} sont respectivement le maximum et le minimum de la densité de probabilité de l'amplitude en fonction des valeurs de phase.

4. Normalized Direct Phase-Amplitude Coupling

Le Normalized Direct Phase-Amplitude Coupling , qui n'est pas une des méthodes les plus fréquemment rencontrées, présente toutefois une avantage certain. En plus de fournir une estimation fiable du couplage phase-amplitude , [Ozkurt \(2012\)](#) démontre l'existence d'un seuil à partir duquel on peut considérer l'estimation du PAC comme étant statistiquement fiable. La beauté de cette méthode, c'est que ce seuil statistique, qui est une fonction de la valeur p désirée, ne dépend que de la taille de la série temporelle. Ce qui rend son utilisation particulièrement simple.

Pour estimer le PAC, une des hypothèses ayant permis d'aboutir à ce seuil

statistique est de devoir normaliser l'amplitude par un z-score dénotée $\tilde{a}(t)$. L'estimation du PAC est quasiment identique au MVL puisque c'est en réalité le carré de celle-ci. Enfin, pour une valeur p désirée, l'article introduit le seuil statistique :

$$x_{lim} = N \times [erf^{-1}(1 - p)]^2 \quad (4.8)$$

où erf^{-1} est la fonction d'erreur inverse. On déduira que l'estimation PAC est significative si et seulement si cette valeur est deux fois supérieure à ce seuil.

5. Autres méthodes : Tout les algorithmes présentés ci-dessus ont été testés, implémentés et comparés. En complément, voici une liste non exhaustive d'autres méthodes existantes :

- *Phase Locking Value (PLV)* ([Cohen, 2008, Penny et al., 2008](#)) : détournement du PLV proposé par [Lachaux et al. \(1999\)](#) qui mesure la synchronie de phase entre deux électrodes. Cette méthode va comparer la phase des basses fréquences avec la phase de l'amplitude des hautes-fréquences.
- *Generalized Linear Model (GLM)* ([Penny et al., 2008](#)) : outil décrit comme adapté aux données courtes et bruitées.
- *Generalized Morse Wavelets (GMW)* ([Nakhnikian et al., 2016](#)) : basée sur des ondelettes, semble particulièrement utile dans le cadre de l'exploration des données.
- *Oscillatory Triggered Coupling (OTC)* ([Dvorak and Fenton, 2014, Watrous et al., 2015](#)) : issue d'une détection de maximums des hautes fréquences.

4.1.4.2 Correction du phase-amplitude coupling et évaluation statistique

Nous avons vu dans la section précédente différentes méthodes permettant de calculer un Couplage phase-amplitude . Toutefois, celui-ci peut être largement amélioré en faisant une estimation du PAC contenu dans le bruit des données. Une fois que cette estimation sera faite, on pourra retrancher ce PAC bruité à la valeur initiale. Tout comme il existe plusieurs méthodes de PAC, les équipes de recherche proposent à tour de rôle de nouvelles méthodes. Parmi elles, on peut citer :

- *Time-lag* : proposée par [Canolty et al. \(2006\)](#), on introduit un délai sur l'amplitude compris entre $[f_s, N - f_s]$ où f_s est la fréquence d'échantillonnage et N est le nombre de points de la série temporelle
- *Shuffling des couples [phase,amplitude]* : ici, on mélange aléatoirement les essais de phase et d'amplitude ([Tort et al., 2010](#))
- *Swapping temporel d'amplitudes (ou de phase)* : on mélange aléatoirement les essais d'amplitude puis on recalcule le PAC avec la phase originale ([Bahramisharif et al., 2013, Lachaux et al., 1999, Penny et al., 2008, Yanagisawa et al., 2012b](#))

Ces trois méthodes produisent une distribution de *surrogates*. On pourra ensuite appliquer un z-score à la véritable estimation en utilisant la moyenne et la déviation de cette distribution. Enfin, l'évaluation statistique se fait également à partir de cette distribution (cf : *pré-requis*)

A ma connaissance, il n'existe pas de comparatif entre ces corrections et je n'ai jamais rencontré d'articles mentionnant que l'on ne puisse pas combiner les méthodes

de PAC avec les différentes corrections. En revanche, ce qui est relaté c'est que le *time-lag* nécessite des données longues dû à l'introduction de ce délai temporel.

4.1.4.3 Comparatif des méthodes

[Penny et al. \(2008\)](#) ont comparé plusieurs méthodes dont le *MVL*, *PLV* et le *GLM* et [Tort et al. \(2010\)](#) ont complété cette étude avec d'autres méthodes (cf. A.3). Enfin, [Canolty and Knight \(2010\)](#) a fait une review qui comprend un descriptif très instructif.

4.1.4.4 Représentation du phase-amplitude coupling

Comparée à la puissance, l'exploration du PAC peut s'avérer plus complexe dû à sa dimensionnalité plus grande. Il existe donc des outils et des méthodes destinées à simplifier cette exploration et à visualiser ces résultats.

Exemple concret, si on cherche à connaître les modulations de puissance contenue dans un signal, on peut représenter une carte temps-fréquence . Pour le PAC, idéalement on voudrait visualiser les phases, les amplitudes et le temps mais ces trois dimensions empêche une représentation simple. On peut donc avoir recours à différents types de représentations complémentaires :

- Puissance phase-locked : cette représentation permet de faire émerger l'existence d'un couplage, pour une phase donnée, et d'observer sa durée. Pour cela, on aligne les phases en détectant le pic le plus proche de l'instant temporel étudié. On calcule les cartes temps-fréquence que l'on va ensuite moyenner après les avoir recalées de la même façon que les phases (c'est-à-dire avec la même latence).
- Comodulogramme : pour une tranche temporelle définie, on représente les valeurs de PAC pour différentes valeurs de phase et d'amplitude

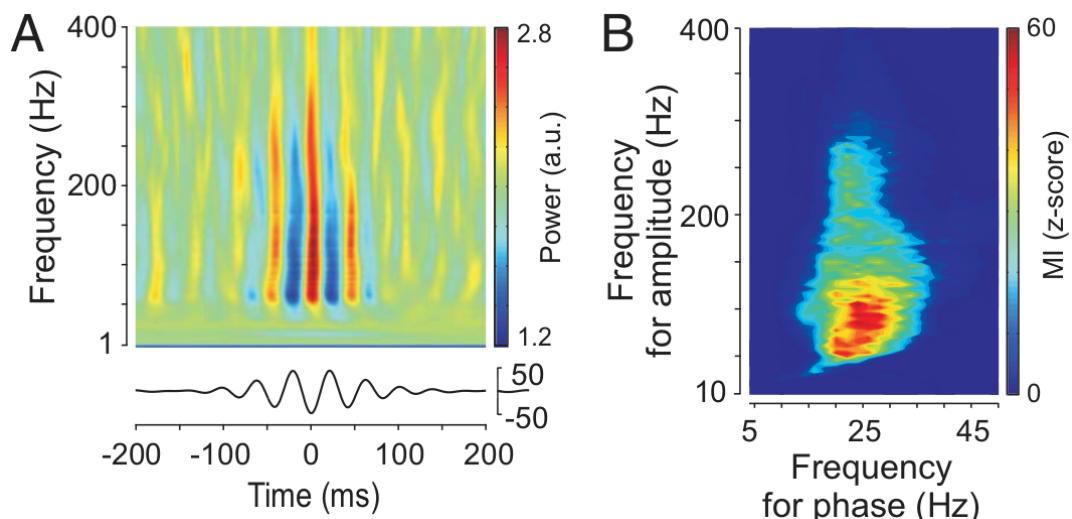


FIGURE 4.4 – (A) Exemple de cartes temps-fréquence phase locked sur le β , (B) Exemple de comodulogramme

La figure 4.4 (de [Hemptinne et al., 2013](#)) met en évidence que la représentation des cartes temps-fréquence phase-locked (A) est limitée d'une part, par la phase sur laquelle on choisit de recaler et d'autre part cette méthode est également limité par l'instant où l'on choisit de recaler. Pour la figure (B), le calcul du PAC se faisant à

travers la dimension temporelle, on a aucune idée de l'évolution du couplage dans le temps.

4.1.4.5 Phase-amplitude coupling : résolution temporel ?

Voytek et al. (2013) introduit une méthode appelée *Event Related Phase-Amplitude Coupling* permettant de mesurer l'évolution temporelle du PAC. L'approche traditionnelle nécessite de connaître un nombre de cycles afin d'en déduire l'existence ou non du couplage, et donc perdre la dimension temps. L'article propose de calculer le PAC à travers les essais (ou répétitions). Pour un jeu de données de M essais de longueur N, on extrait respectivement les phases et les amplitudes $\phi_M(t)$ et $a_M(t)$ puis, pour chaque point temporel, on calcul la corrélation à travers les essais (corrélation linéaire-circulaire (Berens and others, 2009) qui se fait entre l'amplitude et des sinus/cosinus de la phase). Il en résulte une valeur de corrélation pour chaque instant et donc, de couplage.

Il est intéressant de noter que ce changement de positionnement dimensionnel est le même dans le cadre du calcul du *Phase-Locking Statistics* (PLS). Lachaux et al. (1999) propose de mesurer la synchronie de phase entre deux sites distants en calculant le PLS à travers les essais (dans ce cas, on conserve la dimension temporelle) ou alors à l'intérieur de chaque essais (Lachaux et al. (2000) - on perd la dimension temporelle mais on conserve la dimension essais ce qui permet d'envisager une classification avec le *single-trial PLS*)

4.1.4.6 Phase préférentielle

La phase préférentielle correspondant à la phase où l'amplitude est maximum. De manière analogue au Kullback-Leibler divergence , l'amplitude est d'abord découpée en fonction de tranches de phases. La tranche de phase donnant une amplitude maximum est alors appelée *phase préférentielle* (PP). Cette mesure est particulièrement proche du Height Ratio , il serait intéressant d'étudier si l'un ou l'autre est porteuse d'une plus grande quantité d'informations.

Dans le cadre de cette thèse, nous avons entamé une exploration de l'utilité de la PP pour décoder des directions de mouvements, à la fois durant la préparation et l'exécution motrice. Nous avons obtenu des résultats faiblement significatifs et difficiles à remettre dans le contexte de la littérature. Pour cette raison, et par manque d'investigations supplémentaires, les résultats n'ont pas été inclus dans les papiers.

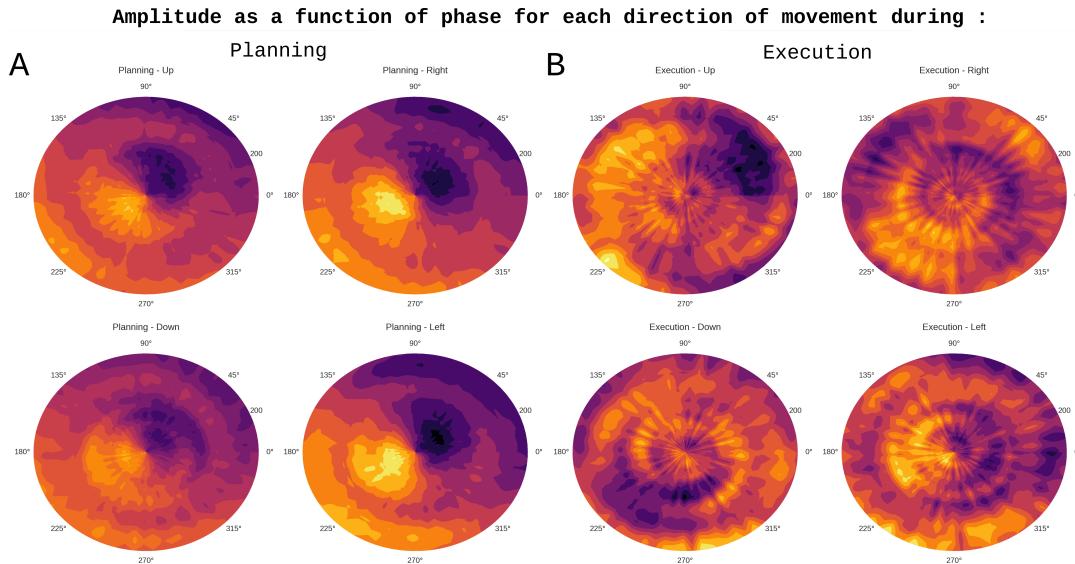


FIGURE 4.5 – Exemple de phase préférentielle - Chaque disque représente les modulations d'amplitude pour différentes valeurs de phase durant (A) une phase de préparation motrice et (B) d'exécution motrice pour quatre directions de mouvement (haut / bas / gauche / droite)

4.2 APPRENTISSAGE SUPERVISÉ

Le travail effectué durant cette thèse s'est exclusivement porté sur l'apprentissage supervisé. Celui-ci consiste à apprendre à la machine à reconnaître des événements qui ont été labellisé au préalable (cf. 4.2.1). A contrario, l'apprentissage non supervisé laisse la machine apprendre par elle-même. En pratique, l'apprentissage se fait sur des attributs. Par exemple, pour différencier des chats et des chiens, ou pourra utiliser l'angle formé par le sommet des oreilles. Les attributs doivent contenir une information pertinente permettant de différencier les classes. Enfin, les algorithmes de classification vont se servir de ces attributs pour définir une frontière entre les classes étudiées. A ce stade, il semble important de préciser que l'utilisation des outils d'apprentissage machine peut s'orienter (globalement) suivant deux axes :

1. Optimisation des attributs : on travail sur un raffinement des attributs afin que ceux-ci soient les plus performants possibles pour séparer les classes
2. Optimisation des paramètres de classification : on considère une base de données comme étant fixe, définitive, optimale et l'on va faire varier les différents paramètres liés à l'apprentissage machine (classificateurs, cross-validation...). C'est le cas des compétitions *BCI* où tout le monde travail sur une même base de données.

Bien sûr, ces deux axes peuvent être cumulés. Dans le cadre de cette thèse, le machine learning a été utilisé comme outil de validation d'hypothèses donc essentiellement porté sur l'optimisation des attributs. Le raffinement des paramètres de classification a également été étudié, mais, au final, il ne constitue pas la majeure partie de l'étude.

Un schéma classique d'analyse peut-être décrit par :

1. Labellisation des données
2. Constitution de données d'entraînement (*training*) et de test (*testing*)

3. Choix d'un classifieur puis entraînement de celui-ci sur les données *training*
 4. Test de ce classifieur entraîné sur les données *testing* et évaluation de la performance
 5. Évaluation statistique de cette acuité de décodage

4.2.1 Labellisation et apprentissage

La labellisation c'est le fait d'associer à chaque événement l'appartenance à une classe ou à une condition. C'est par ce procédé que l'on va pouvoir apprendre ensuite au classifieur à identifier les classes. Par exemple, considérons *up* et *down* deux classes qui reflètent des mouvements de la main vers le haut ou vers le bas. On va donc construire un vecteur $y_{direction}$ qui labellise chaque essais avec direction effectuée (ce vecteur peut aussi être booléen ou contenir des entiers. L'essentiel est que à chaque classe soit attribué une valeur qui lui est propre). Ce vecteur y est appelé *vecteur label*, qui vient labelliser chaque essais d'un vecteur d'attributs x .

$$y_{direction} = \begin{pmatrix} up \\ down \\ down \\ \vdots \\ up \end{pmatrix}, y_{bool} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, x = \begin{pmatrix} x_{trial_1} \\ x_{trial_2} \\ x_{trial_3} \\ \vdots \\ x_{trial_N} \end{pmatrix} \quad (4.9)$$

D'où le nom apprentissage supervisé. Finalement, l'apprentissage machine se fera grâce à ce vecteur label y et cette matrice d'attributs x . Ce qui nous amène directement aux notions de *training set* et de *testing set*.

x												
y	0	0	0	1	1	1	2	2	2	3	3	3

FIGURE 4.6 – Labellisation de données - pour un vecteur de données x , chacun des essais se voit attribuer une classe $c_i \in \{0; 3\}$

4.2.2 *Training, testing et validation-croisée*

Cette section est sans aucun doute la plus importante pour le machine learning puisque c'est elle qui assure la conformité méthodologique.

Un bon exemple pour comprendre cette partie est celui des contrôles de mathématiques. Avant l'examen, l'étudiant s'entraîne sur une série d'exercices. C'est la phase de *training*. D'ailleurs, plus il s'entraîne, plus ses chances de réussir à l'examen sont grandes. Le jour du contrôle, le professeur teste l'étudiant sur une série de nouveaux exercices en lien avec ce qu'il a étudié. C'est le *testing*. Ici, c'est un test parfait puisque l'étudiant est naïf sur le contenu de l'examen ce qui veut dire que l'on teste ses capacités mathématiques pures. Toutefois, il peut arriver durant la scolarité que l'on soit testé sur des exercices que l'on a déjà vu dans la phase de *training*. Dans ce cas, la moyenne des notes des étudiants est généralement beaucoup plus élevée puisque l'on ne teste plus des capacités mathématiques, mais la capacité à restituer un apprentissage.

4.2.2.1 *Training set, testing set et naïveté*

Pour en revenir à la question du machine learning, on définit une partie des données pour entraîner la machine. Ensuite, on teste cette machine entraînée sur un

nouveau jeu de données de test. Il est essentiel d'avoir une séparation stricte entre des données définies comme *training* et des données de *testing* afin d'assurer la naïveté du classifieur. Même si cela peut paraître évident, nous verrons que ça n'est pas toujours aussi facile que ça.

Se pose maintenant la question de comment l'on choisit de couper les données en *training* et *testing*. Une méthode serait de prendre une partie des données de manière aléatoire, de la définir comme *training* et sur tester sur les données restantes. Toutefois, ce choix ne représenterait qu'une partie des données. Une méthode plus exhaustive et plus rigoureuse consiste à utiliser une validation-croisée (ou cross-validation).

4.2.2.2 Validation-croisée

La validation-croisée (CV) est une procédure permettant de séparer les données en *training* et *testing*. Pour comprendre comment cela fonctionne, prenons un ensemble composé de N échantillons. Il existe plusieurs types de CV mais de manière générale, toutes dérivent du même principe qui est la cross-validation k-Fold ([Efron and Tibshirani, 1994](#), [Kohavi and others, 1995](#)). On coupe les N échantillons en k paquets de tailles égales (ou proches). Ensuite, le classifieur est entraîné sur $k - 1$ paquets puis on le teste sur le paquet restant. Cette procédure est ensuite appliquée k fois afin que chaque paquet passe au *testing*. On dira que la cross-validation est *stratified* si la proportion de classes représentées au sein de chaque dossier est approximativement uniforme à travers les folds. On pourra aussi rencontrer le terme *shuffle* si il y a un mélange supplémentaire. Tout cela nous emmène à des CV k-fold, k-fold stratified, k-fold shuffle ou encore k-fold stratified shuffle.

Concernant le nombre de folds, on rencontre en général 3 valeurs à travers la littérature : 3-folds, 5-folds ou 10-folds ([Latinne et al., 2001](#), [Yanagisawa et al., 2009](#), [Besserve et al., 2007](#), [Waldert et al., 2008](#)). Un cas particulier, mais si le nombre de folds $k = N$, ça revient à entraîner la machine sur $N - 1$ échantillons tester sur celui qui a été isolé et on répète cette procédure N fois. C'est ce que l'on appelle le *Leave-One-Out*. Toutefois cette dernière possède une grande variance et peut conduire à des estimations non fiables ([Efron and Tibshirani, 1994](#), [Kohavi and others, 1995](#)). Un autre cas particulier, est celui du *Leave-p-Subject-Out* ([Vidaurre et al., 2009](#), [Lajnef et al., 2015](#)) qui consiste à entraîner sur p sujets et tester sur les sujets restants. Cette procédure est particulièrement exigeante puisqu'elle nécessite d'avoir une certaine reproductibilité entre les sujets. Cette validation-croisée est fréquente avec des données EEG mais impossible à mettre en œuvre pour la sEEG à cause de l'implantation unique de chaque sujet.

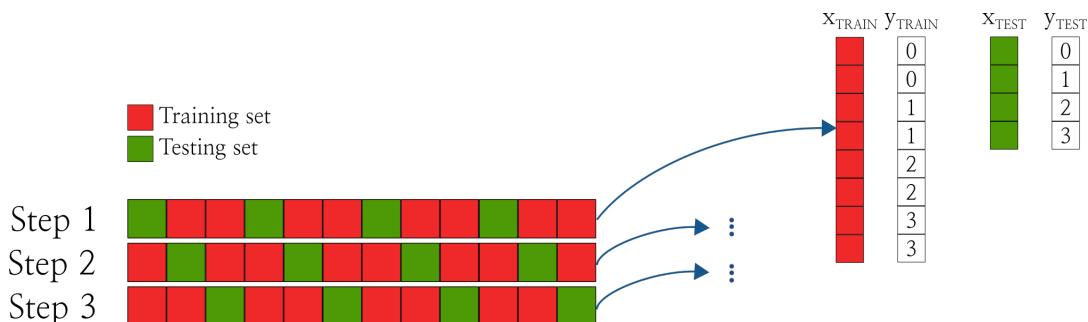


FIGURE 4.7 – Exemple d'une cross validation 3-folds

4.2.3 Classificateurs

1. Linear Discriminant Analysis (LDA)

Le LDA ([Fisher, 1936](#)) est un classifieur linéaire. Pour un problème à deux classes, le LDA tente de trouver un hyperplan qui va maximiser la distance entre les classes tout en minimisant la variance inter-classes. Ce classifieur fait l'hypothèse que les données sont normalement distribuées avec la même co-variance. Un problème multi-classes pouvant être transformée en multiple bi-classes, le LDA tente de trouver un hyperplan séparant la classe du reste (*One-vs-All*)

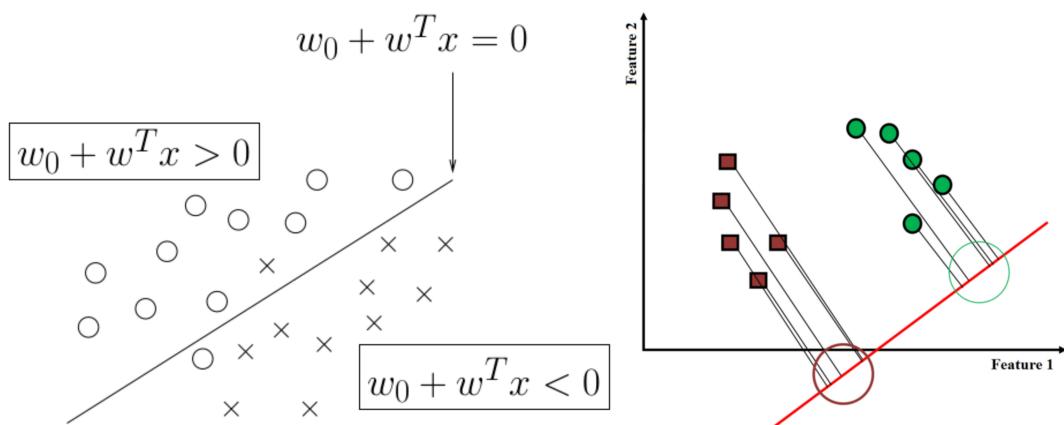


FIGURE 4.8 – Principe du Linear Discriminant Analysis ([Lotte et al., 2007](#), [Naseer and Hong, 2015](#))

2. Support Vector Machine (SVM)

Le SVM ([Boser et al., 1992](#), [Cortes and Vapnik, 1995](#), [Vladimir and Vapnik, 1995](#)) utilise également un hyperplan pour séparer deux classes. Toutefois, cet hyperplan optimal est trouvé en maximisant les marges (ou distance) entre ce plan et les attributs les plus proches. Le SVM possède une particularité, il utilise un noyau qui peut permettre de résoudre les problèmes linéaire (*linear SVM*) mais également les problèmes non-linéaire en projetant les données dans un espace de dimension supérieure (*kernel trick*). Un noyau que l'on retrouve assez régulièrement est le *Radial Basis Function (RBF)* ([Burges, 1998](#)). Les problèmes multi-classes peuvent également être traités en *One-vs-All*. Pour une utilisation optimale du SVM, il est vivement conseillé d'utiliser la librairie C *libsvm* ainsi que de lire les pré-requis sur son utilisation et optimisation ([Chang and Lin, 2011](#)).

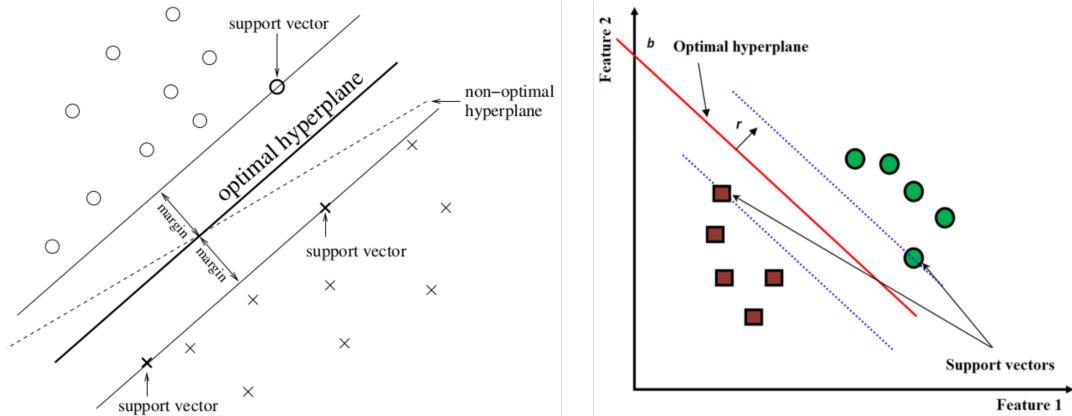


FIGURE 4.9 – Principe du Support Vector Machine ([Lotte et al., 2007](#), [Naseer and Hong, 2015](#))

3. k-Nearest Neighbor (KNN)

Pour un nouveau point de testing, le KNN ([Fix and Hodges Jr, 1951](#)) mesure la distance avec les k plus proches voisins et déduit la classe de ce point en fonction des classes de ces k -voisins (l'attribution de la classe se fait donc par vote)

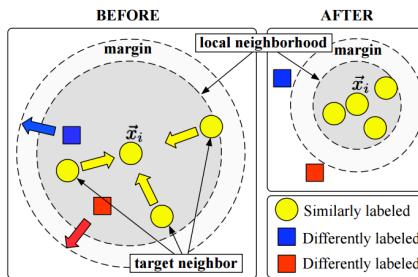


FIGURE 4.10 – Principe du k-Nearest Neighbor ([Weinberger et al., 2005](#))

4. Naive Bayes (NB)

Le NB ([Fukunaga, 1990](#)) est un classifieur probabiliste. Une des hypothèses du NB est que les données dans les classes doivent être normalement distribuées et indépendante.

La figure A.5 en annexe, issue de l'excellentissime librairie python scikit-learn dédiée au machine learning, illustre le comportement de chaque classifieur face à trois types de données. D'autres informations détaillées à propos des classificateurs peuvent être trouvées dans [Lotte et al. \(2007\)](#), [Wieland and Pittore \(2014\)](#), [Wu et al. \(2008\)](#)

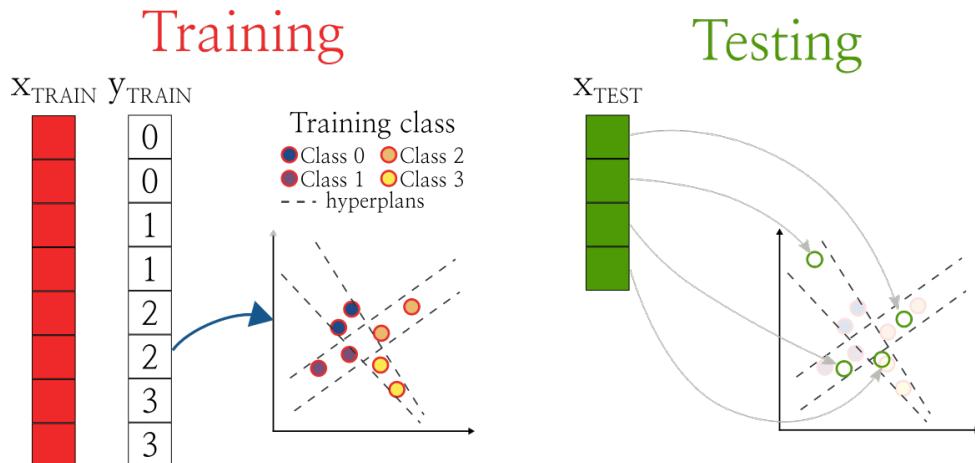


FIGURE 4.11 – Entraînement puis test d'un classifieur linéaire

4.2.4 Évaluation de la performance de décodage

La question qui se pose maintenant, c'est comment évaluer la performance de décodage. Pour cela, on peut par exemple utiliser le *Decoding accuracy* ou le *roc*

1. Decoding accuracy (DA)

L'utilisation (DA) est ce que l'on retrouve le plus fréquemment. Le calcul est simple, on compare les véritables labels avec les labels prédits par le classifieur. En faisant la somme des labels correctement prédits divisé par le nombre d'essais, on obtient un ratio qui correspond à l'acuité de décodage. Le plus souvent, ce ratio est ensuite exprimé en pourcentage. Le taux d'erreurs peut-être calculé en prenant $1 - DA$.

y_{TRUE}	$y_{PREDICTED}$
0	0
1	2
2	2
3	3
0	0
1	0
2	2
3	2
0	0
1	2
2	0
3	3

$8/12 \Rightarrow 75\%$

FIGURE 4.12 – Calcul de l'acuité de décodage

2. Receiver operating characteristic (ROC)

Une autre méthode pour évaluer la performance de décodage est l'utilisation de l'aire sous la courbe (AUC) ROC ([Ling et al., 2003](#), [Huang and Ling, 2005](#), [Bradley, 1997](#)). Celle-ci prend en compte le nombre d'essais correctement et incorrectement classifiés et pourrait donc prendre davantage de valeur possible comparé au Decoding accuracy .

4.2.5 Seuil de chance et évaluation statistique de la performance de décodage

De manière théorique, le seuil de chance est donné par $1/c$ où c est le nombre de classes. Par exemple, un problème à quatre classes donne un seuil de chance de 25%. Toutefois, ce seuil de chance est atteint pour un nombre de sample n infinis. En pratique, nous travaillons avec un nombre réduis de données, parfois même, avec très peu de sample. Dans ce cas, on peut obtenir des DA très élevés qui pourtant, ne sont pas pertinents. Les méthodes présentées ci-dessous ont pour but de trouver le seuil de chance associé à un jeu de donnée et de trouver pas la même occasion, la valeur p .

1. Loi binomiale

En faisant l'hypothèse que l'erreur de classification suit une distribution binomiale cumulative, on peut utiliser la loi suivante pour en déduire la probabilité de prédire au moins z fois la classe c :

$$P(z) = \sum_{i=z}^n \binom{n}{i} \times \left(\frac{1}{c}\right)^i \times \left(\frac{c-1}{c}\right)^{n-1} \quad (4.10)$$

2. Permutation

Les permutations présentent l'avantage d'être calculées à partir des données (*data driven*). [Ojala and Garriga \(2010\)](#) nous renseigne sur les différents types de permutations possibles dans le cadre du décodage :

- (a) *Full permutation* : les données sont mélangés
- (b) *Shuffle y* : le vecteur de label est mélangé. C'est la procédure la plus fréquemment rencontrée.
- (c) *Intra-class shuffle* : les données sont mélangées à travers la dimension *features* (colonne) et ce, à l'intérieur de chaque classe.

Autant les méthodes (a) et (b) nous renseigne véritablement sur la consistance d'un décodage par rapport aux données, autant la méthode (c) donne des informations un peu différentes. En effet, en cas de décodage non-significatif, on pourra soit conclure qu'il n'y a pas de consistance dans les attributs à l'intérieur des classes, soit que le classifieur est incapable d'utiliser cette l'interdépendance. [Ojala and Garriga \(2010\)](#) précise que dans ce cas, il n'est pas nécessaire d'utiliser un classifieur compliqué et qu'un classifieur simple devrait suffire.

Cette partie a fait l'objet d'une publication scientifique (cf. II - ([Combrisson and Jerbi, 2015](#))).

Un pipeline standard de classification est proposé en annexe (cf. A.4).

4.2.6 Du single au multi-features

Dans les sections précédentes, nous avons vu comment extraire des attributs de l'activité neuronale et comment les classifier. C'est ce que l'on appelle le *single feature* (SF), c'est-à-dire que l'on évalue la performance de chaque attribut séparément. Cette approche permet de constituer un set de features pertinents et répond à des questions neuro-scientifique. Cette démarche de SF a donc un but exploratoire.

La question que l'on peut maintenant se poser, c'est quelle performance de décodage puis-je obtenir si je combine ces attributs et dans quel cas est-ce utile? C'est le multi-attributs (ou *multi-features* (MF)). Tout d'abord, le MF est utilisé lorsqu'il y a

soit un désir soit un besoin de performances accrue. Par exemple, on utilisera le MF dans les compétitions de décodage ou tout simplement, pour une BCI où la performance est essentielle. Si l'on construit un système de bras robotisé piloté par activité neuronale, on comprend sans peine que celui-ci doit être le plus efficace possible et donc, le MF s'impose. Le dernier cas où l'on rencontre du MF, et ce n'est pas le cas le plus glorieux, c'est le cas où il y a un besoin de pallier à des résultats de SF assez faibles. La littérature expose des Decoding accuracy toujours plus hauts, des méthodes toujours plus complexes et donc, pour publier correctement un article, il faut avoir des résultats au-moins aussi perspicaces.

Le multi-features c'est donc l'utilisation de multiples attributs pour aboutir à une classification et ce, sans sélection particulière. Individuellement, les attributs d'un même set n'auront pas la même performance. Certains seront des bons marqueurs et d'autres, n'ajouteront pas ou peu d'information. Donc en combinant ces features, il est probable que l'acuité de décodage soit moins bonne que la performance en attribut unique. Pour cela, on pourra donc utiliser des algorithmes de sélections de marqueurs (*feature selection*). Le but de cette sélection est de trouver dans un set d'attributs, un sous-ensemble dont la performance groupée est meilleure que la performance individuelle.

Cette sélection est une procédure exigeante où le risque de sur-apprentissage est grand. C'est la raison pour laquelle cette sélection doit être mise à l'intérieur d'une cross-validation . Donc on définit un set de *training* et de *testing* grâce à la validation croisée, puis sur le *training* , on lance la *feature selection*. On aboutit à un sous-ensemble de marqueurs qui va servir à entraîner le classifieur. Ensuite, on sélectionne ce subset dans le *testing* et on test le classifieur avec ce subset. Toute ceci étant enfin répété pour chaque *fold* de la cross-validation . A la vue de cette procédure, deux problèmes émergent :

- La sélection d'attributs se faisant à l'intérieur des folds de la cross-validation , on peut très bien aboutir à des listes d'attributs différentes. Pour obtenir une information finale, on pourra donc parler des attributs les plus fréquemment choisis. Par exemple, si la sélection se fait dans un cross-validation 10-folds, on pourra dire que le feature 1 a été choisi 7/10, le feature 2, 3/10...
- En fonction de la sélection choisie et de la cross-validation , le pipeline complet peut être très (très) lourd et long.

Les mécanismes de *feature selection* peuvent être regroupés en deux grandes familles ([Guyon and Elisseeff, 2003](#), [Liu et al., 2008](#), [Das, 2001](#)) : les *Filter methods* et les *Wrapper methods*.

4.2.6.1 Filter methods

Ces méthodes sont basées sur un critère et sont indépendantes du classifieur. Parmi elles, on retrouve des outils de corrélation, d'information mutuelle ou encore de statistiques. Ces derniers outils évaluent la contribution de chaque feature de manière indépendante sans tenir compte de la corrélation entre ces features. Pour résoudre ce problème, [Yu and Liu \(2004\)](#), [Ding and Peng \(2005\)](#) introduisent le *minimal-redundancy-maximal-relevance* qui en plus de trouver les features les plus pertinents, va permettre d'éliminer ceux qui sont redondants.

Pour terminer, ces méthodes sont effectivement indépendantes de l'algorithme de classification mais elles peuvent s'avérer optimales pour tel ou tel classifieur (ex :

l'utilisation du critère de Fisher pour filtrer les features est très performant lorsqu'il est ensuite associé au Linear Discriminant Analysis ([Duda et al., 2001](#))).

4.2.6.2 Wrapper methods

Contrairement aux méthodes de filtrage, les *wrapper* utilisent le classifieur comme outil de sélection. Le premier inconvénient que l'on peut d'ors et déjà leur reprocher, c'est que le résultat final sera donc classifieur-dépendant, donc difficile pour la généralisation.

Parmi ces *Wrapper methods*, on peut citer :

1. Sélection exhaustive : on teste toutes les combinaisons de features possibles puis on sélectionne la meilleure. Procédure qui ne peut être faisable qu'en présence d'un jeu de données particulièrement restreint.
2. Sélection sur la statistique de décodage : on utilise le classifieur pour évaluer l'acuité de décodage de chaque feature séparément pour en déduire une valeur p (cf : [4.2.5](#)). Enfin, on sélectionne les features dont la valeur p est inférieur à un seuil désiré.
3. Sélection séquentielle : processus où l'on va ajouter/enlever des features de manière séquentielle jusqu'à atteindre un décodage optimal. Ce type de sélection se fait suivant deux directions :
 - (a) *Forward feature selection (FFS)* : la première étape consiste à évaluer la performance de chaque attributs. On sélectionne le meilleur que l'on va ensuite combiner en couple avec tout les features restant. On sélectionne le meilleur couple puis on teste les combinaisons des meilleures triplettes... On continu tant que la performance s'améliore. Si le DA d'une étape i est inférieur au DA de l'étape $i - 1$, on considère le nouveau subset de features à $i - 1$.

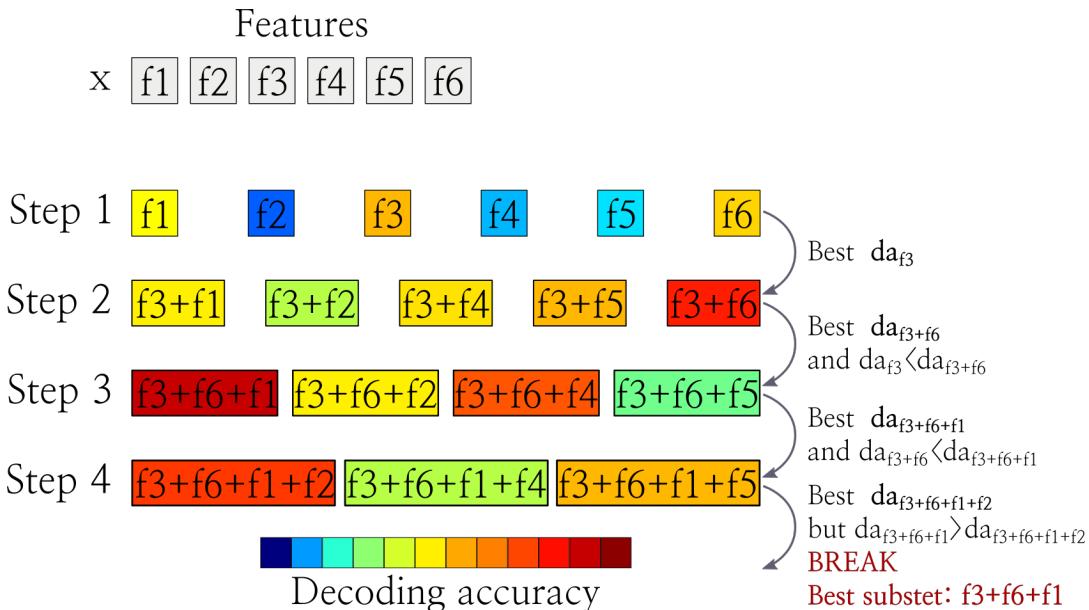


FIGURE 4.13 – Exemple d'une *Forward feature selection* appliquée sur six features

- (b) *Backward feature elimination (BFE)* : la philosophie est la même que pour un *forward*. On classifie d'abord les N features pris ensemble, puis on

enlève à tour de rôle chaque marqueur. On sélectionne le subset composé de $N - 1$ features ayant fourni le meilleur résultat, puis on enlève de nouveau chaque feature... L'algorithme s'arrête de la même façon que le *forward*.

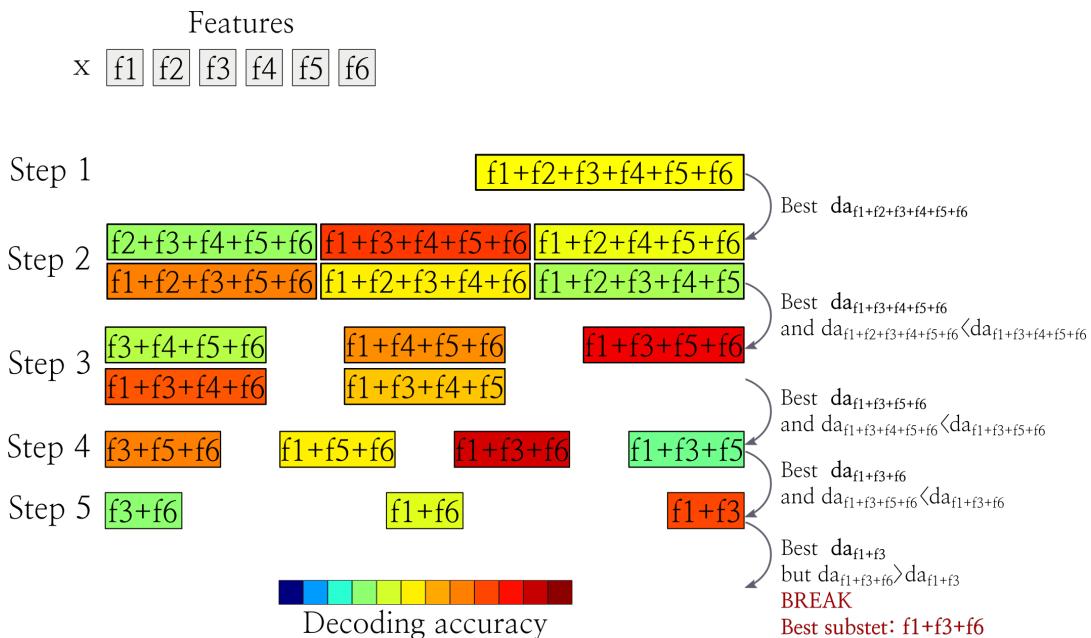


FIGURE 4.14 – Exemple d'une Backward feature elimination appliquée sur six features

De manière générale, il est rapporté que la FFS converge plus rapidement que la BFE (Guyon and Elisseeff, 2003). Toutefois, la FFS tombe plus facilement dans des minimums locaux et donc, mène à un décodage moins bon. En effet, la *forward* sélectionne pas-à-pas les meilleurs attributs, elle est donc moins ensembliste que la *backward*.

Les méthodes de filtrage demandent moins de ressources et représentent donc un premier choix pour les larges sets de données. En revanche, elles peuvent ne pas déceler les phénomènes de complémentarité entre features. Pour cette dernière raison, les méthodes de wrapper fournissent en général de meilleurs résultats ([Chai and Domeniconi, 2004](#)).

4.2.7 Généralisation temporelle

L'introduction du *single-feature* faite plus haut était une présentation générique, c'est-à-dire que celle-ci est vraie quelque soit les features étudiés. On pourra donc classifier des attributs de puissance, de PAC, de phase, d'entropie... On peut également envisager l'étude un seul marqueur mais dans sa dimension temporelle. En effet, cela consiste à entraîner et tester un classifieur à différents instants temporels pour voir si le décodage varie dans le temps. Une des limitations de cette utilisation d'un classifieur est que, à chaque instant, celui-ci change. Donc on ne peut inférer aucune généralisation. Pour envisager une généralisation, il faut entraîner le classifieur à un instant puis le tester à travers toute la dimension temporelle restante. Dans ce cas, on pourra parler de généralisation mais reste encore le problème du

choix de l'instant temporel qui servira à entraîner le prédateur.

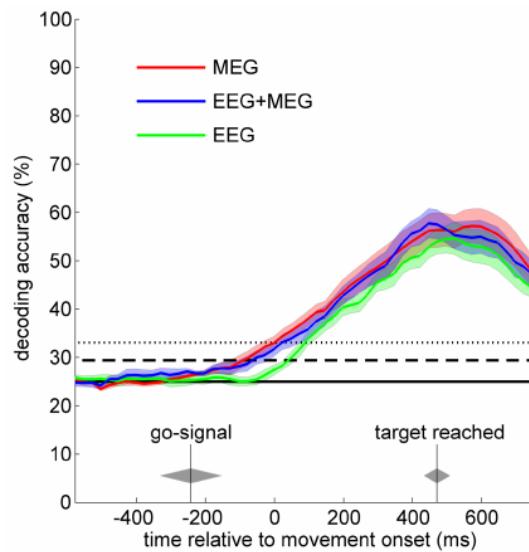


FIGURE 4.15 – Exemple de décodage temporel (Waldert et al., 2008). Ici, l'auteur décide 4-directions de mouvements de la main dans le temps. A chaque instant, un classifieur est créé, entraîné puis testé à ce même instant.

Pour répondre à cette question, King and Dehaene (2014) introduisent une idée particulièrement esthétique visant à généraliser le comportement d'un classifieur à travers le temps. Sans trop de surprise, ils ont nommé cette méthode la *généralisation temporelle*. Elle permet de répondre à deux limitations :

- Comment généraliser le comportement d'un classifieur lors d'une étude temporelle ?
- Comment choisir l'instant qui servira à entraîner le classifieur et quel impact ce choix aura-t-il sur le reste du décodage temporel ?

La méthodologie consiste à prendre un instant i , entraîner le classifieur et tester celui-ci sur tout les instants. Puis, on prend l'instant suivant $i + 1$, on entraîne un nouveau classifieur et on teste... Et on répète cette procédure pour tous les instants. On obtient ainsi une représentation 2D où, par convention, l'axe des ordonnées matérialise l'endroit où le classifieur a été entraîné (*Training time*) et l'axe des abscisses pour tester ce prédateur sur le reste de la dimension temporelle (*Generalization time*). La couleur permettra de signaler la performance de décodage.

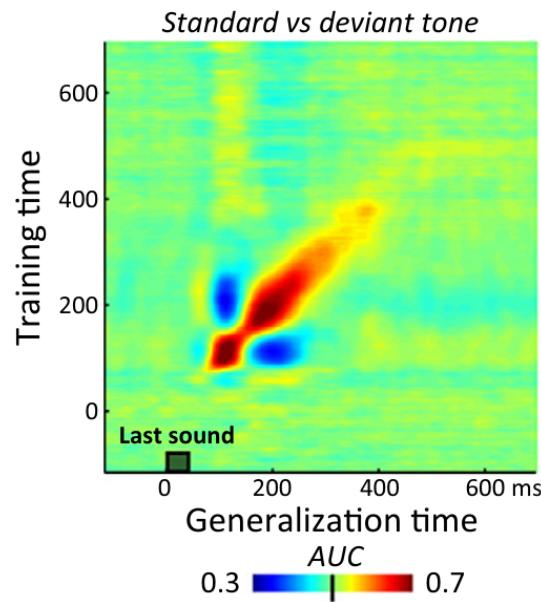


FIGURE 4.16 – Exemple de généralisation temporelle ([King and Dehaene, 2014](#))

DÉVELOPPEMENTS INFORMATIQUES

A l'heure actuelle, il existe une vaste diversité de *toolboxs*/logiciels permettant d'analyser puis de visualiser des données neuro-scientifiques, que ce soit en *Matlab*, en *Python* ou tout autre langage. Parmi ces solutions, on pourrait citer *Brainstorm*, *FieldTrip*, *MNE python*, *Nipipe*, *ELAN*... Toutes sont développées depuis des années, par des équipes hautement qualifiées et expertes et jouissent d'une excellente réputation. Durant cette thèse, nous avons souhaité proposer des solutions informatiques pour les raisons suivantes :

Maîtrise et compréhension des outils : bien qu'il soit tout à fait envisageable d'analyser ligne par ligne ces *toolboxs*, le code peut parfois être assez dense et difficile à comprendre. Coder soi-même ses outils est une merveilleuse méthode pour les démystifier et surtout, pour les utiliser correctement c'est-à-dire connaître les avantages et les limites de chacun.

Adaptation, amélioration et indépendance : lorsque l'on choisit une *toolbox* on est limité aux possibilités et à la qualité d'implémentation de celle-ci. Il se peut que des besoins très spécifiques ne soient donc pas couverts (pour des données intracrânienne comme dans le cadre de cette thèse, il existe peu de solution). Pour ces raisons, le développement d'outils personnels permet une meilleure couverture des besoins spécifiques et assure une indépendance face aux limites d'une boîte à outils.

Acquisition de compétences : l'inconvénient majeur de l'implémentation d'outils est le temps, un temps qui est forcément pris sur autre chose. En revanche, c'est une somme de compétences non-négligeables.

Identité, communication et communauté : si les outils développés sont de qualité et forment un ensemble cohérent, une communauté d'utilisateurs peut se mettre en place ce qui peut contribuer à faire connaître une équipe ou un laboratoire.

5.1 CHOIX DU LANGAGE : PYTHON

Dans leur première version, les solutions informatiques ont été développées en *Matlab*. *Python* s'est imposé plus tard notamment grâce à son confort d'écriture et sa qualité syntaxique, l'abondance de documentations, d'utilisateurs et de modules. De plus, c'est une langue portable, pouvant être installé sur toute machine et tout système et surtout, *Open Source* distribuable à souhait. A noter que le langage *Julia* (<https://julialang.org/>) a également été testé. Ce langage se veut particulièrement prometteur puisqu'il promet une syntaxe élégante à l'instar de *Python* et des performances se rapprochant du *C*, devançant ainsi *Python* dans sa version non-optimisée. Il a toutefois été écarté, non à cause de ses performances mais parce que

c'est un langage encore récent et comportant un nombre plus réduit de modules que *Python* et une communauté plus petite.

5.2 PAQUETS DÉVELOPPÉS DURANT CETTE THÈSE

Trois paquets ont été développés pour proposer une solution cohérente de l'extraction des données à la visualisation des résultats :

ipywksp : workspace s'intégrant dans les notebooks *Jupyter*.

brainpipe : paquet permettant d'analyser des données (pré-traitements, extraction de features, classification et visualisation 2D)

visbrain : ensemble de modules destinés à des visualisations complexes et de hautes performances.

5.2.1 ipywks

Ce paquet est destiné aux utilisateurs venant de *Matlab* et souhaitant retrouver un workspace semblable. **ipywksp** mêle plusieurs langages (*Python*, *HTML* et *JavaScript*) et permet de visualiser le type, le contenu et la taille des variables, de les sauvegarder/charger et de les visualiser. Enfin, ce paquet

5.2.1.1 Installation et utilisation

Dans un terminal, lancer :

```
git clone https://github.com/EtienneCmb/ipywksp.git
python setup.py install
```

Pour utiliser le workspace, lancer dans un notebook *Jupyter* :

```
# Chargement du module :
from ipywks import workspace

# Ouverture du workspace avec un thème noir et s'affichant automatiquement
# au survol de la souris :
workspace(theme="dark", autoHide=True)
```

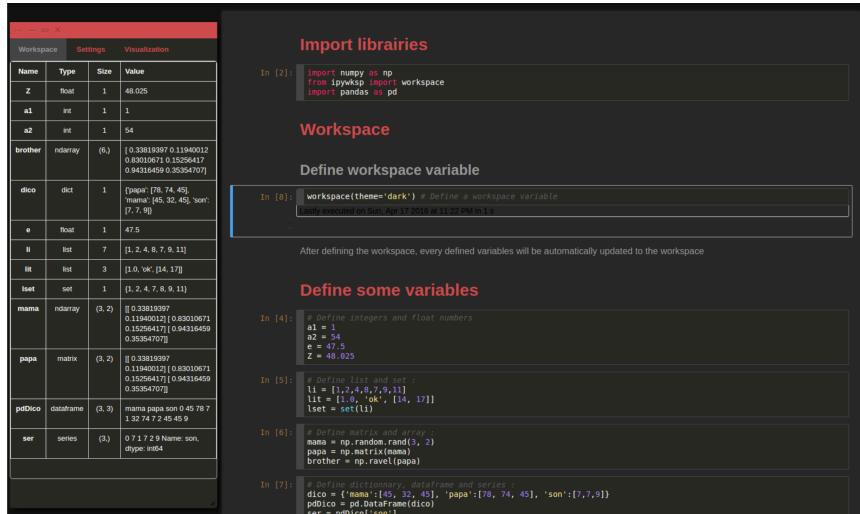


FIGURE 5.1 – *ipywksp* : Exemple de workspace pour *Jupyter*

5.2.2 Brainpipe

Ce paquet est destiné à l'analyse de données de tout type même si il est particulièrement adapté aux données intracrânienne. Il permet d'extraire un ensemble d'attributs, de les classifier, d'effectuer des analyses statistiques et de visualiser les résultats sur des graphes simples. Tout les résultats obtenus durant cette thèse ont été obtenu avec ce module et donc, toutes les méthodes y sont implémentées.

5.2.2.1 Fonctionnalités

5.2.2.1.1 Study : Ce sous-module permet de gérer plusieurs études et plusieurs jeux de données, de gérer un très grands nombres de fichiers, de créer une arborescence de dossiers propre, une meilleure gestion des chemins d'accès ce qui est un atout majeur pour des collaborations.

```
# Importation des librairies :
import numpy as np
from brainpipe.system import study

# Création de deux études :
st = study('MEG')
st.add('~/Python/')
st = study('EEG')
st.add('~/Python/')

# Création et sauvegarde d'une variable dans le sous-dossier database :
x = np.random.rand(1000)
st.save('database', 'test.npy', x)
```

5.2.2.1.2 Pré-traitements : Ensemble d'outils pour pré-traiter les données, c'est-à-dire des outils de filtrage performants, la bipolarisation et la recherche des structures anatomiques associées à des coordonnées MNI/Talairach.

```
# Chargement des librairies :
from brainpipe.preprocessing import bipolarization, xyz2phy
from brainpipe.system import study

# Chargement de l'étude en cours :
st = study('CenterOut')

# Chargement des données à pré-traiter :
data, channels, xyz = st.load('database', 'centerout_data.npz')
# Où :
# - data : les données intracrâniennes
# - channels : nom des channels monopolaires
# - xyz : les coordonnées MNI des channels

# Bipolarisation et recherche des structures anatomiques :
data_bip, channels_bip, xyz_bip = bipolarization(data, channels, xyz=xyz)
phy = xyz2phy().get(xyz_bip, channels_bip)
```

5.2.2.1.3 Attributs : Brainpipe intègre une collection relativement importante de features calculables :

- Signal filtré
- Amplitude
- Puissance (hilbert, wavelet ou PSD)

- Phase Amplitude Coupling (nombreuses méthodologies / possibilité de générer des signaux synthétiques couplés)
- Phase-Locking Factor (PLF)
- Cartes temps-frequencies
- Phase-Locked Power (puissance alignées sur un cue)
- Event-Related Phase Amplitude Coupling (ERPAC)
- Phase préférentielle
- Phase Locking Value (PLV, soit à travers le temps, soit à travers les trials)
- Entropie spectrale

A noter que certains attributs intègrent un fenêtrage et le calcul dans des bandes de fréquences. De plus, tous ont été implémentés de façon matricielle et peuvent être calculés en parallèle pour un temps de calcul le plus réduit possible. Enfin, tous comprennent de nombreuses configuration possibles, intègrent le calcul de significativité et les outils de visualisation.

```
# Chargement des librairies :
from brainpipe.feature import *
import numpy as np
import matplotlib.pyplot as plt

sf = 1024 # Fréquence d'échantillonage

# On génère des données contenant un couplage entre 10 et 100hz :
data = cfcRndSignals(sf=sf, fPha=10, fAmp=100, ndatasets=10, noise=2, chi
                      =.5) [0].T
npts = data.shape[0]

# On génère des vecteurs phase et amplitude :
pVec, aVec, pha, amp = cfcVec()

# Calcul du PAC :
pacO = pac(sf, npts, pha_f=pha, amp_f=amp, Id='133')
xPac = pacO.get(data, data, matricial=True)[0]
xPac = np.squeeze(xPac) # Suppression des dimensions inutiles

# Plot du PAC avec les fonctions intégrées :
fig = plt.figure()
pacO.plot2D(fig, xPac.mean(-1), vmin=0, vmax=16, xvec=pVec, yvec=aVec,
            xlabel='Fréquence de phase (hz)',
            ylabel='Fréquence amplitude (hz)', title='Exemple de PAC',
            cmap='viridis', cblabel='Couplage PAC')
plt.show()
```

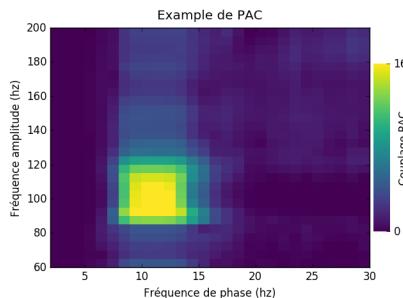


FIGURE 5.2 – Exemple de calcul PAC avec brainpipe

5.2.2.1.4 Classification : L'essentiel de la classification est assuré par *scikit-learn* (Pedregosa et al., 2011). Toutefois, brainpipe offre certaines fonctionnalités d'ordre pratiques qui, de manière non-exhaustive, peuvent être résumées à :

- Possibilité de définir une cross-validation et différents classifieurs et surtout, de pouvoir comparer leur performances de manière plus compactes.
- Adaptation de la classification aux données neuro-scientifiques, notamment en offrant un calcul en parallèle plus efficace car mieux adapté à nos petites données (en comparaison aux énormes banques de données d'images). De plus, de nombreuses études utilisent des classification particulière telles que le *Leave-p-Subject-Out*, présente dans brainpipe tout comme les cross-validation de type *10-times*....
- Généralisation temporelle (King and Dehaene, 2014)
- Calcul de la significativité des décodages plus synthétique (loi binomiale ou permutations)

Pour les neuro-scientifiques, brainpipe est un bon point d'entrée au monde de la classification puisqu'il permet de rapidement classifier nos données en un minimum de lignes et de manière lisible. Toutefois, pour une utilisation plus fine, un programme en *scikit-learn* pure reste moins limitatif.

5.2.2.1.5 Statistiques : En plus des statistiques calculés pour chaque attribut et pour la classification, brainpipe met à disposition un ensemble d'outils d'analyses statistiques. Calcul et gestion de permutations, correction multiple (Bonferroni, False Discovery Rate, Maximum statistic) ainsi que des outils pour les données circulaires (comme des données de phase).

5.2.2.1.6 Visualisation : Enfin, des fonctions pour visualiser des données ont également été ajoutées. Celles-ci permettent de créer des graphes 2D esthétiques et hautement configurables (plot de lignes avec déviation, ajout de valeur p , de lignes verticales/-horizontales, plot d'image, de contours...).

```
# Chargement des librairies :
from brainpipe.visual import BorderPlot
import numpy as np

# Définition d'un vecteur temps et de 10 sinusoïdales:
sf = 1024
t = np.mgrid[0:10, 0:1000] / sf
x = np.sin(2*np.pi*5*t) + .5 * np.random.rand(*t.shape)

# Plot du signal et de sa déviation :
BorderPlot(t[0, :], x.T, kind='std', xlabel='Temps', ylabel='Amplitude',
            title='Exemple de visualisation')
plt.show()
```

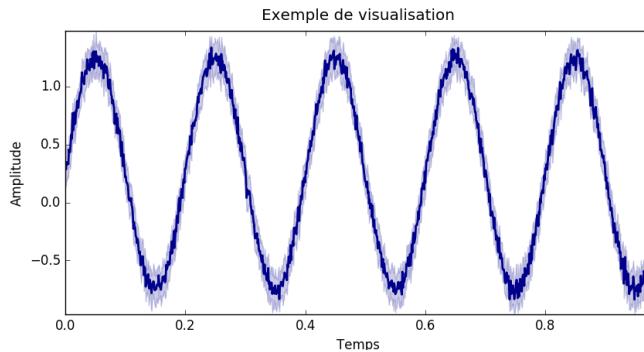


FIGURE 5.3 – Exemple de plot d'un signal et de sa déviation

5.2.2.2 Installation et documentation

Pour installer brainpipe, lancer dans un terminal :

```
git clone https://github.com/EtienneCmb/brainpipe.git
python setup.py install
```

Pour finir, une documentation complète est disponible en ligne <https://etiennevcmbo.github.io/brainpipe>

5.2.3 Visbrain

Visbrain est un paquet destiné à la visualisation de données neuro-scientifiques. Sa particularité réside dans le fait qu'il se base sur Vispy [Campagnola et al. \(2013\)](#) qui lui même utilise *OpenGL*. Les calculs sont envoyés sur la carte graphique ce qui, en conséquence, offre de très hautes performances en terme de fluidité et de temps de calcul. De plus, les interactions entre l'utilisateur et les différents modules se font via des interfaces graphiques (*Graphical User Interface, GUI*) construites à partir de *PyQt*.

5.2.3.1 Présentation des modules

5.2.3.1.1 Brain : *Brain* est une GUI avec un cerveau MNI dans lequel il est possible d'insérer des objets :

- **Sources** : dispositions de sources matérialisées par des sphères de couleur
- **Connectivité** : possibilité d'afficher des liens de connectivité entre ces sources
- **Structures** : ajout de structures 3D internes soit basées sur les aires de Brodmann soit sur l'AAL (*Automated Anatomical Labeling*)
- **Autres** : tout autre objet à trois dimensions peut-être rajouté par l'utilisateur.

Il n'y a aucune limite sur le nombre d'objets pouvant être ajoutés et ils peuvent tous être contrôlés indépendamment (couleur, transparence, taille, forme...). De plus, certains de ces objets peuvent interagir ensemble. Par exemple, l'activité des sources peuvent être projetées sur la surface du cerveau ou sur des structures internes.

Dernier point important, toutes les interactions possibles depuis l'interface graphique (et par raccourcis) sont également possibles en ligne de commande (Voir les [User functions](#) dans la documentation). Cette fonctionnalité est particulièrement utile pour produire un grand nombre de figures puisque tout peut être automatisé.

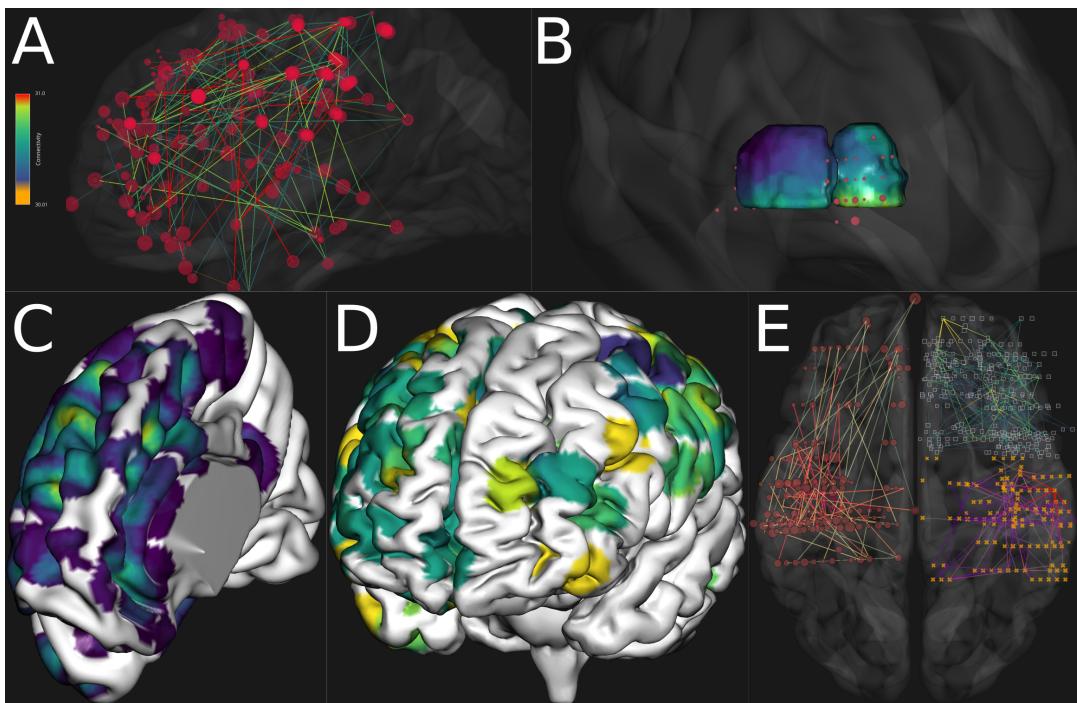


FIGURE 5.4 – Exemples des principales fonctionnalités de **Brain**, (A) Sites intracrâniens et connectivité, (B) Sources MEG et projection de leur puissance bêta sur le thalamus, (C) Nombre de sources contribuant à chaque point de l'hémisphère droit, (D) Projection de l'activité corticale de plusieurs sources intracrâniennes, (E) Exemple de scène complexe mêlant différents objets possédant chacun leur configuration

5.2.3.1.2 Sleep : *Sleep* est un module particulièrement performant pour visualiser, analyser et éditer des données de sommeil. Il a été développé en collaboration avec [Raphael Vallat](#).

Parmi les fonctionnalités principales, on peut citer :

- Chargement de fichiers *.edf, *.eeg (Brainvision, ELAN) et *.trc
- Visualisation temporelle des données polysomnographiques (avec possibilité d'afficher/cacher les channels, contrôle des unités temporelles, de la taille de fenêtre, de l'amplitude...), en spectrogramme ou sous forme topographique
- Chargement/visualisation/édition/sauvegarde de l'hypnogramme
- Implémentation de détection de spindles, K-complexes, slow waves, rapid eye movements (REM), contraction musculaire ou encore de pic. Chaque détection peut être lancée sur les channels souhaités et des repères visuels sont ajoutés à l'hypnogramme
- Outils de traitement de signal (suppression des composantes linéaire et de moyenne, bipolarisation/re-référencement, outil de filtrage pour afficher le signal filtré, l'amplitude, la puissance ou encore la phase)
- Nombreux raccourcis pour une interaction la plus efficace possible.
- Une fonction d'*Auto scoring* est actuellement en cours de développement.

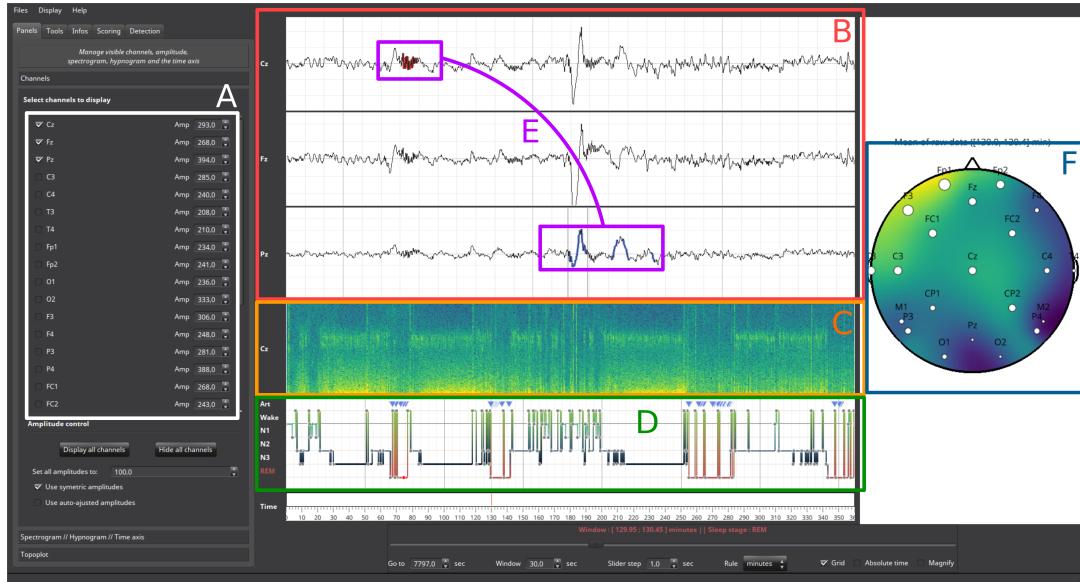


FIGURE 5.5 – Exemples des principales fonctionnalités de *Sleep*, (A) Possibilité de choisir les channels à afficher et contrôle indépendant des amplitudes, (B) Représentation temporelle des données polysomnographique, (C) Spectrogramme d'un channel, (D) Visualisation de l'hypnogramme et possibilité de l'édition, (E) Détections de spindle et de REM sur deux channels, (F) Exemple de représentation topographique (topoplot)

5.2.3.1.3 Ndviz Le module *Ndviz* a été conçu pour fouiller et explorer des données multi-dimensionnelles. Un des soucis majeurs des étudiants qui ne sont pas familiers avec la programmation est de se faire une image de ce que signifie une matrice et surtout, arriver à gérer les dimensions. Par exemple, des données organisées en $(n_{channels}, n_{points}, n_{essais})$ offrent un certain nombre de visualisation possible à travers les dimensions : essais par essais par channel, la moyenne des essais par channel voir la moyenne à travers certains channels et essais... De plus, pour des données que l'on ne connaît pas, il peut être difficile de rechercher des artefacts, des activités épileptiques... *Ndviz* essaye de répondre à ses différentes problématiques en offrant différentes fonctionnalités :

- Dans tout *Ndviz* il est possible de sélectionner les dimensions à inspecter ce qui permet de se familiariser avec les matrices.
- Possibilité de visualiser plusieurs milliers de signaux disposés en grille en même temps. Cette fonctionnalité est issue d'un exemple de *Vipy* originalelement codé par [Cyrille Rossant](#). Par exemple, pour des données organisées en $(n_{channels}, n_{points}, n_{essais})$, il serait possible d'afficher une grille de $n_{channels}$ lignes et n_{essais} colonnes et où sur chaque point de cette grille serait disposé un signal de n_{points} temporels. Cette fonctionnalité permet donc de visualiser des données comportant au maximum trois dimensions.
- De plus, en sélectionnant trois dimensions on peut aussi visualiser les données sous forme d'image (avec la couleur en guise de troisième dimension) ce qui pourrait par exemple être utile pour inspecter un grand nombre de carte temps-fréquence. Enfin, en sélectionnant deux dimensions, les données peuvent être représentées sous forme linéaire, en nuage de points, en histogramme ou spectrogramme.

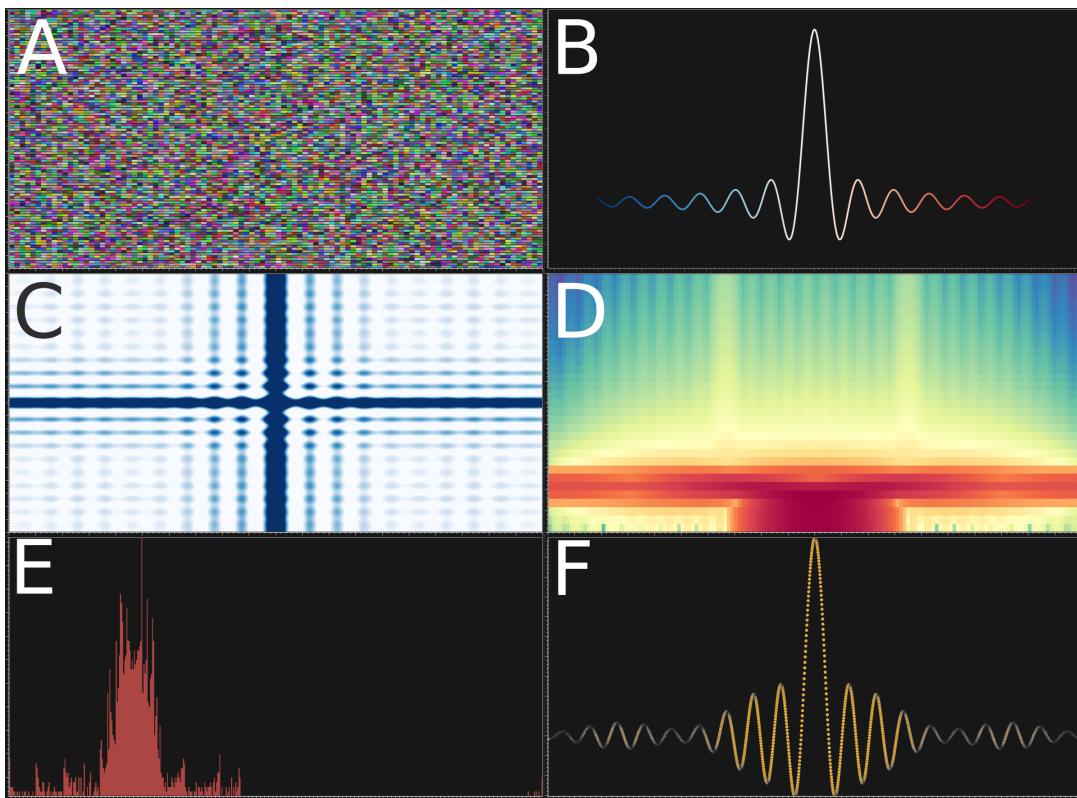


FIGURE 5.6 – Exemples des principales fonctionnalités de *Ndviz*, (A) Visualisation de 40000 signaux disposés dans une grille de (200, 200). Chaque signal fait plusieurs milliers de points et il est possible de zoomer sur chaque signal, (B) Représentation linéaire d'un signal, (C) Exemple de représentation sous forme d'image, (D-E) Calcul du spectrogramme et d'un histogramme d'un signal, (F) Représentation en nuage de points. Cette dernière représentations pourrait être utilisée pour inspecter des features

5.2.3.1.4 Figure Ce dernier module est le plus simple et certainement le plus utile. Il permet de faire des mises en page complexes de figures qui peuvent ensuite être exportées en haute définition et prête à être intégrée dans un papier. Il peut charger des images, les couper, les disposer en grille, ajouter des colorbars (soit pour chaque figure soit des colorbar communes à plus images), contrôler la couleur de l'arrière plan, ajouter des titres à tous les axes...

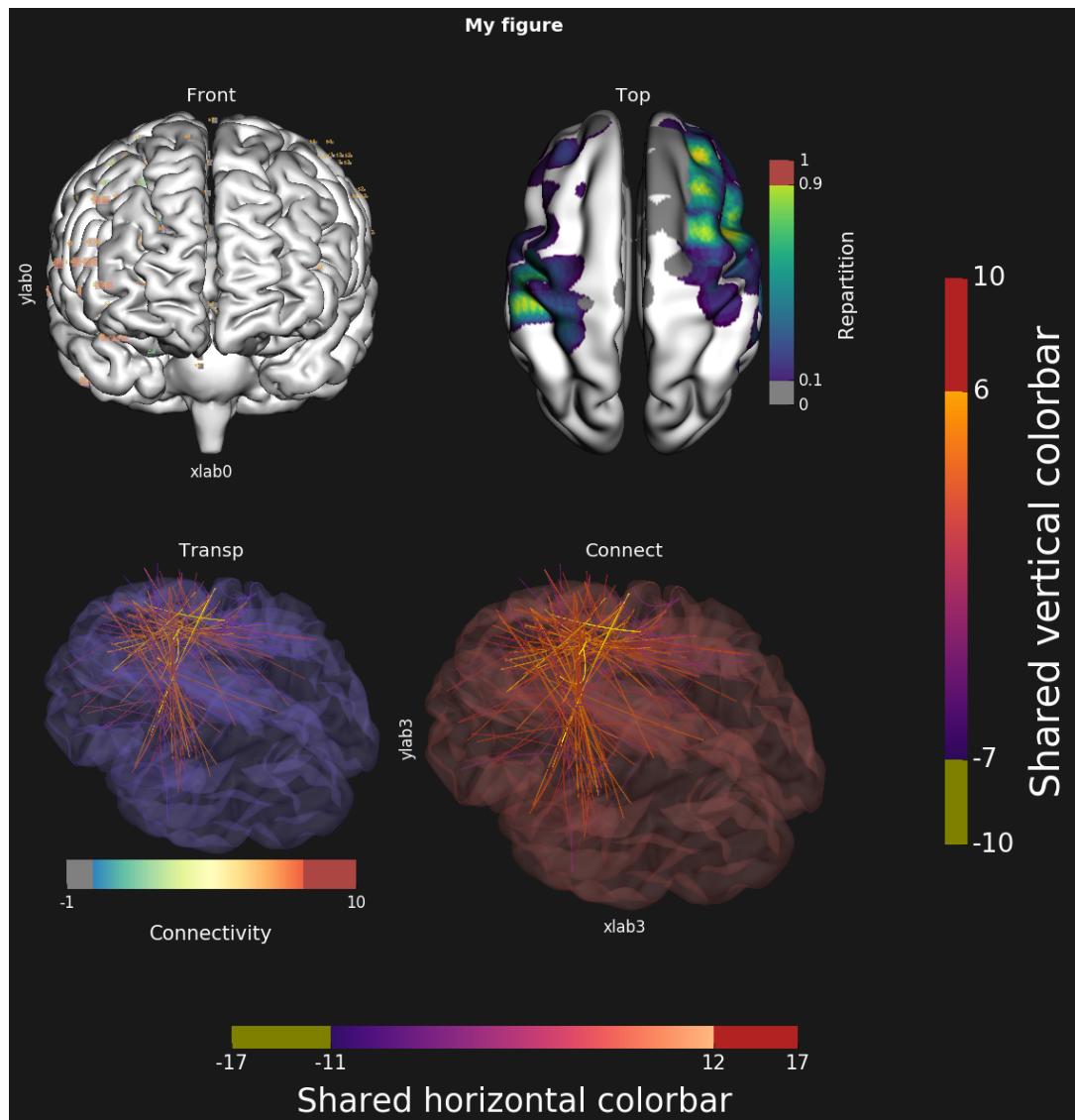


FIGURE 5.7 – Exemple de mise en page avec le module **Figure**

5.2.3.2 Installation et documentation

La procédure d’installation est plus complexe car elle possède plus de dépendances. Elle a donc été décrite plus largement dans la documentation <http://etiennecmb.github.io/visbrain/>. A noter que cette documentation décrit et illustre les fonctionnalités de chaque module et des exemples complets sont également mis à disposition <https://github.com/EtienneCmb/visbrain/tree/master/examples>

DONNÉES EXPÉIMENTALES

Durant cette thèse, l'exploration s'est faite chez l'Homme par le biais, principalement, de données de type Stéréoélectroencéphalographie (SEEG). Ces données rares de très grande qualité (cf. 6.1.2) ont été acquise avant le début de la thèse, ce qui a permis de rentrer dans le vif du sujet très rapidement, après une période d'acclimatation aux différents traitements, propres à ce type d'enregistrement. D'autres types tel que l'EEG, la MEG ou les micro-électrodes ont également été approchés mais de manière ponctuelle, comme ce fût le cas dans l'Étude 1 (cf. II) ou dans différentes collaborations. Toutefois, étant donné que le temps consacré à ces données ne représente qu'une faible portion du travail total, nous allons ici nous concentrer uniquement sur l'intra.

Pour commencer, nous verrons ce que l'analyse de la Stéréoélectroencéphalographie a de particulier (richesse des données, qualité, les traitements associés, les avantages et les limitations). Enfin, nous verrons concrètement les enregistrements qui ont été utilisés dans le cadre de cette thèse.

6.1 DONNÉES INTRACRÂNIENNES

6.1.1 Acquisition

La première question que l'on est en droit de se poser, c'est comment est-il possible de travailler, chez l'Homme, avec des enregistrements qui nécessitent une implantation invasive, c'est-à-dire dans le cortex? Certaines personnes présentent des formes agressives d'épilepsies, pouvant s'avérer pharmacorésistantes. En fonction de la localisation du foyer épileptogène, les méfaits engendrés par les décharges épileptiques peuvent être variés. Dans ce cas, il est nécessaire de localiser ce foyer avec, de préférence, des techniques non-invasives telles que l'EEG ou la MEG. Mais si ces dernières ne permettent pas une localisation précise le patient sera implanté avec des macro-électrodes comme la SEEG pour tenter de localiser puis d'enlever ce foyer par intervention chirurgicale. Cette implantation a un second objectif, déterminer quel est le rôle fonctionnel de la structure lésée (rôle moteur, langage, vision...). C'est dans ce contexte que les chercheurs proposent au patient de participer à une étude scientifique.

6.1.2 Avantages et limitations

Le paragraphe précédent met en exergue la rareté de ces données. De plus, ce type d'acquisition enregistre l'activité cérébrale d'une population relativement restreinte de neurones. En conséquence, on peut espérer que ce petit groupe s'active

dans des processus précis et ainsi, étudier des phénomènes fins. Enfin, le rapport signal sur bruit (RSB) de la SEEG est excellent, ce qui doit permettre l'étude de processus, même en essais unique là où d'autres types d'enregistrements auront besoin d'une large banque d'essais avant de pouvoir constater l'émergence d'un phénomène.

La SEEG présente toutefois quelques limitations que l'on peut nuancer. Le problème majeur est certainement la généralisation d'un phénomène ou la reproductibilité à travers les sujets. La pathologie est propre à chaque patient, donc son implantation aussi. Ce qui signifie qu'il n'y a aucune chance que plusieurs sujets présentent rigoureusement la même implantation. Pour contourner cette limitation, on pourra utiliser :

- Des régions d'intérêt (ROI) : on va regrouper les électrodes des différents sujets par "proximité" en faisant l'hypothèse que celles-ci s'activent de façon similaire face à un processus. Ces ROI pourront être par exemple les gyrus ou les aires de Brodmann. Bien sûr, ce que l'on gagne en généralisation, on le perd en précision.
- Projection corticale : autour de chaque électrode, on définit une sphère d'intérêt (généralement 10 millimètres de rayon) puis on prend l'intersection de ces sphères avec la surface du cerveau. Cette technique permet une visualisation des activités proches de la surface à travers les sujets mais on perd de la lisibilité sur ce qui se passe en profondeur.

Une utilisation combinée des ROI et de la projection corticale permet de palier, au moins partiellement, au problème de reproductibilité inter-sujets.

Autre limitation, on ne dispose que d'une couverture partielle puisque le neurochirurgien implante une quantité limitée d'électrodes. Ce dernier point est traité en augmentant le nombre de sujets. Enfin, la dernière limitation que l'on soulèvera ici, concerne le fait de travailler sur un cerveau "malade" empêchant donc une généralisation à des sujets sains. On limite ce problème par un ensemble de prétraitements ([Jerbi et al., 2009b](#)) décrits dans le prochain paragraphe.

Un dernier point que l'on peut argumenter à la fois comme avantage ou limitation, c'est de ne pas pouvoir contrôler l'implantation pour étudier un phénomène précis. Par exemple, si l'on analyse l'encodage moteur, on s'attendrait à concentrer les efforts sur le cortex moteur primaire ou pré-moteur. Or l'implantation SEEG peut très bien contenir du frontal, du pariétal, du temporal. Là où finalement on peut considérer ça comme un avantage, c'est que l'on a accès à un ensemble de structures jugées non-primordiales mais dont l'ajout pourrait permettre la compréhension d'un processus de manière plus globale.

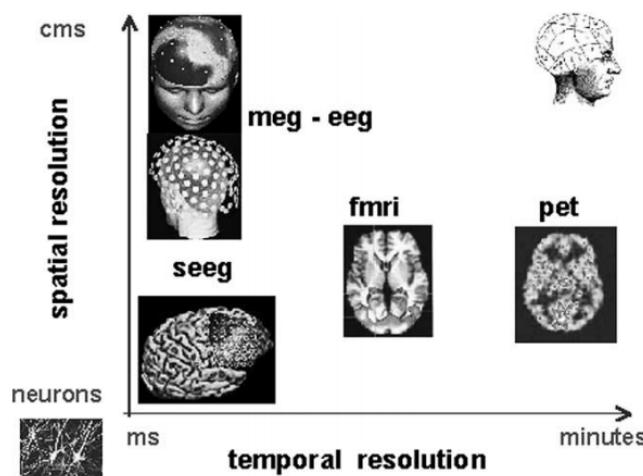


FIGURE 6.1 – Comparatif de résolution spatiale et temporelle pour différentes techniques d'imagerie (Lachaux et al., 2003). La SEEG offre à la fois une résolution spatiale équivalente à la PET ou fMRI et une résolution temporelle proche de celle de la MEG et de l'EEG ce qui en fait une technique de choix, sans compromis de résolution.

6.1.3 Inspection visuelle

BrainTV (Lachaux et al., 2007, Jerbi et al., 2009a)

6.1.4 Prétraitements

Le premier prétraitement appliqué a été une réjection des sites bruités ou présentant une activité pathologique, c'est-à-dire des décharges épileptiques. C'est par cette inspection manuelle que l'on augmente le potentiel de généralisation aux sujets sains.

Autre prétraitement, les données peuvent être bipolarisées comme c'est le cas dans de nombreuses études (Bastin et al., 2016, Ossandon et al., 2011, Jerbi et al., 2009b). La bipolarisation part du principe que deux sites proches enregistrent des activités neuronales différentes mais que toute source de bruit, ou influence de sources lointaines, se retrouvera sur ces deux sites. La technique de bipolarisation consiste donc à soustraire les activités neuronales de sites proches ce qui a pour effet de supprimer la partie commune, le bruit. Par exemple prenons une électrode contenant les sites k9, k10, k11 et k12. Après bipolarisation, on considérera les sites matérialisés par k10 – k9, k11 – k10 et k12 – k11. Les bénéfices de la bipolarisation peuvent être résumés par :

1. Limitation des influences des sources lointaines et de la tension secteur (50hz)
2. Augmentation de la spécificité qui, pour un site bipolarisé, est estimée à 3mm (Kahane et al., 2006, Lachaux et al., 2003, Jerbi et al., 2009b)

6.2 DONNÉES D'ÉTUDE

Trois jeux de données intracrâniaux ont été exploré :

1. *Center-out* : étude de l'encodage et du décodage des actions et des intentions motrices lors de mouvements de main
2. *Occulo* : étude des intentions et décision de mouvements oculaires
3. *Emotions* : étude de l'encodage des émotions

6.2.1 Données *Center-out*

C'est le jeu de données qui a été le plus largement exploité. En effet, celui-ci a servi à étudier l'encodage (cf. III) et le décodage (cf. IV) des actions motrices chez l'homme.

6.2.1.1 Descriptif des données

Six sujets (six femmes), implantés au département de l'épilepsie de l'hôpital de Grenoble ont donné leur consentement écrit pour passer l'expérience, sous la supervision du personnel médical. Le tableau 6.2.1.1 résume les détails clinique des différents sujets.

	Dominance	Age	Genre	Zone épileptique
P ₁	D	19	F	Frontal (RH)
P ₂	D	23	F	Frontal (LH)
P ₃	D	18	F	Frontal (RH)
P ₄	D	18	F	Frontal (RH)
P ₅	D	31	F	Insula (RH)
P ₆	D	24	F	Frontal (LH)
Moyenne : 22.17 ± 4.6				

FIGURE 6.2 – Détails cliniques des sujets ayant participé à la tâche *Center-out*

6.2.1.2 Matériel d'acquisition

De 12 à 15 multi électrodes ont été implantées dans différentes structures. Chaque multi électrode possède entre 10 et 15 sites mesurant 0.8mm et séparés de 1.5mm. La localisation anatomique des électrodes s'est faite en utilisant le schéma d'implantation (exemple en annexe A.6) et l'atlas proportionnel de Talairach et Tournoux (Talairach and Tournoux, 1993). La visualisation de la pré-implantation s'est faite par *IRM – 3D* et un *CT – scan* a été utilisé pour la post-implantation. Enfin, un *IRM* a également servi pour visualiser les électrodes implantées dans la matière blanche. Les coordonnées Talairach ont été déduites du *CT – scan* puis ont été transformées en MNI afin de pouvoir les superposer dans un cerveau standard (cf. figure ci-dessous).

Le système Micromed a été utilisé pour visionner l'acquisition de l'activité neuronale. Une électrode prise dans la matière blanche a été prise comme référence et un filtrage *passband* entre [0.1, 200hz] a également été effectué *online*. La fréquence d'échantillonnage est de 1024hz.

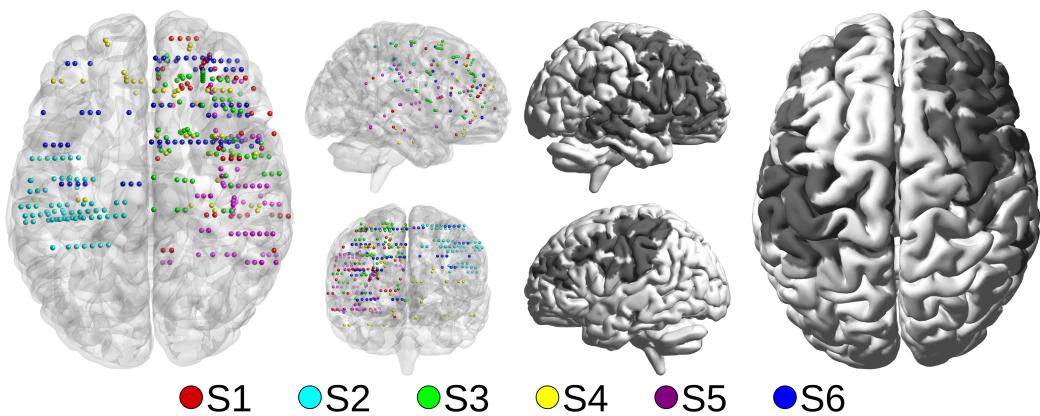


FIGURE 6.3 – Implantation intracrâniale et couverture corticale de six sujets épileptiques ayant passé la tâche Center-out

6.2.1.3 Descriptif de la tâche

La tâche est composée de trois phases :

1. Phase de repos : on demande au sujet de rester immobile pendant une durée de une seconde
2. Phase de préparation motrice (**CUE 1**) : une direction est imposée à l'écran (haut/bas/gauche/droite). On demande au sujet de se préparer pendant 1.5s à bouger la souris dans la direction imposée.
3. Phase d'exécution motrice (**CUE 2**) : le sujet exécute le mouvement en bougeant la souris du centre à l'extrémité de l'écran indiquée (environ 1.5s) puis de revient vers le centre.

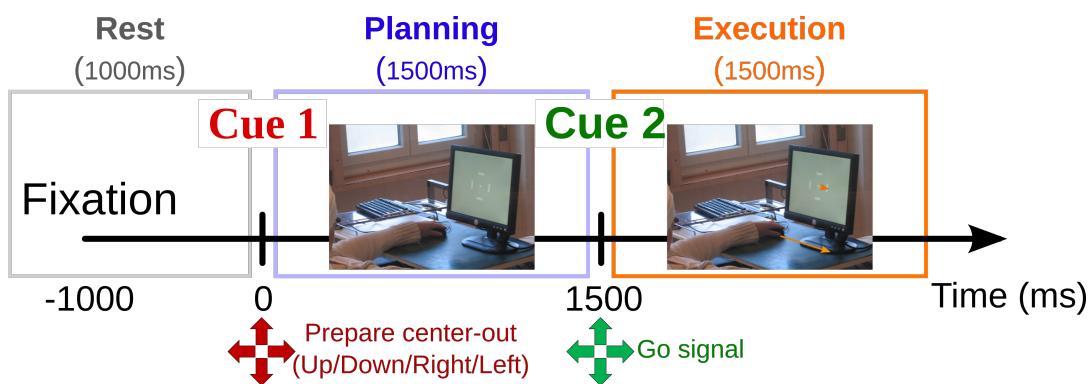


FIGURE 6.4 – Descriptif de la tâche Center-out

Cette tâche à conduit à deux études essentielles :

- L'étude de l'encodage des intentions motrices (cf. III) : on étudiera le décodage du repos vs préparation, repos vs exécution et de la préparation vs exécution
- L'étude du décodage des directions de mouvement (cf. IV) que ce soit pendant la préparation ou pendant l'exécution motrice.

6.2.2 Autres données

- Occulo : Données occulo mais un seul sujet donc pas super cool. De plus, les résultats sont un peu vieux et mériteraient de se pencher dessus une fois pour toute.
- Emotions : Ce serait pas mal que j'ai quelques résultats sur les émotions, histoire de montrer aussi un peu la diversité des données utilisées.

6.3 DELAYED TASK : PROTOCOLE EXPÉRIMENTAL

okok

Deuxième partie

Étude 1 : Niveau de chance et évaluation statistique des résultats de classification par apprentissage supervisé

INTRODUCTION

L'utilisation des outils de *machine-learning*, dans le domaine des neurosciences, est de plus en plus fréquente et se sert de ces méthodes comme outils de validation. En effet, l'extraction d'attributs puis la classification de ceux-ci permet d'explorer ce qui, dans l'activité neuronale, reflète un changement d'état. Il y a donc un besoin d'évaluer la performance de décodage et sa validité statistique et donc quantifier à quel point ce décodage s'éloigne de ce que l'on pourrait atteindre par chance.

Lorsque l'on parle de chance dans le domaine de l'apprentissage machine, il faut distinguer la chance théorique de la chance empirique. Des problèmes à deux classes, à quatre classes ou à huit classes auront respectivement des seuils de chance de 50%, 25% et 12.5%. Il faut considérer ces seuils à titre indicatif, comme première estimation, puisque en réalité ceux-ci sont atteints pour une infinité d'échantillons. Pour comprendre ce problème de seuil de chance, prenons un exemple basé sur des pièces. On sait que le seuil de chance d'avoir soit pile soit face est de 50%. Si on lance quatre pièces en même temps ce seuil de 50% est exprimé lorsqu'il y a autant de piles que de faces, donc deux pièces chacun. Maintenant, si une pièce ne représente pas son groupe, les répartitions sont de 75% contre 25%. Avec 10 pièces, si une pièce ne représente pas son groupe on obtient 60% contre 40%. Pour 100 pièces on a 51% contre 49% etc. Plus le sac contient de pièces, plus le seuil de chance pratique va tendre vers le seuil de chance théorique. Si les spécialistes du *machine-learning* ont parfaitement conscience de ce problème, la communauté neuro-scientifique affiche parfois un manque de rigueur face à cette limitation.

Cette étude, qui se destine principalement les étudiants ou aux personnes débutant dans le domaine, a pour objectif de quantifier ce problème du nombre d'essais et de présenter les outils pour le mesurer. Pour cela, nous avons générer des signaux Gaussien aléatoires et nous avons mesuré l'impact du nombre d'essais sur le seuil de chance ainsi que sur les principaux outils du *machine-learning* (classifieur / cross-validation / nombre de répétitions). En effet, nous avons cherché à savoir si les classificateurs (Linear Discriminant Analysis , Naive Bayes , Support Vector Machine) étaient affectés de la même façon pour des petits échantillons, si certains types de cross-validation (k-Fold et Leave-One-Out) étaient plus robustes et si le nombre de répétitions permettaient de minimiser cet effet. De plus, nous rappelons que des outils analytiques (loi binomiale) ou empiriques (permutations) préviennent de ce problème du nombre d'échantillons en introduisant des valeurs statistiques. Enfin, nous avons répliqué cette étude sur des données MEG et iEEG et fournissons une toolbox Matlab pour reproduire ces résultats et contenant les outils statistiques présentés.



Computational Neuroscience

Exceeding chance level by chance: The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy

Etienne Combrisson ^{a,b}, Karim Jerbi ^{a,c,*}

^a DYCOG Lab, Lyon Neuroscience Research Center, INSERM U1028, UMR 5292, University Lyon I, Lyon, France

^b Center of Research and Innovation in Sport, Mental Processes and Motor Performance, University of Lyon I, Lyon, France

^c Psychology Department, University of Montreal, QC, Canada

ARTICLE INFO

Article history:

Received 28 July 2014

Received in revised form 6 January 2015

Accepted 7 January 2015

Available online 14 January 2015

Keywords:

k-Fold cross-validation

Small sample size

Classification

Multi-class decoding

Brain-computer-interfaces (BCIs)

Machine learning

Binomial cumulative distribution

Classification significance

Decoding accuracy

MEG

ECoG

Intracranial EEG

ABSTRACT

Machine learning techniques are increasingly used in neuroscience to classify brain signals. Decoding performance is reflected by how much the classification results depart from the rate achieved by purely random classification. In a 2-class or 4-class classification problem, the chance levels are thus 50% or 25% respectively. However, such thresholds hold for an infinite number of data samples but not for small data sets. While this limitation is widely recognized in the machine learning field, it is unfortunately sometimes still overlooked or ignored in the emerging field of brain signal classification. Incidentally, this field is often faced with the difficulty of low sample size. In this study we demonstrate how applying signal classification to Gaussian random signals can yield decoding accuracies of up to 70% or higher in two-class decoding with small sample sets. Most importantly, we provide a thorough quantification of the severity and the parameters affecting this limitation using simulations in which we manipulate sample size, class number, cross-validation parameters (*k*-fold, leave-one-out and repetition number) and classifier type (Linear-Discriminant Analysis, Naïve Bayesian and Support Vector Machine). In addition to raising a red flag of caution, we illustrate the use of analytical and empirical solutions (binomial formula and permutation tests) that tackle the problem by providing statistical significance levels (*p*-values) for the decoding accuracy, taking sample size into account. Finally, we illustrate the relevance of our simulations and statistical tests on real brain data by assessing noise-level classifications in Magnetoencephalography (MEG) and intracranial EEG (iEEG) baseline recordings.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Applying machine learning algorithms to brain signals in order to predict intentions or decode cognitive states has become an increasingly popular technique over the last decade. The surge in the use of machine learning methods in neuroscience has been largely fueled by the tremendous increase in brain-computer interface (BCI) and brain signal decoding research either using non-invasive recordings such as Electroencephalography (EEG) or Magnetoencephalography (MEG) (e.g. Aloise et al., 2012; Besserve et al., 2007; Jerbi et al., 2011; Krusienski and Wolpaw, 2009; Toppi et al., 2014; Waldert et al., 2008) or with intracranial EEG (e.g. Ball et al., 2009; Derix et al., 2012; Hamamé et al. (2012); Korczyn

et al. (2013); Lachaux et al., 2007a,b; Leuthardt et al., 2004, 2006; Mehring et al., 2004; Pistohl et al., 2012; Schalk et al., 2008; Jerbi et al., 2007a,2009a,2013). Machine learning and signal classification techniques are powerful and complex tools that have to be used with caution. While most machine learning experts are well aware of the various caveats to watch out for, certain theoretical limitations of these methods can easily elude students and neuroscience researchers new to the field of machine learning and brain-computer interface research.

In supervised learning, samples of a subset of the data and knowledge of their corresponding class (label) are used to train a model to distinguish between two or more classes. The trained classifier is then tested on the remaining data samples (the hold-out samples). This procedure is generally repeated several times by varying the subsets used for training and those used for testing, a standard procedure known as cross-validation. The percent of over-all correct label (or class) prediction across the test samples of the multiple folds is known as the correct classification

* Corresponding author at: Psychology Department, University of Montreal, QC, Canada.

E-mail address: karim.jerbi@umontreal.ca (K. Jerbi).

rate (sometimes called decoding accuracy). Conversely, the mean of misclassified samples over the folds is a measure of classifier prediction error.

The performance of a classifier in neural decoding studies is often assessed by how close its correct classification rate is to the maximum of 100%, or alternatively, how strongly it departs from the *chance-level* rate achieved by a classifier that would randomly associate the samples to the various classes. For instance, in a two-class or four-class classification problem, the probabilistic chance level indicating totally random classification is 50% or 25% respectively. Yet, although such probabilistic chance-levels widely applied in brain signal classification studies, they can be problematic because they are strictly speaking only valid for infinite sample sizes. While it will not come to anyone as a surprise that no study to date was able to acquire infinite data, it is intriguing how rarely brain signal classification studies acknowledge this limitation or take it into account. For a two-class classification problem with small sample size, 60%, 70% or even higher decoding percentages can in theory arise by chance (see simulation results below). As a consequence, for finite samples, a decoding percentage can only be considered reliable if it substantially, or better still, *significantly* departs from the theoretical level in statistical terms. But how can we assess the significance of the departure of a decoder from the outcome of total random classification? For a given sample size and a given number of classes, what would be the statistically significant threshold of correct classification that one needs to exceed in order to consider the decoding *statistically significant*? Although these questions have been widely recognized and addressed in the machine learning field (e.g. Kohavi, 1995; Martin and Hirschberg, 1996a,b), it is unfortunately often overlooked in the emerging field of brain signal classification which, incidentally, is often faced with low sample sizes for which the problem is even more critical.

Not all the previous brain decoding reports suffer from the caveat of using theoretical chance-level as reference. However, numerous studies only apply statistical assessment when testing for significant differences between the performance of multiple classifiers, or when comparing decoding across experimental conditions, but unfortunately neglect to provide a statistical assessment of decoding that accounts for sample size (e.g. Felton et al., 2007; Haynes et al., 2007; Bode and Haynes, 2009; Kellis et al., 2010; Hosseini et al., 2011; Sitaran et al., 2011; Hill et al., 2006; Wang et al., 2010; Bleichner et al., 2014; Babiloni et al., 2000; Ahn et al., 2013; Morash et al., 2008; Neuper et al., 2005; Kayikcioglu and Aydemir, 2010; Momennejad and Haynes, 2012). A number of such studies use theoretical percent chance-levels (e.g. 50% in a 2-class classification) as a reference against which classifier decoding performance is assessed. By doing so, such studies fail to account for the effect of finite sample size. This may have little effect in the case of large sample size or when extremely high decoding results are obtained, however, the bias and erroneous impact of such omissions can be critical for smaller sample sizes or when the decoding accuracies are barely above the theoretical chance levels.

Note however, that the rigorous assessment of significant classification thresholds is not equally ignored across the various types of neuronal decoding studies; it seems that the omissions (or unfortunate tendency to rely on the theoretical chance levels) are more common in more recent sub-branches of the neuronal decoding field. This is the case for signal classification and BCI studies based on non-invasive (fMRI, EEG and MEG) brain recordings in humans, and possibly electrocorticographic macro-electrode recordings in patients, where the methods (including classifiers, features and statistics) are less well-established than in the field of neuronal spike decoding in primates for instance.

In this brief article, we address caveats related to interpreting brain classification performances with small sample sizes. The paper is written with the broad neuroscience readership in mind

and is oriented, in particular, to students and researchers new to neural signal classification. First of all, we describe how applying signal classification to randomly generated signals can yield decoding accuracies (correct classification rates) that strongly depart from theoretical chance levels, with values up to 70% and higher with small sample sizes (instead of the expected theoretical 50% for 2-class decoding). Most importantly, we illustrate and quantify the phenomenon by using simulations in which we manipulate sample size, class number, cross-validation parameters and classifier type. In addition to raising a red flag of caution, we recommend practical alternatives to overcome the problem. We describe a straight-forward method to derive a statistically significant threshold that accounts for sample size and provides confidence intervals for the classification accuracy achieved by cross-validations. A reference table is also provided to allow readers to quickly look-up the percent correct classification thresholds that need to be exceeded in order to assert statistical significance of the findings for a range of possible sample sizes, classes and significance levels.

2. Materials and methods

2.1. Data simulation and classification

2.1.1. Generating normally distributed random data

In order to simulate a situation with classification results that approach the theoretical chance level, we generated 100 data sets of zero-mean Gaussian white noise. The normally distributed variables in each data set were generated in MATLAB (Mathworks Inc., MA, USA) via a pseudo-random number generator. Each one of the 100 data sets was randomly split into c subsets data (here we used $c = 2$ - or 4-classes) and we then evaluated the classification performance obtained by applying different classification algorithms to these simulated datasets. Because the variables in each 'simulated class' were drawn from the exact same Gaussian random distribution data set, applying supervised machine learning algorithms should fail to distinguish between classes and should theoretically yield chance-level classification rates (50% for $c = 2$ and 25% for $c = 4$). To examine the effect of sample size on how close the empirical classifications are to the theoretically expected chance level we varied the total number of samples n from 24 to 500. In other words, in the 2-class simulation for instance, the number of samples in each class varied from 12 to 250. Note that the code we implemented for the generation of random data for classification purposes is provided online (see Appendix A).

2.1.2. Classification algorithms

We implemented three types of machine learning algorithms: linear discriminant analysis (LDA), naïve Bayes (NB) classifier and a support vector machine (SVM), the latter with two different kernels: a linear kernel and a radial basis function (RBF) kernel. These three methods, which are frequently used for neural signal classification in the context of brain-computer interface research are briefly described in the following.

Linear discriminant analysis: LDA (Fisher, 1936) is a straight-forward and fast algorithm which assumes that the independent variables in each class are normally distributed with identical covariance (homoscedasticity assumption). For a two dimension problem, the LDA tries to find a hyperplane that maximizes the mean distance between the two classes while minimizing the inter-class variance. A multiclass problem can be tackled as a multiple two-class problem by discriminating each class from the rest using multiple hyperplanes.

Naive Bayesian classifier: The NB model (e.g. Fukunaga, 1990) is a probabilistic classifier that assigns features to the class to which they have the highest probability of belonging. NB assumes that the

features in each class are normally distributed and independent. The name arises from the fact that it is based on applying Bayes' theorem with strong (naive) independence assumptions.

Support vector machine: SVM (Boser et al., 1992; Burges, 1998; Cortes and Vapnik, 1995; Vapnik, 1995) classifiers originate from statistical learning theory. An SVM searches for a hyperplane that maximize margins between the hyperplane and the closest features in the training set. For non-linearly separable classes, SVM uses a kernel function to project features in a higher dimensional space in order to reduce the nonlinear problem to a linear one, which is then separable by a hyperplane. The (Gaussian) Radial Basis Function (RBF) kernel is a popular choice. In this study, both linear and RBF kernels were used for SVM classification.

Details of the theoretical background of various classifiers can be found in standard statistics and machine learning textbooks and various reviews (e.g. Lotte et al., 2007; Wieland and Pittore, 2014). Here, we used MATLAB implementation for the LDA and NB and the libsvm library for multi-class SVM.

2.1.3. Repeated and stratified k-fold cross-validation

To compute the decoding accuracy achieved by each one of the classifiers on the random data, we used standard stratified k -fold cross-validation. For a given data set size, all available N samples are partitioned into k folds, where $(k - 1)$ folds are used for training the classifier model (training set) and the remaining fold is used for validation (test set). This procedure is then repeated k times so that each fold is used once as test set. The stratified option ensures that each fold has approximately the same proportion of samples from each class as in the original dataset as a whole. The case $k = N$ (e.g. 200 folds in a data set of 200 samples) is called leave-one-out (LOO) cross-validation because one element is used to test the performance of a classifier trained on the rest of the data. Because k -fold cross-validation involves a random partition, the variance of the classifier can in theory be reduced by repeating the full cross validation procedure q times. Therefore, in addition to testing different classifier types, this study explores the effect of the following parameters: n (sample size, 20–500), k (number of cross-validation folds: 5, 10 and leave-one-out) and q (number of repetitions: 1, 5 and 20).

2.2. Statistical significance of classification using a binomial cumulative distribution

For a given number of classes c , the percent theoretical chance level of classification is given by $100/c$. For example, for a 4-class problem, the chance level is $100/4 = 25\%$. This threshold is based on the assumption of infinite sample size. In practice, the empirical chance level depends on the number of samples available. One way to address this limitation is to test for the statistical significance of the decoding accuracy. This can be done by assuming that the classification errors obey a binomial cumulative distribution, where for a total of n samples and c classes, the probability to predict the correct class at least z times by chance is given by:

$$P(z) = \sum_{i=z}^n \binom{n}{i} \times \left(\frac{1}{c}\right)^i \times \left(\frac{c-1}{c}\right)^{n-i}$$

Although neural signal classification studies predominantly evaluate decoding performance by how well the results depart from the theoretical chance level, several BCI studies have in addition, used the binomial cumulative distribution to derive statistical significance thresholds (e.g. Ang et al., 2010; Demandt et al., 2012; Pistohl et al., 2012; Waldert et al., 2007, 2008, 2012). In this study, we use the MATLAB (Mathworks Inc., MA, USA) function *binoinv* to compute the statistically significant threshold $St(\alpha) = binoinv(1 - \alpha, n, 1/c) \times 100/n$, where α is the significance level given by $\alpha = z/n$

(i.e. the ratio of tolerated false positives z – i.e. number of observations correctly classified by chance with respect to all observations n). For instance, for a sample size of $n = 40$ and a 2-class classification problem ($c = 2$), computing the threshold for statistical significance of the decoding at $\alpha = 0.001$ using the above formulation yields 70.0%. In other words, at $n = 40$, any decoding percentage below 70% is not statistically significant (at $p < 0.001$), whereas if one relied on the theoretical threshold for two classes (i.e. 50%) a decoding accuracy of 67% might have been considered relevant. Table 1 provides the minimal thresholds as a function of selected sample sizes, class number and significance levels. Note that code for the calculation of these analytical significance levels is provided online (see Appendix A).

2.3. Statistical significance of classification using permutation tests

The statistical significance of decoding can also be assessed by non-parametric statistical methods, namely using permutation tests (Good, 2000; Nichols & Holmes, 2002). By randomly permuting the observations across classes and calculating classification accuracy at each permutation, it is possible to establish an empirical null distribution of classification accuracies on random observations. The tails of this distribution can then be used to determine significance boundaries for a given rate of tolerated false positives (i.e. correct classifications that occur by chance). For instance, if the original (without randomization) classification accuracy is higher than the 95 percentile of empirical performance distribution established by randomly permuting the data, then one can assert that the original classification is significant with $p < 0.05$. The advantage of this empirical approach is that it does not require particular assumption about statistical properties of the samples.

An intuitive illustration of this procedure would be as follows: one performs for example 99 random permutations of the labels (classes) in the data and computes the classification accuracy for each permutation. This provides an empirical distribution of 99 classification accuracy values. Now if the classification performance obtained with the original (unpermuted) data is higher than the maximum of the empirical distribution, one can conclude that it is significant with $\alpha = 0.01$.

Permutations test provide a useful empirical approach to deriving statistical significance of classifier performance (e.g. Golland and Fischl, 2003; Ojala and Garriga, 2010; Meyers and Kreiman, 2011). To demonstrate the utility to derive significance boundaries as a function of sample size and thus compare it to the use of the binomial formula. To this end, we used simulated random data with associated labels (as described in Section 2.1) and computed the classification performance (using LDA) for 10,000 permutations (randomly exchanging labels of the original observations). From this we derived the accuracy thresholds that correspond to the 99%, 99.9% and 99.99% percentile of the distribution (i.e. $p < 0.01$, $p < 0.001$, and $p < 0.0001$ respectively). This was done for each sample size value n (20–500), which allowed us to depict the evolution of the empirical significance boundaries as a function of sample size. Note that code for the calculation of permutation-based empirical significance levels is provided online (see Appendix A).

2.4. Classification of baseline data from real brain signals

Because real data does not necessarily have the same properties as those implemented in our random data simulations (zero-mean Gaussian white noise), we also calculated the correct classification rate (as a function of sample size) that is achieved when classifying real brain data that do not contain any true discrepancies. This was carried out for pre-stimulus or baseline recordings in MEG (4 subjects) and with intracranial EEG recordings (4 patients). The

Table 1

Look-up table for statistically significant classification performance. Minimal correct classification rate (%) to assert statistical significance (at a given p -value) as a function of sample size n and number of classes c . Threshold values are based on the binomial cumulative distribution function and are rounded to the first digit.

n	c	2-Classes				4-Classes				8-Classes			
		$p < 0.05$	$p < 0.01$	$p < 10^{-3}$	$p < 10^{-4}$	$p < 0.05$	$p < 0.01$	$p < 10^{-3}$	$p < 10^{-4}$	$p < 0.05$	$p < 0.01$	$p < 10^{-3}$	$p < 10^{-4}$
20	70.0%	75.0%	85.0%	90.0%	40.0%	50.0%	55.0%	65.0%	25.0%	30.0%	40.0%	45.0%	
40	62.5%	67.5%	75.0%	77.5%	37.5%	42.5%	47.5%	52.5%	22.5%	25.0%	30.0%	35.0%	
60	60.0%	65.0%	70.0%	73.3%	35.0%	38.3%	43.3%	46.7%	20.0%	23.3%	26.7%	30.0%	
80	58.7%	62.5%	67.5%	70.0%	32.5%	36.2%	41.2%	43.7%	18.7%	21.2%	25.0%	27.5%	
100	58.0%	62.0%	65.0%	68.0%	32.0%	35.0%	39.0%	42.0%	18.0%	21.0%	24.0%	26.0%	
200	56.0%	58.0%	61.0%	63.0%	30.0%	32.5%	35.0%	37.0%	16.5%	18.0%	20.0%	22.0%	
300	54.7%	56.7%	59.0%	60.7%	29.0%	31.0%	33.0%	34.7%	15.7%	17.0%	18.7%	20.0%	
400	54.0%	55.7%	57.7%	59.2%	28.5%	30.0%	31.7%	33.2%	15.2%	16.5%	17.7%	19.0%	
500	53.6%	55.2%	57.0%	58.2%	28.2%	29.6%	31.2%	32.4%	15.0%	16.0%	17.2%	18.2%	

rationale here is that baseline (pre-stimulus) data is not expected to show any genuine discriminative brain patterns related to post-stimulus events, and as such, it is comparable to random background noise. Therefore, signal classification on these baseline periods should fail, and the accuracies that classifiers achieve can be taken as an empirical representation of chance-level decoding.

2.4.1. Illustrative data from MEG rest activity

We used illustrative data from 4 subjects scanned with a whole-head MEG system (151 sensors; VSM MedTech, BC, Canada) acquired at 1250 Hz sampling rate and with a band pass filter of 0–200 Hz. The participants provided written informed consent, and the experimental procedures were approved by the Institutional Review Board and by the National French Science Ethical Committee. The MEG data segments used for the purpose of the current analysis were extracted from the pre-stimulus baseline of a visuomotor MEG experiment (Jerbi et al., 2007b), and each trial was assigned one of 2 (or of 4) arbitrary labels for the 2-class (or 4-class) classification. Oscillatory alpha (8–12 Hz) power was computed using Hilbert transform and subsequently used as feature in an LDA-based classification procedure. We used 10-fold cross-validation and the whole procedure was repeated for increasing values of trial numbers (sample size n) ranging from 20 to 200 (in steps of 8).

2.4.2. Illustrative data from intracranial EEG baseline activity

We used illustrative data from 4 epilepsy patients stereotactically implanted with intracranial depth electrodes (0.8 mm diameter, 10–15 contact leads, DIXI Medical Instruments, Besançon, France). The intracerebral EEG (iEEG) recordings were conducted using a video-SEEG monitoring system (Micromed, Treviso, Italy), which allowed for the simultaneous recording from 128 depth-EEG electrode sites (More details of the routine SEEG acquisitions in Jerbi et al., 2009b). The data were bandpass filtered online from 0.1 to 200 Hz and sampled at 1024 Hz. The recordings were performed at the epilepsy department of the Grenoble University Hospital (headed by Dr. Philippe Kahane). All participants provided written informed consent, and the experimental procedures were approved by the Institutional Review Board and by the National French Science Ethical Committee.

The data segments used here were extracted from the pre-stimulus (baseline) of a standard motor task and each trial was associated with one of 2 (or of 4) labels for the 2-class (or 4-class) classification. The labels assigned to each pre-stimulus baseline trial were in fact the genuine post-stimulus events for the same trials (but no true discrimination can be expected prior to stimulus onset as the post-stim event could not be known or inferred during the pre-stimulus period). Broadband gamma (60–250 Hz) power was computed using Hilbert transform and subsequently used as feature in an LDA-based classification procedure. As for the MEG

data, we used 10-fold cross-validation and the whole procedure was repeated for increasing values of trial numbers (sample size n) ranging from 20 to 200 (in steps of 8).

3. Results

3.1. Empirical evaluation of chance level decoding as a function of sample size

Fig. 1 shows the decoding accuracies obtained by conducting 10-fold cross validation on 100 randomly generated data sets. The decoding is depicted as a function of increasing sample size (from 24 to 500) and for the case of 2-class (left column) and 4-class (right column) classification. Although the theoretical chance levels for these configurations are 50% and 25% respectively, the results show how much the empirical decoding accuracies obtained with random data deviate from these probabilistic values.

The small sample size problem: as expected, the variance of the decoding accuracy across the 100 simulated random data sets is high, and the more so for small sample sizes. As illustrated in **Fig. 1**, while the decoding does converge toward the theoretical chance level as the sample size increases, the values achieved with small sample size ($n < 100$) can be disturbingly high. For instance, the highlighted examples (solid black line) in panels (a) to (f) illustrate how decoding accuracies as high as 70% for 2-class classification (or 50% for 4-classes) can be observed even when conducting classification on subsets of randomly generated data with randomly associated labels.

The small sample issue is persistent and qualitatively similar across all classifiers used. The first three rows of **Fig. 1** show the results obtained with LDA, NB and SVM (with an RBF kernel). Panels (g) and (h) of **Fig. 1** show that cross-validation results in all three classifiers have comparable deviation across the 100 simulated data sets. The variance of cross-validation over the 100 random data sets is high for small sample sizes (<200 observations) and drops off with increasing sample size.

3.2. Tweaking cross-validation parameters does not solve the small sample problem

It might be tempting to think that changing the cross-validation parameters might be a way to get around the small sample problem illustrated here. To address this we evaluated the impact of varying (a) the number of cross-validation folds, and (b) the number of repetitions of the cross-validation, on the reported deviation of the cross-validation results (cf. **Fig. 1g** and h) across the 100 data sets and all sample sizes. The results in **Fig. 2(a–c)** show that applying 5- and 10-fold cross-validation to the random data yielded substantially the same results, and that leave-one-out (LOO) cross-validation actually provided worse results (i.e.

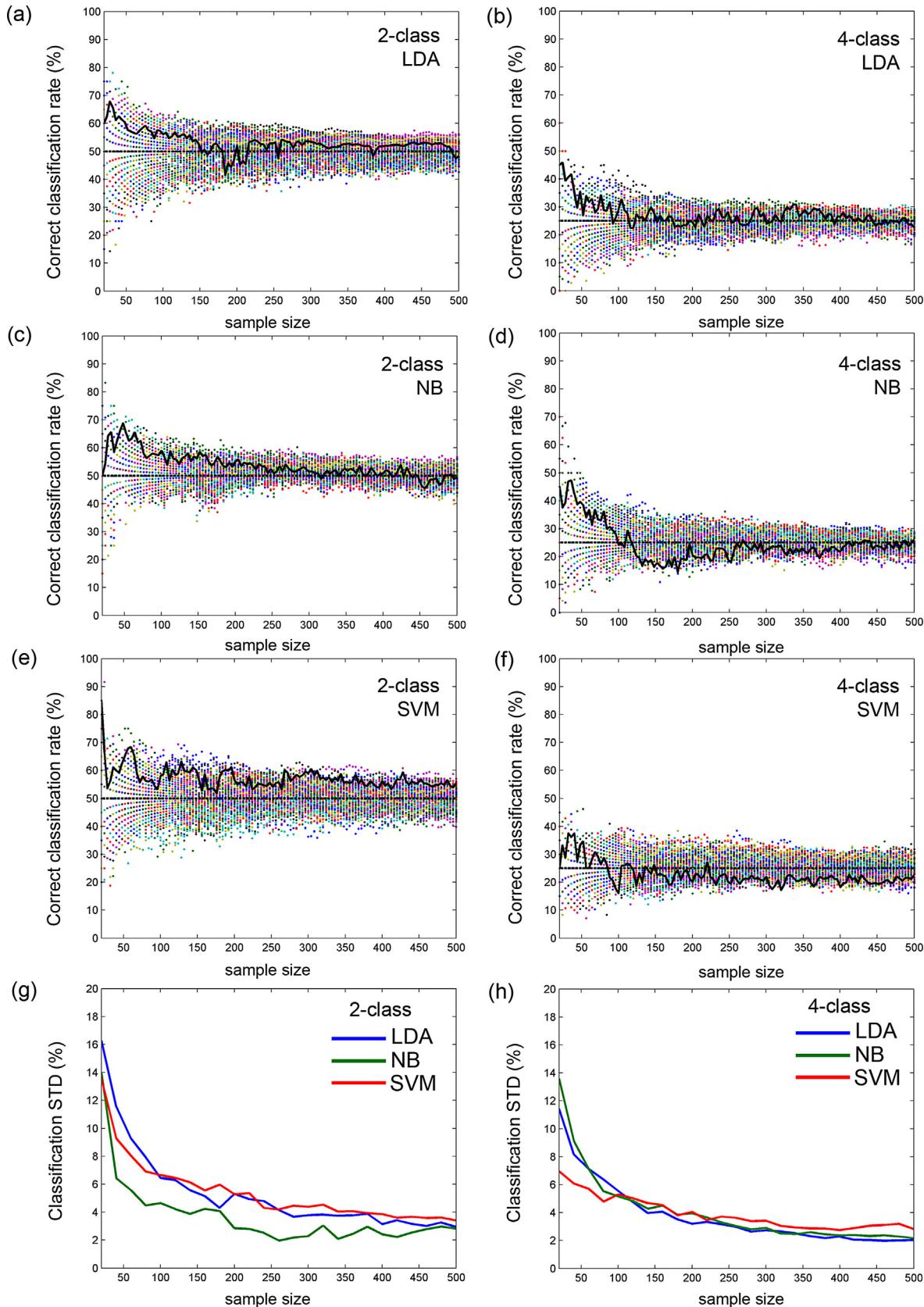


Fig. 1. Classifier decoding rates as a function of sample size when applied to random data sets using 10-fold cross-validation. (a) Two-class LDA classification rate (%) as a function of sample size (empirical results increasingly deviate from the 50% chance-level as the sample size gets smaller). The backline line shows the evolution of cross-validation results for one specific data set out of the 100 depicted in multiple colors. (b) Same as panel (a) but using 4-class classification, i.e. at each sample size n , the data is split into 4 virtual classes instead of two, (c and d) Same as (a and b) but for a Naïve Bayesian classifier. (e and f) Same as (a and b) but for an SVM classifier using an RBF kernel. (g) Evolution of cross-validation standard deviation across the 100 data sets for each of the three classifiers for 2-class decoding. (h) Same as panel (g) but for 4-class decoding.

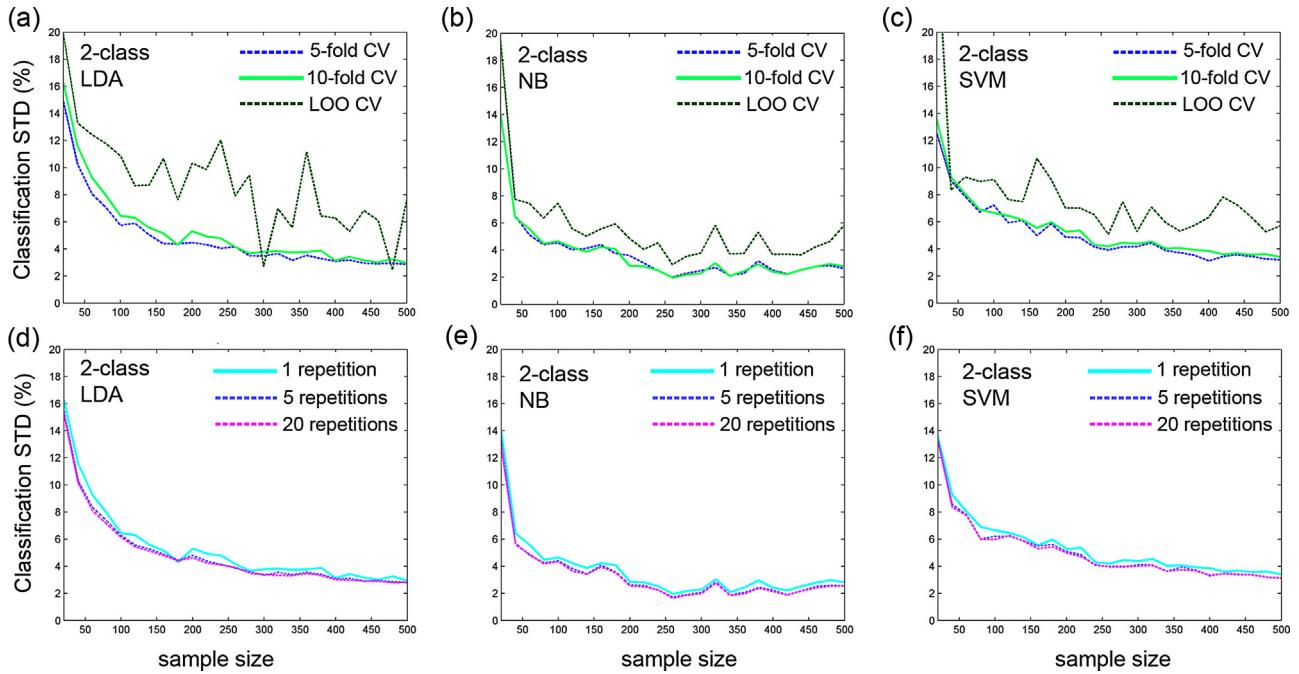


Fig. 2. Effect of cross-validation parameters on the variability of 2-class decoding performance computed across 100 sets of random data. (a–c) Effect of the number of folds (k): drop in cross-validation variance as sample size n increases, shown for $k=5$, $k=10$ (default), and $k=n$ (i.e. leave-one-out) and for all three classifiers LDA (panel a), NB (panel b) and SVM (panel c). (d–f) Effect of cross-validation repetition number: drop in cross-validation variance as sample size n increases, shown for repetition values $q=1$ (default), $q=5$, and $q=20$ and for all three classifiers LDA (panel d), NB (panel e) and SVM (panel f). Note that the strong deviation from 50% chance-level for small sample sizes is persistent across all panels, and appears to be worst for LOO cross-validation with LDA.

higher variance). Moreover, repeating the cross-validation procedure (whether 5 or 20 times) achieved a negligible reduction of variance (Fig. 2d and e). Overall, these observations indicate that neither changing the number of folds nor to the number of overall repetitions has an impact on the variance of decoding accuracy (i.e. cross-validation results) across the 100 sets of Gaussian white noise.

3.3. Estimating statistical significance of decoding accuracy: binomial formula and permutation tests

Panels (a) and (b) in Fig. 3 show the evolution of the minimal statistically significant decoding rate as a function of sample size (respectively for 2- and 4-classes) using the binomial cumulative distribution (described in Section 2.2). The plots depicted for three distinct significance levels (10^{-2} , 10^{-3} and 10^{-4}) all show that the minimal correct decoding rate that is required in order to assert significance, decreases as the number of samples increases. Given small sample sizes (e.g. below 100 observations), to be statistically significant, the decoding accuracy must be substantially higher than the probabilistic chance level. For example, for 40 observations, a 2-class decoding is statistically significant (at $p < 0.001$) only if it exceeds the threshold of 75%. Note that for sample sizes as high as 500 observations, statistical significance still requires correct decoding higher than 55% (at $p < 0.01$), i.e. at least 5% above the theoretical chance level. A more comprehensive overview of the statistical decoding thresholds (wider ranges of p -values and of class number) computed for selected sample sizes (20–500), is provided in Table 1.

Panels (c) and (d) in Fig. 3 depict not only the evolution of the decoding boundaries for 2-class and 4-class decoding, using the binomial formula but also using the permutation test approach (see Section 2.3). Interestingly, the boundaries (for each level of admitted false positives) using both methods are reasonably close. The boundaries obtained with permutations show a slight tendency to

be more restrictive than the binomial formula. While this is a little more apparent for small values of n , the difference between the two methods rapidly vanishes as n increases.

3.4. MEG and iEEG baseline data reveal erroneously high decoding results

Fig. 4 depicts the results of the empirical estimation of *de facto* chance-level decoding in illustrative MEG and iEEG data segments taken during pre-stimulus baseline periods (where no decoding is theoretically expected). Similarly to our findings using random data simulations (Fig. 1 a–f), the baseline MEG and iEEG data trials also led to decoding rates that strongly departed from the theoretical chance levels of 50% for 2-class classification and 25% for 4-class classification. Also in line with the results of the simulated data, the effects observed here were again highest for small sample sizes and dropped off slowly with increasing n . Note that the results in Fig. 4 show consistent performances across the 4 subjects at each value of n (with MEG and with iEEG). Finally, the superimposed gray curves (which depict the significance boundary given by the binomial formula as a function of sample size) nicely follow the trend of the % correct classification rate, and also illustrate cases of tolerated false positives for a given alpha.

4. Discussion

The current study has two primary take-home messages. The first is emphasizing the importance of watching out for a potential caveat that may arise when using departure from the theoretical chance-level as evidence for meaningful decoding. By launching various classifiers on normally distributed random data (Gaussian white noise), we demonstrate and quantify to which extent small samples lead to decoding accuracies that overshoot the chance-level merely by chance. This observation follows from the fact that

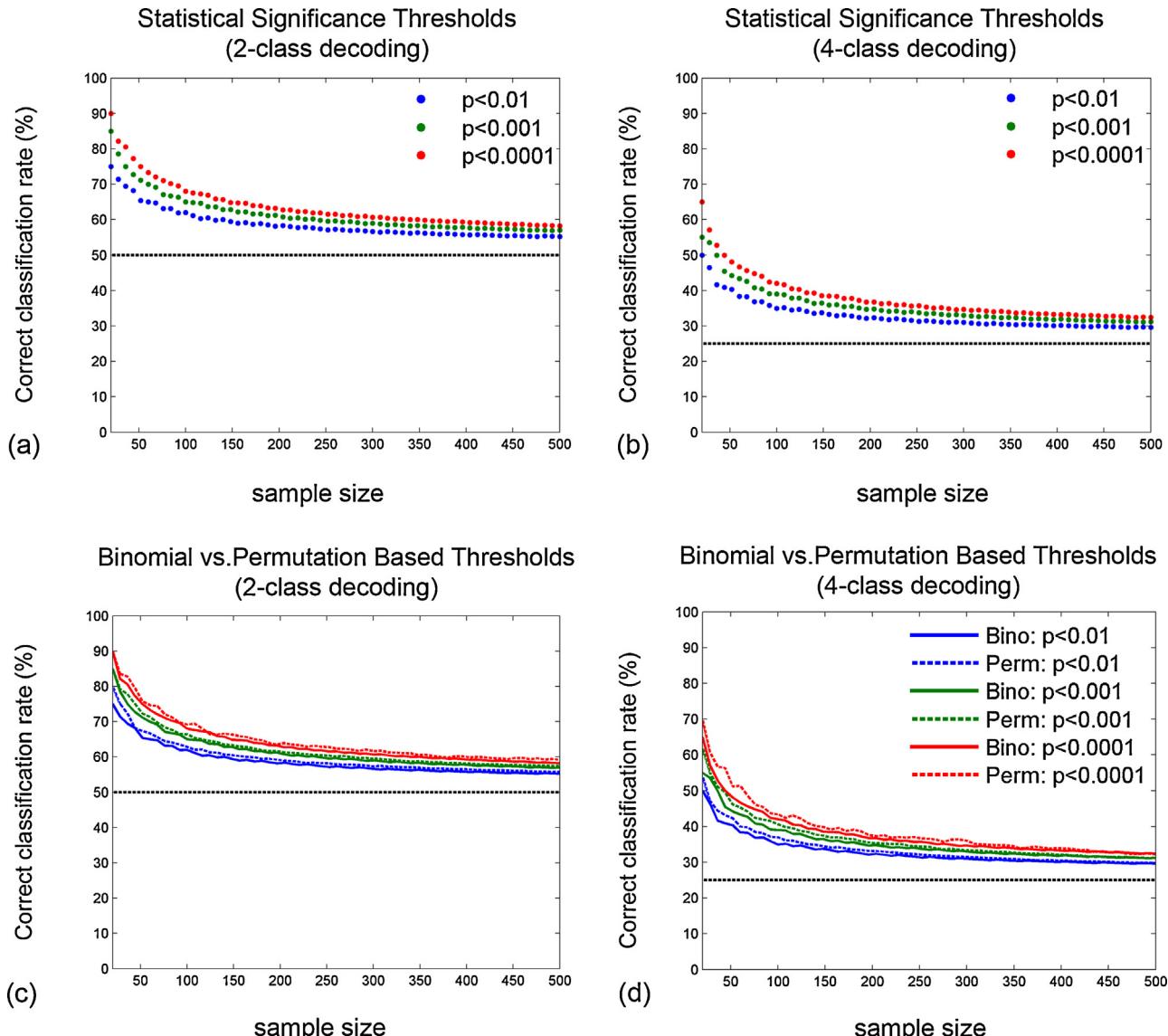


Fig. 3. Estimation of the statistical significance thresholds for 2- and 4-class classification as a function of sample size (assuming prediction errors are binomially distributed). Panels (a) and (b) show the evolution of the minimal statistically significant decoding rate as a function of sample size (respectively for 2 and 4 classes) using the binomial cumulative distribution (see Section 2.2). The plots were derived for significance levels 10^{-2} , 10^{-3} and 10^{-4} . As an example: panel (a) indicates that given a total of 100 data samples, a 2-class decoding result can only be considered statistically significant (at $p < 0.001$) if it exceeds 65%. This minimal value drops to 59% for 300 samples, but rises up to 75% if only 40 data points are available (See Table 1). Panels (c) and (d) show the same statistically significant decoding rate as a function of sample size (respectively for 2 and 4 classes) but now using both the binomial cumulative distribution (continuous lines) and the data-driven permutation-based approach (dashed lines) applied to the simulated random data (see Section 2.3 for details).

small samples are a bad approximation of true randomness and that as a result, the level $100/c$ (where c is the number of classes) is a purely theoretical chance-level that only holds for infinite sample sizes and that is particularly violated for small sample sizes. This basic fact is often overlooked in the neuronal decoding literature, where it is sometimes tempting to interpret for instance a 65% decoding accuracy in a 2-class classification as reflecting true neuronal decoding, without taking sample size into account. We have shown here that such levels of classification can be achieved with small samples of randomly generated data. This issue is not problematic for huge data samples, however, in data obtained from brain signal recordings in humans (such EEG or MEG), sample size can often be small. The effect of small samples on the reliability of probabilistic thresholds is therefore of particular importance in neural decoding and brain-computer interface studies. This effect is possibly even more critical when attempting to decode neuronal

signals acquired using intracranial recordings (electrocorticography or stereoacoustic-EEG) and in clinical BCI applications where even less data samples might be available.

Furthermore, our exploration of the effects of classifier type (LDA, NB and SVM), cross-validation partition (number of folds) and cross-validation repetition number (up to 20), indicates that none of these parameters has a noticeable impact on the variance of the classification when applied to random data. The small sample size problem cannot be circumvented by tweaking these parameters and even for larger sample sizes of white noise any reduction in classification variance remains negligible. Note that the explored parameters and classifier comparisons performed here only address the variance and bias of the techniques when applied to normally distributed random data, reviews and comparisons of classifiers can be found elsewhere (e.g. [Lotte et al., 2007](#)).

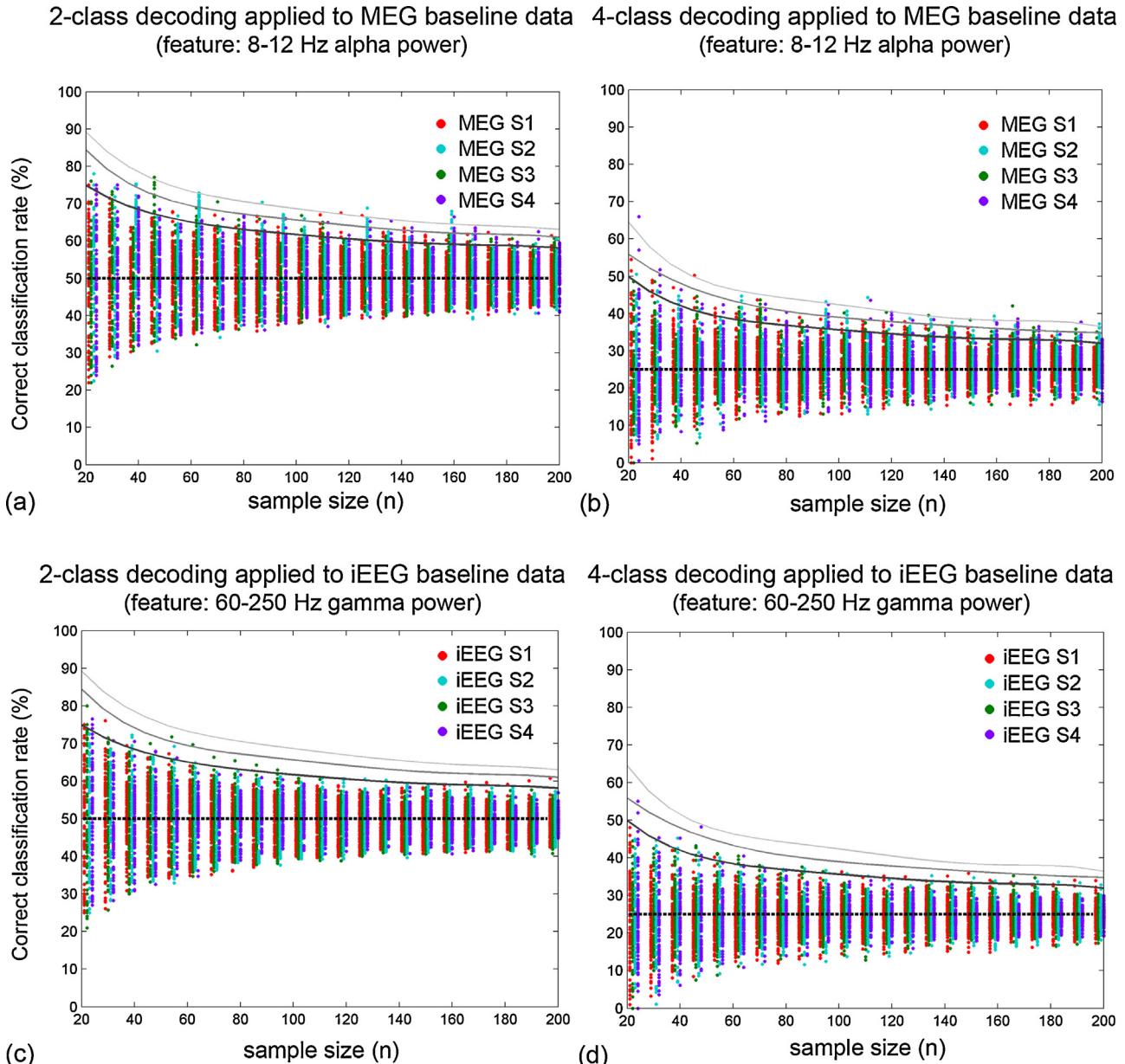


Fig. 4. Experimental assessment of chance-level classification accuracy in baseline (pre-stimulus) MEG and intracranial EEG data. (a) Two-class LDA classification rate (%) of MEG baseline data (alpha power features) as a function of sample size (illustrative data in 4 participants MEG S1–S4). The gray lines show the evolution of statistical significance boundaries computed with the binomial formula. Points lying above the gray lines thus represent false positives (type I errors) (b) Same as panel (a) but using 4-class classification, (c and d) Same as (a and b) but using baseline data (gamma power features) from intracranial EEG recordings (illustrative data in 4 epilepsy patients iEEG S1–S4).

Ten-fold cross-validation, which we used here as default, has been shown to be a reasonable choice providing low variance (Kohavi, 1995; Martin and Hirschberg, 1996a). Nevertheless, we also explored 5-fold and LOO cross-validation, alongside repetition number (Fig. 2). We found that none of these parameters could help reduce the cross-validation variance for low sample sizes. What is more, leave-one-out cross-validation showed even higher variability (in particular when using LDA), which is in agreement with previous reports suggesting that, despite its low bias, its high variance leads to unreliable estimates (Efron, 1983). Note that estimating the variance of cross-validation results across its k folds is generally problematic. Naive estimators that do not take into account error correlations due to the overlap between training and test sets (across the cross-validation folds) can severely underestimate variance (Bengio and Grandvalet, 2004). The cross-validation

variances reported here were computed across the 100 independent data sets of Gaussian white noise.

The second take-home message from our study is an important reminder that one way to overcome this limitation is to seek statistically significant thresholds on decoding accuracy, rather than relying solely on the theoretical chance-level to claim successful decoding. This has been demonstrated here using a sample-size dependent threshold computation derived from the binomial cumulative distribution function. The underlying assumption that the number of errors is binomially distributed is commonly used in statistical learning (Kohavi, 1995; Breiman et al., 1984) but the statistical bounds it provides are unfortunately rarely exploited in brain signal decoding studies (e.g. Quiroga and Panzeri, 2009; Müller-Putz et al., 2008; Ang et al., 2010; Arvaneh et al., 2013; Demandt et al., 2012; Galan et al., 2014; Lampe et al., 2014; Pistohl

et al., 2012; Waldert et al., 2008, 2007, 2012). Kohavi (1995) provides a proof that k -fold cross-validation is binomial if the classifier induction method is stable under cross-validation. Note also that the validity of the assumption that prediction errors are binomially distributed has also been demonstrated for the specific case of 10-fold cross-validation with small samples (Martin and Hirschberg, 1996b). The latter study also emphasizes that the textbook formula based on the normal approximation to the binomial is not a good approximation to the confidence interval of an error rate estimate for small samples.

In addition to the binomial formula, we have also demonstrated the use of permutation tests as an alternative method to derive statistical significance boundaries for classifier performance as a function of sample size (Fig. 3c and d). Permutation tests provide a reliable and data-driven approach to the problem and has been proposed and used in numerous previous studies (e.g. Golland and Fischl, 2003; Ojala and Garriga, 2010; Meyers and Kreiman, 2011). Our analysis shows how, via multiple random shuffling of the data (or class labels), permutation tests can provide an estimate of sample-size dependent chance-level decoding accuracy. These empirical chance levels need to be exceeded in order to assert significance of a classification for a given rate of tolerated false positives. When applied to random noise signals, we found that the significance boundaries derived using permutations are reasonably close to those obtained using the binomial formula. Deciding which of the two approaches is more convenient when applied to real brain signals will likely depend on the data at hand. Permutation tests do not make any assumptions about the distribution of the data and provide a data-driven approach; however they also come with the burden of high computational cost, which dramatically increases with sample size, and with the level of statistical significance required.

Meyers and Kreiman (2011) note that deriving significance thresholds via the binomial formula as discussed here and elsewhere (e.g. Quiroga and Panzeri, 2009) comes with theoretical limitations that one should keep in mind, in particular, when combined with cross-validation; its application to mean performance over all folds violates the assumption of data point independence and leads to p -values that are too small. From a practical perspective, the impact of this theoretical limitation is likely to depend on the data at hand and on the selected cross-validation parameters. Simulations show that cross-validation parameters (number of cross-validation folds and repetitions) have an impact on the cumulative distribution function of classification accuracies (e.g. Noirhomme et al., 2014). As a result, cross-validation parameters, alongside classifier type and feature space, collectively lead to deviations from a binomial cumulative distribution. These deviations can be significant for small sample sizes (e.g. $N < 100$), which would advocate against using the binomial formula for statistical assessments under such circumstances (Noirhomme et al., 2014). In contrast, permutation tests being inherently data-driven, do take cross-validation parameters into account. As far as the Gaussian white noise simulated in the current study is concerned, permutation tests and the binomial formula appear to provide reasonably similar significance boundaries. Comparing the output of the binomial formula and (the more time consuming) permutation test, on at least a portion of the data, could be a pragmatic way to decide on whether the former provides a suitable and fast approximation of the latter.

Moreover, our analysis of decoding accuracy using real brain signals (with random labeling) is in line with our simulation results. This is a reassuring finding, as the latter were based on zero-mean Gaussian white noise while the former were based on power features (alpha and gamma-bands) derived from real brain data. The baseline-period MEG and iEEG data suggest that the binomial formula provides a reasonable estimation of chance-level

decoding in these data sets. As a general rule, whenever possible, it is highly recommended to use baseline data as a recording in which no task-dependent encoding occurs and thus within chance-level decoding is expected. Comparisons with pre-stimulus (baseline) decoding performances should be used as an additional sanity check whenever such data is available (Meyers and Kreiman, 2011).

An alternative framework that can be applied to measure and compare classifier performance, is the use of receiver operating characteristic (ROC) analysis and in particular the area under the ROC curve (AUC) (Ling et al., 2003; Huang and Ling, 2005; Bradley, 1997). It has also been shown that calculating the probability density function (pdf) for each point on a ROC curve for any given sample size can be used to produce confidence intervals for ROC curves that are valid for small sample sizes (Tilbury et al., 2000). Adaptations of this method might be particularly suited to assessing classifier performance in BCI research (Hamadicharef, 2010). Other solutions that have been proposed to tackle the small sample size problem include frameworks that combine cross-validation with bootstrapping (e.g. Fu et al., 2005) and the use of class-dependent PCA in conjunction with linear discriminant feature extraction (Das and Nenadic, 2009). It is noteworthy that a few authors have even suggested that classification studies should be based primarily on effect size estimation with confidence intervals, rather than on significance tests and p -values (Berrar and Lozano, 2013).

In summary, the notion of statistical significance for decoding rates (or prediction error) and the small sample size problem have been tackled in the field of statistical learning for a long time (e.g. Raudys and Jain, 1991). However, these notions have not been sufficiently acknowledged in the relatively recent surge in application of machine learning methods in neuroscience. In the worse cases, this can unfortunately lead to erroneous interpretation of decoding results. Beyond its importance for brain-computer interface research specifically, signal classification is also increasingly used in neuroscience with the broader aim of elucidating the functional role of specific neuronal features (i.e. unraveling neuronal encoding by investigating single-trial neuronal decoding). Incidentally, this is where researchers are likely to be tempted to consider low (but above chance-level) decoding accuracies (e.g. 68% in a two-class classification) as being relevant. The use of confidence intervals and robust estimation of statistical significance is of particular importance in such studies, and even more so in cases with low trial numbers (e.g. below 150 observations). Machine learning and cross-validation accuracy in multi-class decoding may therefore not be thought of as a less-strict approach that can circumvent traditional rigorous statistical comparisons of data from multiple experimental conditions. Finally, whether signal classification is used in a BCI context *stricto sensu* or within a framework to conduct basic neuroscience analysis, we highly recommend systematically reporting the decoding accuracy as well as its statistical significance. We hope that the simulation results, statistical approaches and practical recommendations discussed here will be helpful in illustrating the problem and providing ways of tackling it.

Acknowledgements

Etienne Combrisson is currently supported by a Ph.D. Scholarship awarded by the Ecole Doctorale Inter-Disciplinaire Sciences-Santé (EDISS), Lyon, France. This work was partly performed within the framework of the LABEX CORTEX (ANR-11-LABX-0042) of Université de Lyon, within the program ANR-11-IDEX-0007. This research was undertaken, in part, thanks to funding from the Canada Research Chairs program. The authors are grateful for the collaboration with the research and clinical staff of the Magnetoencephalography (MEG) center at the Pitié-Salpêtrière Hospital

in Paris and the University Hospital in Grenoble (Dr. Philippe Kahane).

Appendix A.

A. Software availability: The MATLAB scripts and functions that were developed and used in this study have been made available online for the community. The provided code can be used to generate, label and classify random data. It also provides routines to compute and plot, as a function of sample size, (a) analytical chance levels via the binomial formula as well as (b) empirical chance levels via permutation tests. We hope that this set of tools will help students and researchers replicate and extend our analyses. The code can be downloaded from Mathwork's File Exchange platform at the following URL: <http://www.mathworks.fr/matlabcentral/fileexchange/48274-random-data-classification>

References

- Ahn M, Ahn S, Hong JH, Cho H, Kim K, Kim BS, et al. Gamma band activity associated with BCI performance: simultaneous MEG/EEG study. *Front Hum Neurosci* 2013;7. Available from: <http://journal.frontiersin.org/journal/10.3389/fnhum.2013.00848/full>.
- Aloise F, Schettini F, Aricò P, Salinari S, Babiloni F, Cincotti F. A comparison of classification techniques for a gaze-independent P300-based brain-computer interface. *J Neural Eng* 2012;9(4):045012.
- Ang KK, Guan C, Sui Geok Chua K, Ang BT, Kuah C, Wang C, et al. Clinical study of neurorehabilitation in stroke using EEG-based motor imagery brain-computer interface with robotic feedback. *IEEE Trans Rehabil Eng* 2010;5549–52 [cited 2014 Jul 4]. Available from: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5626782.
- Arvaneh M, Guan C, Ang KK, Quek C. EEG data space adaptation to reduce intersession nonstationarity in brain-computer interface. *Neural Comput* 2013;25(May (8)):2146–71.
- Babiloni F, Cincotti F, Lazzarini L, Millán J, Mourão J, Varsta M, et al. Linear classification of low-resolution EEG patterns produced by imagined hand movements. *IEEE Trans Rehabil Eng* 2000;8(June (2)):186–8.
- Ball T, Schulze-Bonhage A, Aertsen A, Mehring C. Differential representation of arm movement direction in relation to cortical anatomy and function. *J Neural Eng* 2009;6(1):016006.
- Bengio Y, Grandvalet Y. No unbiased estimator of the variance of k-fold cross-validation. *J Mach Learn Res* 2004;5:1089–105.
- Berrar D, Lozano JA. Significance tests or confidence intervals: which are preferable for the comparison of classifiers? *J Exp Theor Artif Intell* 2013;25(June (2)):189–206.
- Besserve M, Jerbi K, Laurent F, Baillet S, Martinerie J, Garnero L, et al. Classification methods for ongoing EEG and MEG signals. *Biol Res* 2007;40(4):415–37.
- Bleichner MG, Jansma JM, Sellmeijer J, Raemaekers M, Ramsey NF. Give me a sign: decoding complex coordinated hand movements using high-field fMRI. *Brain Topogr* 2014;27(March (2)):248–57.
- Bode S, Haynes J-D. Decoding sequential stages of task preparation in the human brain. *Neuroimage* 2009;45(April (2)):606–13.
- Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. *ACM* 1992;144–52 [cité 25.07.14].
- Bradley AP. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognit* 1997;30(july (7)):1145–59.
- Breiman L, Friedman JH, Olshen R, Stone CJ. Classification and regression trees; 1984.
- Burges CJ. A tutorial on support vector machines for pattern recognition. *Data Min Knowl Discov* 1998;2(2):121–67.
- Cortes C, Vapnik V. Support-vector networks. *Mach Learn* 1995;20(September (3)):273–97.
- Das K, Nenadic Z. An efficient discriminant-based solution for small sample size problem. *Pattern Recognit* 2009;42(May (5)):857–66.
- Demant E, Mehring C, Vogt K, Schulze-Bonhage A, Aertsen A, Ball T. Reaching movement onset- and end-related characteristics of eeg spectral power modulations. *Front Neurosci* 2012;6. <http://www.frontiersin.org/journal/10.3389/fnins.2012.00065/full>.
- Deriz J, Iljina O, Schulze-Bonhage A, Aertsen A, Ball T. "Doctor" or "darling"? Decoding the communication partner from ECoG of the anterior temporal lobe during non-experimental, real-life social interaction. *Front Hum Neurosci* 2012;6. <http://www.frontiersin.org/journal/10.3389/fnhum.2012.00251/full>.
- Efron B. Estimating the error rate of a prediction rule: improvement on cross-validation. *J Am Stat Assoc* 1983;78(June (382)):316–31.
- Felton EA, Wilson JA, Williams JC, Garell PC. Electrocorticographically controlled brain-computer interfaces using motor and sensory imagery in patients with temporary subdural electrode implants. Report of four cases. *J Neurosurg* 2007;106(March (3)):495–500.
- Fisher RA. The use of multiple measurements in taxonomic problems. *Ann Eugen* 1936;7(September (2)):179–88.
- Fu WJ, Carroll RJ, Wang S. Estimating misclassification error with small samples via bootstrap cross-validation. *Bioinformatics* 2005;21(May (9)):1979–86.
- Fukunaga K. Introduction to statistical pattern recognition. 2nd ed. Boston: Academic Press; 1990.
- Galan F, Baker MR, Alter K, Baker SN. Missing kinaesthesia challenges precise naturalistic cortical prosthetic control. May. Report no.: 004861; 2014, <http://biorxiv.org/lookup/doi/10.1101/004861>.
- Golland P, Fischl B. Permutation tests for classification: towards statistical significance in image-based studies. In: Proc. IPMI: international conference on information processing and medical imaging, LNCS, vol. 2732; 2003. p. 330–41.
- Good PI. Permutation tests: a practical guide to resampling methods for testing hypotheses. 2nd ed. New York: Springer; 2000.
- Hamadicharef B. AUC confidence bounds for performance evaluation of Brain-Computer Interface. In: IEEE 3rd International (Volume:5) Conference on Biomedical Engineering and Informatics (BMEI); 2010. p. 1988–91. Available from: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5639671.
- Hamamé CM, Vidal JR, Ossandón T, Jerbi K, Dalal SS, Minotti L, et al. Reading the mind's eye: online detection of visuo-spatial working memory and visual imagery in the inferior temporal lobe. *NeuroImage* 2012;59(January (1)):872–9.
- Haynes J-D, Sakai K, Rees G, Gilbert S, Frith C, Passingham RE. Reading hidden intentions in the human brain. *Curr Biol* 2007;17(February (4)):323–8.
- Hill NJ, Lal TN, Schröder M, Hinterberger T, Wilhelm B, Nijboer F, et al. Classifying EEG and ECoG signals without subject training for fast BCI implementation: comparison of nonparalyzed and completely paralyzed subjects. *IEEE Trans Neural Syst Rehabil Eng* 2006;14(June (2)):183–6.
- Hosseini SMH, Mano Y, Rostami M, Takahashi M, Sugiura M, Kawashima R. Decoding what one likes or dislikes from single-trial fNIRS measurements. *NeuroReport* 2011;22(April (6)):269–73.
- Huang J, Ling CX. Using AUC and accuracy in evaluating learning algorithms. *IEEE Trans Knowl Data Eng* 2005;17(3):299–310.
- Jerbi K, Bertrand O, Schoendorff B, Hoffmann D, Minotti L, Kahane P, et al. Online detection of gamma oscillations in ongoing intracerebral recordings: From functional mapping to brain computer interfaces. In: Noninvasive Funct Source Imaging Brain Heart Int Conf Funct Biomed Imaging 2007 NFSI-ICFB 2007 Jt Meet 6th Int Symp On. IEEE; 2007a. p. 330–3, http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4387767.
- Jerbi K, Lachaux J-P, Karim N, Pantazis D, Leahy RM, Garnero L, et al. Coherent neural representation of hand speed in humans revealed by MEG imaging. *Proc Natl Acad Sci* 2007b;104(18):7676–81.
- Jerbi K, Freyermuth S, Minotti L, Kahane P, Berthoz A, Lachaux J. Watching brain TV and playing brain ball International review of neurobiology. In: Brain machine interfaces for space applications: enhancing astronaut capabilities. San Diego: Elsevier Academic Press; 2009a. p. 159–68 [chapter 12]. <http://linkinghub.elsevier.com/retrieve/pii/S0074774209860121>.
- Jerbi K, Ossandón T, Hamamé CM, Senova S, Dalal SS, Jung J, et al. Task-related gamma-band dynamics from an intracerebral perspective: review and implications for surface EEG and MEG. *Hum Brain Mapp* 2009b;30(June (6)):1758–71.
- Jerbi K, Vidal JR, Mattout J, Maby E, Lecaingard F, Ossandón T, et al. Inferring hand movement kinematics from MEG, EEG and intracranial EEG: from brain-machine interfaces to motor rehabilitation. *IRBM* 2011;32(February (1)):8–18.
- Jerbi K, Combrisson E, Dalal SS, Vidal JR, Hamamé CM, Bertrand O, et al. Decoding cognitive states and motor intentions from intracranial EEG: how promising is high-frequency brain activity for brain-machine interfaces? *Epilepsy Behav* 2013;28(2):283–302.
- Kayikcioglu T, Aydemir O. A polynomial fitting and k-NN based approach for improving classification of motor imagery BCI data. *Pattern Recognit Lett* 2010;31(August (11)):1207–15.
- Kellis S, Miller K, Thomson K, Brown R, House P, Greger B. Decoding spoken words using local field potentials recorded from the cortical surface. *J Neural Eng* 2010;7(October (5)):056007.
- Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection; 1995. p. 1137–45, <http://frostiebek.free.fr/docs/Machine%20Learning/validation-1.pdf>.
- Korcyn AD, Schachter SC, Brodie MJ, Dalal SS, Engel J, Guekht A, et al. Epilepsy, cognition, and neuropsychiatry (Epilepsy, Brain, and Mind, part 2). *Epilepsy Behav* 2013;28(2):283–302.
- Krusienski DJ, Wolpaw JR. Brain-computer interface research at the wadsworth center developments in noninvasive communication and control. *Int Rev Neurobiol* 2009;86:147–57.
- Lachaux J-P, Jerbi K, Bertrand O, Minotti L, Hoffmann D, Schoendorff B, et al. A Blueprint for Real-Time Functional Mapping via Human Intracranial Recordings. *PLOS ONE* 2007a;2(October (10)):e1094.
- Lachaux J-P, Jerbi K, Bertrand O, Minotti L, Hoffmann D, Schoendorff B, et al. BrainTV: a novel approach for online mapping of human brain functions. *Biol Res* 2007b;40(January (4)):401–13.
- Lampe T, Fiederer LDJ, Voelker M, Knorr A, Riedmiller M, Ball T. A brain-computer interface for high-level remote control of an autonomous, reinforcement-learning-based robotic system for reaching and grasping. In: Proceedings of the 19th international conference on intelligent user interfaces. New York, NY, USA: ACM; 2014. p. 83–8, <http://dx.doi.org/10.1145/2557500.2557533>.
- Leuthardt EC, Schalk G, Wolpaw JR, Ojemann JG, Moran DW. A brain-computer interface using electrocorticographic signals in humans. *J Neural Eng* 2004;1(June (2)):63.
- Leuthardt EC, Miller KJ, Schalk G, Rao RPN, Ojemann JG. Electrocorticography-based brain computer interface—the Seattle experience. *IEEE Trans Neural Syst Rehabil Eng* 2006;14(June (2)):194–8.

- Ling CX, Huang J, Zhang H. AUC: a statistically consistent and more discriminative measure than accuracy; 2003. p. 519–24, <http://arion.csd.uwo.ca/faculty/ling/papers/ijcai03.pdf>.
- Lotte F, Congedo M, Lécuyer A, Lamarche F, Arnaldi B. A review of classification algorithms for EEG-based brain-computer interfaces. *J Neural Eng [Internet]* 2007 [cited 2012 Oct 3];4. Available from: <http://hal.archives-ouvertes.fr/docs/00/13/49/50/PDF/article.pdf>.
- Martin JK, Hirschberg DS. Small sample statistics for classification error rates I: error rate measurements. Irvine: Information and Computer Science, University of California; 1996a, <http://www.ics.uci.edu/~dan/pubs/TR96-21.pdf>.
- Martin JK, Hirschberg DS. Small sample statistics for classification error rates II: confidence intervals and significance tests [Internet]. Information and Computer Science. Irvine: University of California; 1996b, Disponible sur: <http://www.ics.uci.edu/~dan/pubs/TR96-22.pdf>.
- Mehring C, Nawrot MP, de Oliveira SC, Vaadia E, Schulze-Bonhage A, Aertsen A, et al. Comparing information about arm movement direction in single channels of local and epicortical field potentials from monkey and human motor cortex. *J Physiol – Paris* 2004;98(July (4–6)):498–506.
- Meyers EM, Kreiman G. Tutorial on pattern classification in cell recordings. In: Kriegeskorte N, Kreiman G, editors. Understanding visual population codes. Boston: MIT Press; 2011.
- Momennejad I, Haynes J-D. Human anterior prefrontal cortex encodes the “what” and “when” of future intentions. *Neuroimage* 2012;61(May (1)):139–48.
- Morash V, Bai O, Furlani S, Lin P, Hallett M. Classifying EEG signals preceding right hand, left hand, tongue, and right foot movements and motor imaginations. *Clin Neurophysiol* 2008;119(November (11)):2570–8.
- Müller-Putz GR, Scherer R, Brunner C, Leeb R, Pfurtscheller G. Better than random? A closer look on BCI results. *Int J Bioelectromagn* 2008;10(1):52–5.
- Neuper C, Scherer R, Reiner M, Pfurtscheller G. Imagery of motor actions: differential effects of kinesthetic and visual-motor mode of imagery in single-trial EEG. *Brain Res Cogn Brain Res* 2005;25(December (3)):668–77.
- Nichols TE, Holmes AP. Nonparametric permutation tests for functional neuroimaging: a primer with examples. *Hum Brain Mapp* 2002;15(1):1–25.
- Noirhomme Q, Lesenfants D, Gomez F, Soddu A, Schrouff J, Garraux G, et al. Biased binomial assessment of cross-validated estimation of classification accuracies illustrated in diagnosis predictions. *NeuroImage: Clin* 2014;4:687–94.
- Ojala M, Garriga GC. Permutation tests for studying classifier performance. *J Mach Learn Res* 2010;11(June):1833–63.
- Pistohl T, Schulze-Bonhage A, Aertsen A, Mehring C, Ball T. Decoding natural grasp types from human ECoG. *NeuroImage* 2012;59(January (1)):248–60.
- Quiroga RQ, Panzeri S. Extracting information from neuronal populations: information theory and decoding approaches. *Nat Rev Neurosci* 2009;10:173.
- Raudys SJ, Jain AK. Small sample size effects in statistical pattern recognition: recommendations for practitioners. *IEEE Trans Pattern Anal Mach Intell* 1991;13(March (3)):252–64.
- Schalk G, Miller KJ, Anderson NR, Wilson JA, Smyth MD, Ojemann JG, et al. Two-dimensional movement control using electrocorticographic signals in humans. *J Neural Eng* 2008;5(March (1)):75–84.
- Sitaram R, Lee S, Ruiz S, Rana M, Veit R, Birbaumer N. Real-time support vector classification and feedback of multiple emotional brain states. *NeuroImage* 2011;56(May (2)):753–65.
- Tilbury JB, Van Etten WJ, Garibaldi JM, Curnsw JSH, Ifeachor EC. Receiver operating characteristic analysis for intelligent medical systems-a new approach for finding confidence intervals. *IEEE Trans Biomed Eng* 2000;47(7):952–63.
- Toppi J, Risetti M, Quitadamo LR, Petti M, Bianchi L, Salinari S, et al. Investigating the effects of a sensorimotor rhythm-based BCI training on the cortical activity elicited by mental imagery. *J Neural Eng* 2014;11(June (3)):035010.
- Vapnik V. The nature of statistical learning theory. New York: Springer Science & Business Media; 1995.
- Waldert S, Braun C, Preissl H, Birbaumer N, Aertsen A, Mehring C. Decoding performance for hand movements: EEG vs. MEG. *IEEE 2007:5346–8*.
- Waldert S, Preissl H, Demandt E, Braun C, Birbaumer N, Aertsen A, et al. Hand movement direction decoded from MEG and EEG. *J Neurosci* 2008;28(January (4)):1000–8.
- Waldert S, Tüshaus L, Kaller CP, Aertsen A, Mehring C. fNIRS exhibits weak tuning to hand movement direction. *PLoS ONE* 2012;7(November (11)):e49266.
- Wang W, Sudre GP, Xu Y, Kass RE, Collinger JL, Degenhart AD, et al. Decoding and cortical source localization for intended movement direction with MEG. *J Neurophysiol* 2010;104(November (5)):2451–61.
- Wieland M, Pittore M. Performance Evaluation of Machine Learning Algorithms for Urban Pattern Recognition from Multi-spectral Satellite Images. *Remote Sens* 2014;6(March (4)):2912–39.

Troisième partie

Étude 2 : Encodage de l'intention et de l'exécution motrice

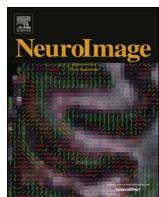
INTRODUCTION

Un changement d'état moteur engendre des modifications dans l'activité neuronale. Ces états sont caractérisés par des variations de synchronisation/désynchronisation dans l'activité oscillatoire. Certaines de ces variations sont déjà bien connues de la littérature (cf. 2.1.2). Toutefois, les changements de phase ou de couplage phase-amplitude (PAC) lorsque le sujet se trouve dans un état de repos, de préparation ou d'exécution motrice sont encore méconnus.

Pour affiner notre connaissance sur ce point, nous avons utilisé des données intracrâniennes chez six sujets souffrant d'épilepsie pharmacorésistantes ayant passé la tâche *Center-out* (cf. 6.2.1). Sur chaque site intracrânien, nous avons extrait des marqueurs de puissance, de phase et de PAC dans plusieurs bandes de fréquence et ce, durant le repos, la préparation et l'exécution motrice. Les modulations significatives de ces différents marqueurs sont ensuite présentées sur un cerveau MNI permettant ainsi de mettre en évidence les différences spectrales entre les états moteurs. Enfin, les outils de *machine learning* ont permis de dégager les sites et les marqueurs présentant des différences entre ces différents états (repos vs. préparation / repos vs. exécution / préparation vs. exécution / repos vs. préparation vs. exécution)

Sur les régions couvertes par ces six sujets, nous avons pu confirmer un ensemble de modulations de puissance, en particulier dans les régions motrices (augmentation de δ et haut- γ , baisse du μ et du β). Le PAC présente également des patterns intéressants puisque, dans le cortex moteur primaire, le couplage α/γ diminue drastiquement lors de l'exécution motrice pour être remplacé par un couplage δ/γ . Le décodage présente des similitudes (en particulier dans les régions motrices et pré-motrices) à travers les différentes conditions. Il n'est pas étonnant que la puissance fournit des décodages élevés, en revanche, le marqueur le plus discriminant a systématiquement été la phase de la composante basse fréquence [0.1, 1.5Hz]. Le PAC présente, dans une moindre mesure, des décodages significatifs indiquant qu'il porte lui aussi une information liée à la motricité.

L'utilisation des marqueurs d'amplitude, de phase et de couplage phase-amplitude permet d'améliorer notre compréhension du rôle respectif de ces différents *features*.



From intentions to actions: Neural oscillations encode motor processes through phase, amplitude and phase-amplitude coupling

Etienne Combrisson^{a,b,c,*}, Marcela Perrone-Bertolotti^{d,e}, Juan LP Soto^f, Golnoush Alamian^a, Philippe Kahane^g, Jean-Philippe Lachaux^c, Aymeric Guillot^b, Karim Jerbi^a

^a Psychology Department, University of Montreal, QC, Canada

^b University Claude Bernard Lyon 1, Inter-University Laboratory of Human Movement Biology, 27-29 Boulevard du 11 Novembre 1918, F-69622, Villeurbanne cedex, France

^c Lyon Neuroscience Research Center, Brain Dynamics and Cognition, INSERM U1028, UMR 5292, Lyon University, France

^d Univ. Grenoble Alpes, LPNC, F-38040 Grenoble, France

^e CNRS, LPNC UMR 5105, F-38040 Grenoble, France

^f Telecommunications and Control Engineering Department, University of Sao Paulo, Sao Paulo, Brazil

^g Neurology Dept and GIN U836 INSERM-UJF-CEA, Grenoble University Hospital, Grenoble, France



ARTICLE INFO

Keywords:

Neural oscillations

Intracranial EEG

Motor intention

Motor planning

Phase

Phase-amplitude coupling

ABSTRACT

Goal-directed motor behavior is associated with changes in patterns of rhythmic neuronal activity across widely distributed brain areas. In particular, movement initiation and execution are mediated by patterns of synchronization and desynchronization that occur concurrently across distinct frequency bands and across multiple motor cortical areas. To date, motor-related local oscillatory modulations have been predominantly examined by quantifying increases or suppressions in spectral power. However, beyond signal power, spectral properties such as phase and phase-amplitude coupling (PAC) have also been shown to carry information with regards to the oscillatory dynamics underlying motor processes. Yet, the distinct functional roles of phase, amplitude and PAC across the planning and execution of goal-directed motor behavior remain largely elusive. Here, we address this question with unprecedented resolution thanks to multi-site intracerebral EEG recordings in human subjects while they performed a delayed motor task. To compare the roles of phase, amplitude and PAC, we monitored intracranial brain signals from 748 sites across six medically intractable epilepsy patients at movement execution, and during the delay period where motor intention is present but execution is withheld. In particular, we used a machine-learning framework to identify the key contributions of various neuronal responses. We found a high degree of overlap between brain network patterns observed during planning and those present during execution. Prominent amplitude increases in the delta (2–4 Hz) and high gamma (60–200 Hz) bands were observed during both planning and execution. In contrast, motor alpha (8–13 Hz) and beta (13–30 Hz) power were suppressed during execution, but enhanced during the delay period. Interestingly, single-trial classification revealed that low-frequency phase information, rather than spectral power change, was the most discriminant feature in dissociating action from intention. Additionally, despite providing weaker decoding, PAC features led to statistically significant classification of motor states, particularly in anterior cingulate cortex and premotor brain areas. These results advance our understanding of the distinct and partly overlapping involvement of phase, amplitude and the coupling between them, in the neuronal mechanisms underlying motor intentions and executions.

1. Introduction

The simple motor act of stretching out your arm to grab a cup of coffee is mediated by a rich and complex chain of neuronal processes. What, in essence, may seem as the execution of a straightforward motor command is, in fact, carried out by a cascade of events ranging

from action selection and planning, to motor execution and monitoring. The neural mechanisms that mediate the transformation of a person's intentions into actions have been the subject of a thriving body of research for decades (Ariani et al., 2015; Brovelli et al., 2005; Desmurget and Sirigu, 2009; Jeannerod, 1994; Kalaska, 2009; Lau, 2004; Paus, 2001; Schwartz, 2016; Snyder et al., 1997). However,

* Correspondence to: Psychology Department University of Montreal, Pavillon Marie-Victorin 90, avenue Vincent d'Indy, Quebec, Canada.
E-mail address: e.combrisson@gmail.com (E. Combrisson).

Table 1

Patient data: handedness, age, gender, and description of epilepsy type, etiology, as determined by the clinical staff of the Grenoble Neurological Hospital, Grenoble, France. The lesions (if any were observed) were determined based on the T1 images. Recording sites with epileptogenic activity were excluded from the analyses.

	Handedness	Age	Gender	Epilepsy type	Etiology	EZ localization	Lesion
P1	R	19	F	Frontal	Secondary	Precentral gyrus (RH)	Dysplasia
P2	R	23	F	Frontal	Cryptogenic	Precentral gyrus (LH)	Absent
P3	R	18	F	Frontal	Cryptogenic	Fronto-basal (RH)	Absent
P4	R	18	F	Frontal	Idiopathic	Fronto-central (RH)	Absent
P5	R	31	F	Insula	Secondary	Operculum (RH)	Cavernoma
P6	R	24	F	Frontal	Secondary	Supra-sylvian posterior (LH)	Vascular sequelae

because the neuronal processes at play can be observed at various spatial scales, and with different recording techniques, parallel streams of research have given rise to a rich but fragmented understanding of the local and large-scale integrative electrophysiological mechanisms that are involved in motor control.

Both human and non-human primate research provides solid evidence that goal-directed motor behavior is associated with changes in the patterns of rhythmic neuronal activity across largely distributed brain areas (Schnitzler and Gross, 2005). Movement initiation and execution are mediated by patterns of synchronization and desynchronization that occur concurrently across distinct frequency bands and within multiple motor cortical areas (Cheyne et al., 2008; Jurkiewicz et al., 2006; Pfurtscheller et al., 2003; Saleh et al., 2010).

To date, motor-related local oscillatory modulations are by and large examined by quantifying increases or suppressions in spectral power (Cheyne et al., 2008; Jurkiewicz et al., 2006; Pfurtscheller et al., 2003; Saleh et al., 2010). However, beyond band-limited oscillatory power, other spectral properties, namely phase and phase-amplitude coupling (PAC), are also thought to play a key role in neuronal encoding and information processing. The involvement of phase information in neuronal encoding has been extensively investigated in numerous perceptual modalities and higher-order cognitive tasks (Drewes and VanRullen, 2011; Dugue et al., 2011; Jensen et al., 2014; Klimesch et al., 2008, 2007; Montemurro et al., 2008; Palva and Palva, 2007; Sauseng and Klimesch, 2008; Sherman et al., 2016; VanRullen et al., 2011). In comparison, the role of phase and phase-based measures mediating motor processes are still insufficiently studied. Interestingly, a few studies provide evidence for the involvement of low-frequency phase and amplitude in the neuronal encoding of movement features (Hammer et al., 2016, 2013; Jerbi et al., 2011, 2007; Milekovic et al., 2012; Miller et al., 2012; Waldert et al., 2009, 2008). Nevertheless, the spatial, temporal and spectral dynamics of putative phase coding in the chain of processes are still largely unresolved: starting from goal encoding, to motor planning and motor command execution.

Recent years have witnessed a surge in interest in the putative mechanistic function of PAC (Cohen et al., 2008; Hemtinne et al., 2013; Lee and Jeong, 2013; Newman et al., 2013; Voytek, 2010, Bahramisharif et al., 2013), and numerous measures of PAC have been proposed (Canolty, 2006; Nakhnikian et al., 2016; Tort et al., 2010; Voytek et al., 2013, Özkurt, 2012). Conceptually, PAC may provide a flexible framework for information processing by means of cross-frequency synchronization (Canolty and Knight, 2010; Hyafil et al., 2015; Maris et al., 2011; Staresina et al., 2015; van der Meij et al., 2012; Weaver et al., 2016). However, despite important advances (Hemtinne et al., 2013; Özkurt and Schnitzler, 2011; Soto and Jerbi, 2012; Yanagisawa et al., 2012), the precise role of PAC in mediating motor planning and execution is not yet fully resolved. Specifically, the distinct functional roles of phase, amplitude and PAC estimates during motor behavior remain generally ill-defined.

In the present paper, we compare the involvement of all three of these features using multi-site intracerebral depth electrode recordings from human subjects performing a delayed motor task. Using high spatial, spectral and temporal resolution, we monitored modulations of

neural activity, not only at movement execution but, also, during the delay time-window when motor intention is present but execution is withheld. In addition to standard statistical comparisons, we used a single-trial classification procedure (supervised learning) to identify the key contributions of three distinct oscillatory features (phase, amplitude and PAC) to the various motor-related processes along the chain of processes, from goal encoding to movement execution.

2. Material and methods

2.1. Participants

Six patients with medically intractable epilepsy participated in this study (6 females, mean age 22.17 ± 4.6). The patients were stereotactically implanted with multi-lead EEG depth electrodes at the Epilepsy Department of the Grenoble Neurological Hospital (Grenoble, France). In collaboration with the medical staff, and based on visual inspection, electrodes presenting pathological waveforms were discarded from the present study. All participants provided written informed consent, and the experimental procedures were approved by the Institutional Review Board, as well as by the National French Science Ethical Committee. Patient-specific clinical details can be found in Table 1.

2.2. Electrode implantation and stereotactic EEG recordings

Each patient was implanted with stereotactic electroencephalography (SEEG) electrodes. Each one of these had a diameter of 0.8 mm and, depending on the implanted structure, was composed of 10 to 15 contacts that were 2 mm wide and 1.5 mm apart (DIXI Medical Instrument). Intracranial EEG signals were recorded from a total of 748 intracerebral sites across all patients (126 sites in each participant, except for one patient who had 118 recording sites). At the time of acquisition, a white matter electrode was used as reference, and data was bandpass filtered from 0.1 to 200 Hz and sampled at 1024 Hz. Electrode locations were determined using the stereotactic implantation scheme and the Talairach and Tournoux proportional atlas (Talairach and Tournoux, 1993). The electrodes were localized in each individual subject in Talairach coordinates (based on post-implantation CT), and then transformed to standard MNI coordinate system according to standard routines and previously reported procedures (Bastin et al., 2016; Jerbi et al., 2010, 2009; Ossandon et al., 2011).

2.3. Delayed center-out motor task

After a rest period of 1000 ms, the participants were visually cued to prepare a movement towards a target in one of four possible directions: up, down, left or right (*Planning phase*). Next, after a 1500 ms delay period, a Go signal prompted the subjects to move the cursor towards the target (*Execution phase*). The Go signal consisted of a central cue changing from white to black. Fig. 1B shows the task design.

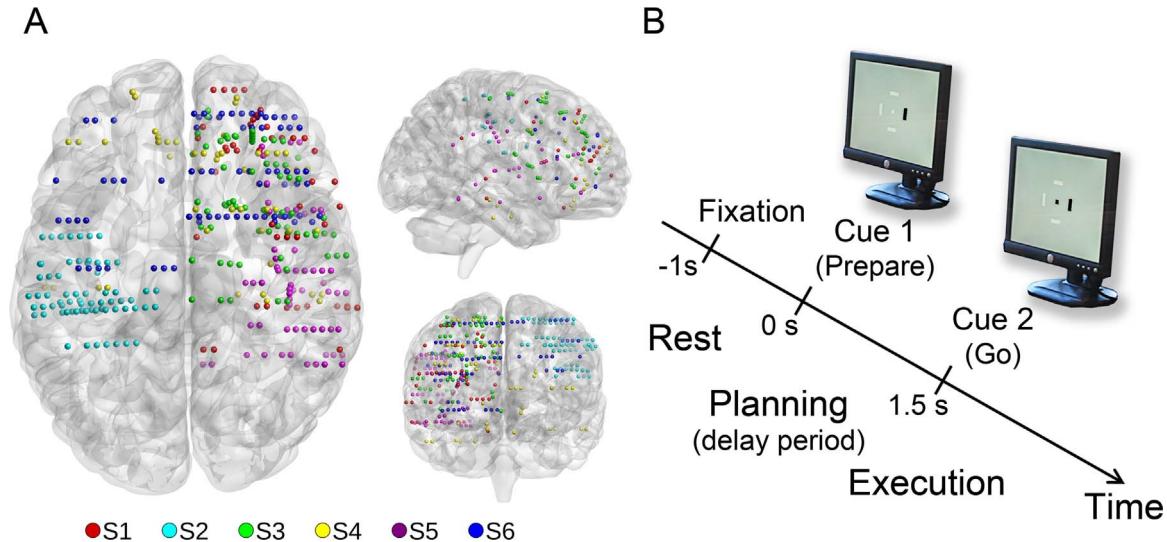


Fig. 1. Implantation visualization for the 6 subjects on a 3-D standard (MNI) brain, and the delayed center-out task definition. (A) Depth electrodes representation from top, right and frontal views. Each subject is associated to a different color. (B) Design of the delayed motor task, where the direction of the movement was instructed at Cue1, and the actual movement was carried out at Cue 2 ('Go signal'). The timeline consisted of three periods: rest, a delay period (motor planning/intention), and movement execution. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2.4. Data preprocessing

SEEG data preprocessing was conducted according to our routine procedures (Bastin et al., 2016; Jerbi et al., 2009). These included signal bipolarization, where each electrode site was re-referenced to its direct neighbor. Bipolar re-referencing can increase sensitivity and reduce artefacts by canceling out distant signals that are picked up by adjacent electrode contacts (e.g. mains power). The spatial resolution of bipolar SEEG of our electrodes were approximately 3 mm (Jerbi et al., 2009; Kahane et al., 2006; Lachaux et al., 2003). Next, using visual inspection and time-frequency explorations of the signal, we excluded electrodes containing pathological epileptic activity. In addition, electrodes located close to the extra-ocular eye muscles were systematically excluded to avoid eye-movement contaminations in our analyses. The pre-processing led to a total of 580 bipolar derivations across all participants (The distribution of all electrode sites across subjects and Brodmann Areas are provided in Supplementary Fig. S1).

2.5. Spectral analyses

We investigated phase, power, and phase-amplitude coupling in several standard frequency bands defined as follows: very low frequency component (VLFC) [0.1–1.5 Hz], delta (δ) [2–4 Hz], theta (θ) [5–7 Hz], alpha (α) [8–13 Hz], beta (β) [13–30 Hz], low-gamma (low γ) [30–60] and broadband gamma (high γ) [60–200 Hz]. The power features were computed in six bands (δ , θ , α , β , low γ and high γ), the phase features were extracted for 4 bands (VLFC, δ , θ and α), and the phase-amplitude coupling was extracted using three combinations (δ , θ and α for phase and high γ for amplitude). In total, 13 features were extracted for each SEEG bipolar derivation. Fig. 2 illustrates time and frequency-domain features extracted from an illustrative iEEG signal in premotor cortex (BA6).

2.5.1. Spectral power estimation

Band-specific power modulations were computed using the Hilbert transform. To this end, and to avoid phase shifting, we first filtered the data in the required band using a two-way zero-phase lag finite impulse response (FIR) Least-Squares filter implemented in the EEGLAB toolbox (Delorme and Makeig, 2004). This filter has been used in several studies where preserving phase is critical (e.g. Kramer et al., 2008; Cohen et al., 2008; Voytek et al., 2013; Yanagisawa et al., 2012).

Next, we computed the Hilbert transform of the filtered signal and calculated power by taking the square of the amplitude component (envelope). For the specific case of high-gamma power, we split the 60–200 Hz range into multiple, non-overlapping, 10 Hz bands. As in our previous studies (Hamamé et al., 2014; Perrone-Bertolotti et al., 2012; Vidal et al., 2014, 2012), broadband gamma power was obtained by taking the mean of all of the successive 10 Hz wide normalized bands (Jerbi et al., 2009; Ossandon et al., 2011; Bastin et al., 2016). Although comparable power estimations were obtained using Morlet wavelets (e.g. Tallon-Baudry et al., 1996), we chose to use the Hilbert-based power computation to have a homogeneous methodological framework across all three features, as the Hilbert method was used to assess phase and PAC features (see below).

2.5.1.1. Rationale for broadband gamma selection. Motor-related power modulations in the gamma-band have been observed with intracranial recordings across a wide range of frequencies extending up to 180 Hz or 200 Hz (e.g. Leuthardt et al., 2004; Crone et al., 2006; Miller et al., 2007). Yet, there is also ample evidence in the literature for prominent motor gamma oscillations within a narrower 60–90 Hz band, often with peaks around 75 or 80 Hz (e.g. Ball et al., 2008; Cheyne et al., 2008; Muthukumaraswamy, 2010, 2011; Jenkinson et al., 2013). Because many of these studies were conducted with EEG or MEG, one might argue that this discrepancy can in part be a reflection of differences between invasive and non-invasive gamma investigations. Electrophysiological recordings in non-human primates have demonstrated that changes in narrow-band gamma oscillations and in broadband gamma activity can occur simultaneously and that they are likely to reflect distinct phenomena (Ray and Maunsell, 2011). We chose to subdivide the gamma band into low gamma (30–60 Hz) and broadband gamma (60–200 Hz) in part because the latter band facilitated comparisons of our power results with our PAC findings, as well as with wide-band gamma power results in the intracranial EEG literature.

Note that for visualization purposes, cortical mapping of significant power modulations on the standard MNI brain (e.g. Figs. 3–5) was done after normalization with respect to a baseline window of 500 ms during pre-stimulus rest (from -750 ms to -250 ms). The power in each frequency band was normalized by computing the relative change compared to this baseline at the same frequency (i.e. subtracting and

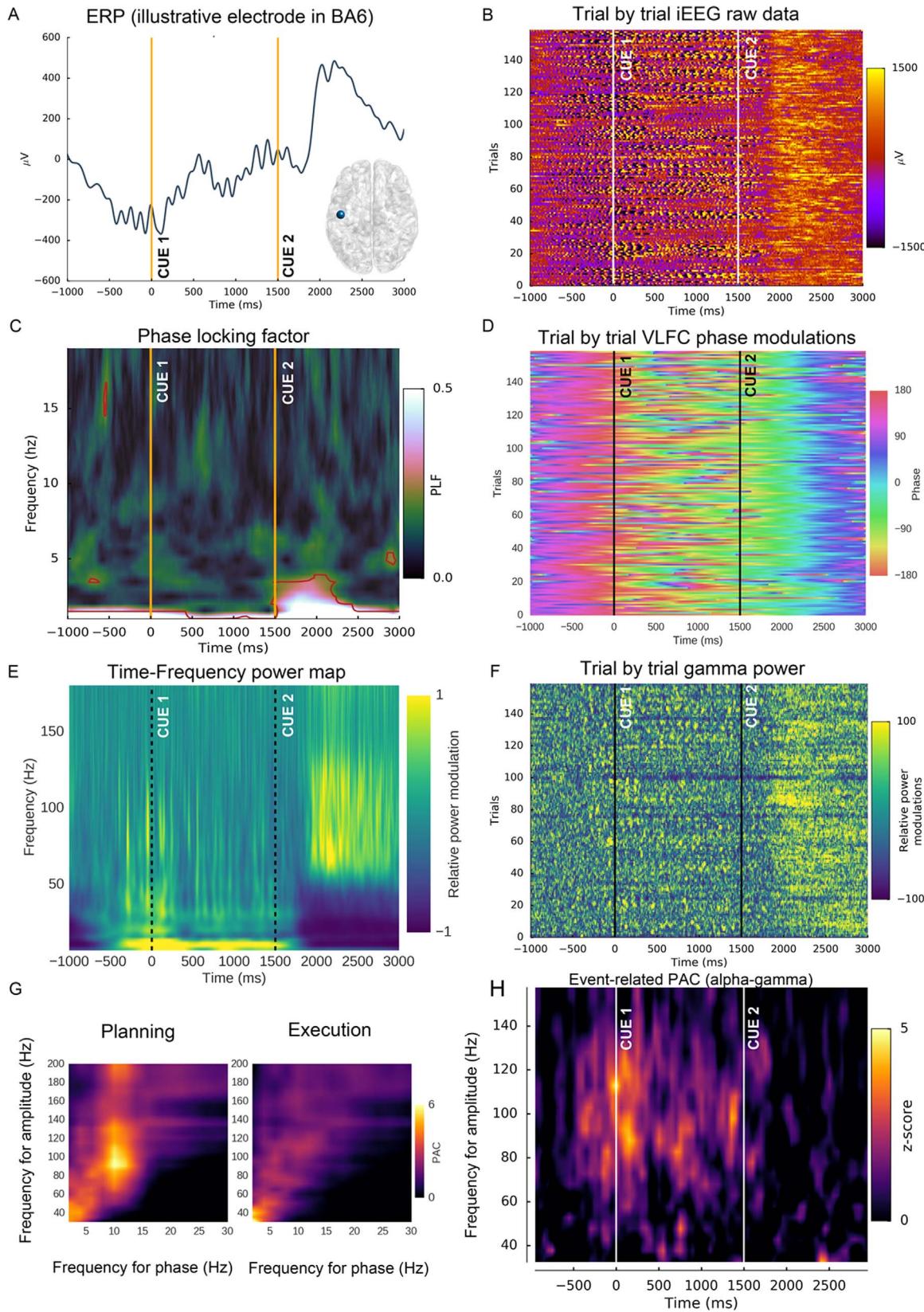


Fig. 2. Oscillatory feature extractions from intracranial EEG (iEEG) signals shown for an illustrative electrode in human premotor cortex (BA6). (A) Event-related potential (ERP) across the experiment timeline and anatomical location of the electrode. (B) Trial-by-trial raw iEEG signals. (C) Phase-locking factor indicating stimulus phase-locking (D) Single-trial phase shown for the VLFC (< 1.5 Hz) frequency range. (E) Time-Frequency representation. (F) Trial-by-trial broadband gamma power (60–200 Hz). (G) Phase-amplitude coupling maps during planning and execution, revealing prominent alpha-gamma coupling present during planning that was absent during execution. (H) Event-related PAC depicts, for the same electrode, the time course of alpha-gamma PAC.

dividing by the mean of the baseline).

2.5.1.2. Statistical evaluation of task-based power modulations. Significant power modulations were obtained by standard two-tailed permutation tests, where power values (across time and frequency) during the task were randomly permuted with the corresponding value during baseline. A total of 1000 permutations were performed yielding a null distribution for the relative power, providing a minimal *p*-value of 0.001.

2.5.1.3. Power-based classification. The power features used for classification were computed as mean power over 500 ms time windows during planning (250 to 750 ms), execution (2000 to 2500 ms) and (−750, −250 ms) during pre-stimulus rest, where t=0 ms corresponds to the onset of Cue 1. Note that when decoding “Exec vs Intention” (Fig. 6C), the single-trial power features used were first baseline normalized (with respect to the rest period) on a trial-by-trial basis. However, no such baseline-based feature normalization was possible when running classifications where the rest window was itself one of the classes (i.e. “Exec vs Rest”, “Intention vs Rest” and the 3-class decoding “Exec vs Intention vs Rest”, Fig. 6 panels A, B and D). While decoding results with and without baseline normalized features (for Execution vs Intention) were very similar, the normalized features appeared to provide more reliable decoding. This could be in part due to a beneficial effect of feature normalization on classifier performance.

2.5.2. Phase estimation

To extract phase features in a given frequency band, we first applied a bandpass filter to the bipolarized signals. Next, we extracted the angle of the complex time-series based on the Hilbert transform. This gave the instantaneous phase for all time points. Phase was extracted using the same type of filter as the one used to extract spectral power.

2.5.2.1. Statistical evaluation of task-based phase modulations. we used Rayleigh's test to compute significant phase modulations (Babiloni et al., 2002; Lakatos, 2005; Tallon-Baudry et al., 1996), using the circular statistics toolbox (Berens et al., 2009).

2.5.2.2. Phase-based classification. For decoding purposes, we considered instantaneous phase features at −500 ms for rest, 500 ms for planning and 2250 ms for execution. These time points were chosen because they correspond to the center of each time window used for power estimation.

2.5.3. Phase-Amplitude Coupling (PAC) estimation

Different methods for the estimation of PAC have been used in the literature (Jensen and Colgin, 2007; Canolty and Knight, 2010; Tort et al., 2010; Soto and Jerbi, 2012; Aru et al., 2015). In order to choose which method to apply here, we simulated synthetic PAC signals (e.g. Tort et al., 2010), and tested the behavior of several methods, namely Mean Vector Length (MVL) (Canolty, 2006), the Height-Ratio (HR) (Lakatos, 2005), Kullback-Leiber divergence (Tort et al., 2010), and normalized direct PAC (ndPAC) (Özkurt, 2012). We found that the tested PAC methods provided very comparable results. The slight differences arose from applying different normalization or surrogate procedures to the data. Based on this finding, we chose to use the MVL method combined with a normalization method, where the surrogate values were calculated by randomly swapping phase and amplitude across trials (cf. Tort et al., 2010). In short, the PAC estimation procedure that was applied can be summarized as follows. First, low-frequency phase and high-frequency amplitude signals were obtained

by filtering and applying Hilbert transformation in the frequencies of interest (Canolty, 2006). Being frequency-dependent, the optimal filter orders were adapted separately for the phase of the slower oscillations (3 cycles used), and for the amplitude of the faster oscillations (6 cycles), as proposed in previous studies (e.g. Bahramisharif et al., 2013). Next, surrogate data were generated by randomly swapping the phase trial data and amplitude trial data (i.e. randomly association of the high frequency amplitude envelope of a trial with the low-frequency phase time course of another trial). This procedure was repeated 1000 times. Finally, the normalized PAC value was obtained by normalizing the MVL by the surrogate data, yielding a z-score. This procedure is in line with previous PAC studies (Canolty, 2006). This form of statistical normalization using surrogate data provides a more robust estimate than non-normalized measures (Özkurt and Schnitzler, 2011).

To visualize the emergence of PAC, we used a combination of methods: (a) phase-alignment of single-trial time-frequency maps (Canolty, 2006; Hemptinne et al., 2013), (b) a comodulogram (e.g. Foster and Parvizi, 2012; Pittman-Polletta et al., 2014), and (c) event related phase-amplitude coupling (Voytek et al., 2013). When computing the comodulogram, we used the entire planning and execution periods (1500 ms for each) in order to maximize the per-trial number of oscillation cycles for the estimation of PAC.

2.5.3.1. Statistical evaluation of task-based PAC modulations. The statistical assessment of PAC was achieved by comparing the true PAC values to the null distribution of PAC values, computed using surrogate data (i.e. random shuffling across trials of phase and amplitude signals, Tort et al. 2010). A real PAC value, which was higher than the 999th highest PAC value obtained with surrogate data, was considered significant at *p* < 0.001.

2.5.3.2. PAC-based classification. For decoding, we used the normalized PAC values computed over the entire window of each condition; in other words [−1000, 0 ms] for Rest, [0, 1500 ms] for planning and [1500, 3000 ms] for execution. Note here, that by normalization, we refer to the transformation to z-scores based on surrogate data, not normalization with regards to pre-stimulus baseline levels.

2.6. Signal classification

We set out to explore the feasibility of using multi-site human Local Field Potential (LFP) data (580 bipolar electrode sites) to perform three types of motor-state classifications: (a) Execution vs Rest, (b) Intention vs Rest, and (c) Execution vs Intention. We compared the performance of several classification algorithms (Linear Discriminant Analysis (LDA), Naïve Bayes (NB), k-th Nearest Neighbor (KNN), Support Vector Machine (SVM) with linear and Radial Basis Function kernels and Random Forest). For single features classification, LDA, NB and SVM all provided similar results. We chose to use the LDA approach for its speed, which was of particular importance given the computationally-demanding permutation tests used to evaluate classifier performance.

2.7. Decoding accuracy and statistical evaluation of decoding performance

Classification performance was evaluated using standard stratified 10-fold cross-validation. First, the data set was pseudo-randomly split into 10, equally-sized, observations: 9 segments were used for training the classifier, and the last one as the test set. This procedure was repeated 10 times, such that every observation in the data was used exactly once for testing, and at least once for training, but never at the

same time. This strict separation of training and testing was critical to ensure the test data was naïve and did not violate basic classification principles (Lemm et al., 2011). The use of stratification seeks to ensure that the relative proportion of labels (or classes) in the whole data set is reasonably preserved within each of the segments after the split. The above procedure was repeated 10 times to reduce the effect of the random generation of folds, yielding a 10 times 10-fold cross-validation framework. Next, the performance of the achieved decoding was calculated using the decoding accuracy (DA) metric, which was computed as the mean correct classification across all folds. Although the use of theoretical chance-levels (e.g. DA=50% for binary classification) can provide some useful indication on classifier performance in the presence of a high number of observations, the use of statistics is mandatory in order to assess the significance of classification performance (Combrisson and Jerbi, 2015). To this end, we used a permutation testing framework where the cross-validation and DA calculations were recomputed after randomly shuffling the labels of the classes. For each site and for each type of feature (i.e. power, phase and phase-amplitude coupling), 1000 permutations were generated, thus allowing for statistical assessments with p values as low as 0.001 (Combrisson and Jerbi, 2015; Ojala and Garriga, 2010, Golland and Fischl, 2003; Meyers and Kreiman, 2012).

2.8. Data mapping to a 3-D standard cortical representation

To facilitate the interpretation of the results, all significant task-based feature modulations and decoding results were remapped from the intracranial electrode sites onto a standard cortical representation. To achieve this, all electrode coordinates were transformed from individual Talairach space to standard MNI space, and custom Matlab code was written to project the data from SEEG sites onto the cortical surface. In practice, data from the iEEG electrodes were assigned to the vertices on the MNI cortical mesh that fell within a fixed radial distance of 10 mm from each electrode. This cortical representation technique is methodologically consistent with methods used in previous iEEG studies (Bastin et al., 2016; Jerbi et al., 2009; Ossandon et al., 2012). In addition to generating brain-wide visualization of all significant features and decoding performances, this method was also used to display the cortical coverage provided by all the electrodes in this study (dark gray areas in Fig. 2A). Note that, by convention, the left hemisphere (LH) is presented on the left in all brain visualizations (Figs. 3 to 5). Furthermore, these 3D cortical maps of task-based feature modulations only show contralateral effects; if the patients were implanted in the left hemisphere (LH), we used data obtained using the right hand and vice versa. For the two patients who were implanted bilaterally we kept the right hemisphere (RH) electrode data when analyzing left hand movements, and the LH electrodes for analysis of right hand movements.

3. Results

The frequency domain analysis of the 580 intracranial bipolar recordings across all participants revealed that the delayed motor task was mediated by complex patterns of spectral modulations across widely distributed brain areas. Fig. 2 illustrates task-related modulations of amplitude, phase and phase-amplitude coupling measured by a recording site in premotor cortex (BA6) in one participant. The ERP and single-trial raw iEEG recordings for this electrode are shown in panels 2A and 2B, respectively. Panels 2C and 2D depict the phase-locking factor and a trial-by-trial phase representation across time. The illustrative time-frequency map (Fig. 2E) depicts a typical power modulation pattern, where movement execution was associated with a strong increase in broadband gamma power, and simultaneous power suppression in the alpha, beta and low-gamma frequency bands. The delay period (between Cue1 and Cue2) at this site showed a different pattern of power increases at slow frequencies in the delta to alpha

range. Because of the high signal-to-noise ratio (SNR) of the intracranial EEG recordings used, it is possible to estimate broadband gamma (60–200 Hz) power on a trial-by-trial basis (Fig. 2F). The PAC maps in Fig. 2G illustrate the changes in PAC values for the same recording site across motor planning and execution. A peak in PAC between alpha phase and high gamma amplitude was observed during the pre-movement delay period, this feature vanishes during execution, when the maximum PAC was observed between delta phase and low-gamma amplitude. The predominance of the alpha-gamma coupling in the delay period was also confirmed when the time-frequency (TF) maps were computed by realigning the single-trial TF maps to alpha phase in the vicinity of the cue (results not shown). Finally, time-resolved, event-related, PAC estimations (Fig. 2H) revealed the existence of an alpha-gamma coupling specific to the delay period. These oscillatory feature estimations were primarily presented as an illustration of the wide variety and high SNR of the features explored in this study, using averaging over trials and, most importantly, in trial-by-trial mode. In the following section, we discuss the global results obtained using data from all participants. First, we describe the task-based oscillatory modulations, and then we examine the results of the data-mining approach.

3.1. Task-based Spectral power modulations

The cortical mapping of power modulations during the delay period and motor execution reveal distinct patterns of increases and decreases across the various frequency bands (Fig. 3). As one would expect from previous, invasive, reports in humans (Crone et al., 1998; Crone et al., 2006; Babiloni et al., 2016; Bundy et al., 2016; Rektor et al., 2006), our results confirmed that motor execution is associated with prominent suppressions of alpha/mu and beta band powers, and with increases in high gamma power, primarily in motor and premotor cortices. Moreover, the power modulations shown in Fig. 3 are indicative of significant task-based modulations beyond these areas, extending to parietal, prefrontal and cingulate areas. When comparing the brain-wide significant power changes for both planning and execution, we first observed a large degree of overlap between the statistically significantly active areas in both conditions. Second, two distinct patterns appeared to emerge when we examined the direction of the effects across frequency bands: the alpha, beta and low-gamma bands showed a reversal of the effect, from a significant increase during planning to a significant suppression during execution. By contrast, the remaining bands (delta, theta and high gamma) displayed a consistent significant power increase during both the planning and execution periods, with a stronger effect at the execution of the movement. Of note, high-gamma increases were prominent over multiple frontal and prefrontal brain areas bilaterally, with strong peaks in motor and premotor cortices during execution. Most of these areas also showed a significant increase of power above baseline levels during the delay period, potentially related to movement goal-encoding and motor planning processes.

3.2. Task-based phase modulations

The patterns of statistically significant changes in phase revealed a predominant effect in the lower frequencies (Fig. 4). In particular, the VLFC range (i.e. < 1.5 Hz) showed consistent phase values in primary motor cortex (with a reversal in sign between planning and execution, at the given time instants). Interestingly, although less prominent, consistent phase effects were also found in the theta and alpha bands, predominantly in prefrontal areas. The significant delta phase angle over primary motor cortex during execution is in line with earlier reports on delta-range coherence between motor cortex and movement parameters (Jerbi et al., 2007).

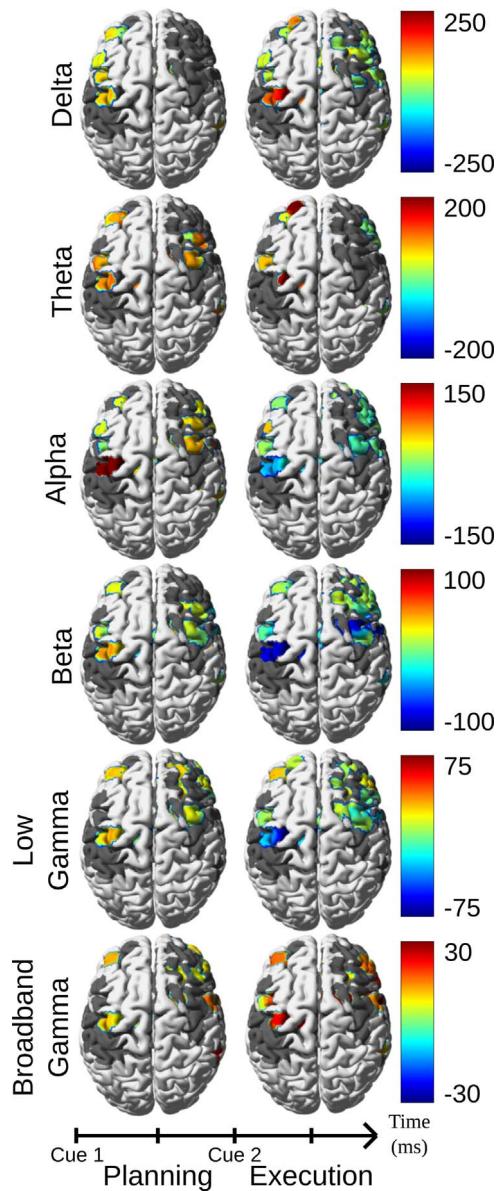


Fig. 3. Task-related modulations of spectral power during planning and execution of upper limb movements. The color bars represent percent relative changes (%) of power during planning (250–750 ms) and during execution (2000–2500 ms), with respect to baseline power during pre-stimulus rest (−750 to −250 ms). The power modulations are shown for delta (2–4 Hz), theta (5–7 Hz), alpha (8–13 Hz), beta (13–30 Hz), low-gamma (30–60 Hz) and broadband gamma (60–200 Hz). All modulations shown were statistically significant (permutation tests, $p < 0.05$, FDR-corrected). Dark gray areas represent cortical regions for which electrode coverage was available but, where the modulations did not reach statistical significance. By convention, the left hemisphere (LH) is presented on the left.

3.3. Task-based PAC modulations

Statistically significant levels of phase-amplitude coupling were observed in multiple brain areas and with very similar distributions at execution, and during the delay period in the absence of movement (Fig. 5). In fact, our results suggest higher PAC during the goal-encoding and planning phase than during movement execution. The most prominent PAC effects in primary motor cortex were obtained for alpha-gamma interactions, but were only present during the planning period. This alpha-gamma PAC in primary motor cortex (M1) virtually disappeared at the time of execution. Interestingly, it was replaced by an M1 delta-gamma PAC during motor execution. In addition, as shown in the right panel of Fig. 5, depth recordings in the medial wall

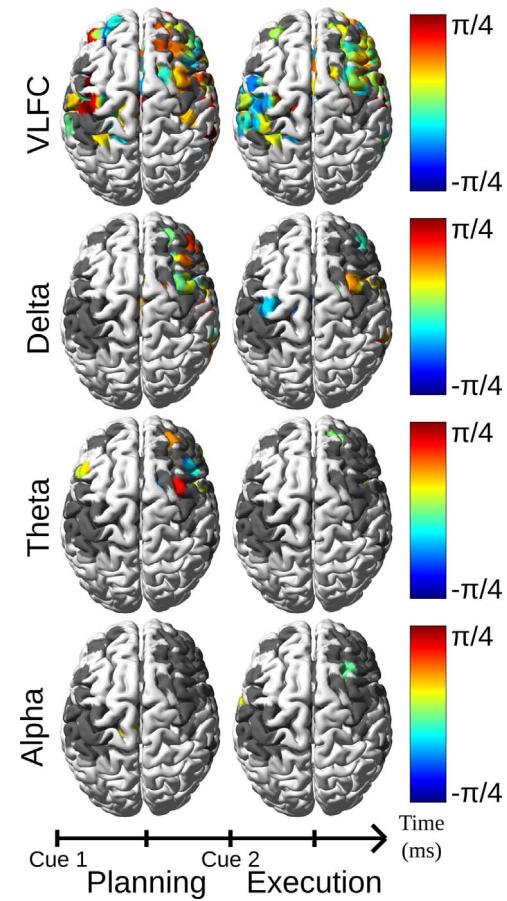


Fig. 4. Task-related modulations of instantaneous phase during planning and execution of upper limb movements. The color bars represent mean phase computed at 500 ms (planning) and at 2250 ms (execution), which corresponds to the centers of the windows used for power (Fig. 3). Phase modulations were computed for very low frequency component (VLFC) (< 1.5 Hz), delta (2–4 Hz), theta (5–7 Hz) and alpha (8–13 Hz) bands. All modulations were statistically significant (Rayleigh's test), based on the circular statistics toolbox (Berens et al., 2009), $p < 0.001$. Dark gray areas represent cortical regions for which electrode coverage was available but, where the modulations did not reach statistical significance. By convention, the left hemisphere (LH) is presented on the left.

revealed statistically significant PAC in anterior cingulate cortex (ACC), in particular, cingulate motor cortex (CMA, BA32) and in medial premotor areas, specifically in supplementary motor area (SMA, BA6) for three slow-frequency bands (delta, theta and alpha). These PAC modulations are likely to reflect the involvement of SMA and cingulate motor areas in action selection, planning, execution, and inhibition.

3.4. Single-trial decoding of motor processes using machine learning

The application of a supervised learning framework allowed us to determine which features, among those discussed above, were useful for single-trial decoding of motor states (binary classification applied pairwise to rest, planning and execution states, as well as a three-class decoding of rest, planning and execution). This was achieved using a cross-validation approach in which a classifier was repeatedly trained on a subset of the data and, then, tested on previously unseen single-epoch observations (test set). This method allowed us to quantify the decoding strength of each feature with a percent correct classification rate.

Fig. 6 depicts mean decoding accuracies (DA) achieved with each feature (power, phase, PAC) across all frequency bands, and all Brodmann areas for which recordings were available (The same significant results pooled by feature, but averaged across Brodmann

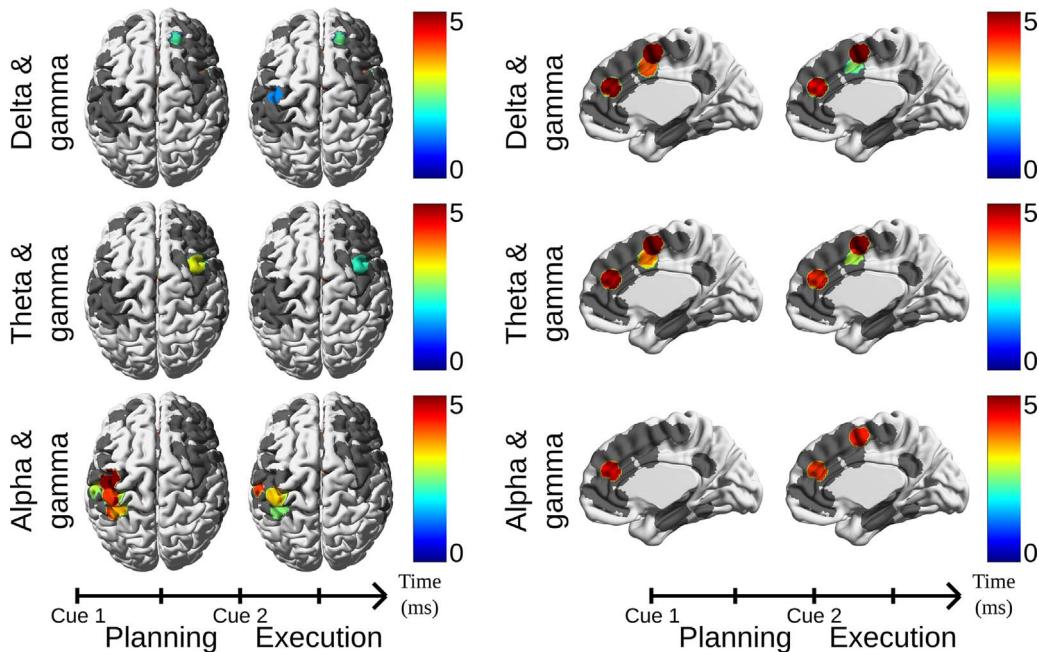


Fig. 5. Task-related modulations of phase-amplitude coupling during planning and execution of upper limb movements (Left panel: top view, Right panel: medial view). The color bars represent mean PAC modulation index (MI), quantifying the co-modulation of broadband gamma amplitude (60–200 Hz) with delta (2–4 Hz), theta (5–7 Hz) and alpha (8–13 Hz) bands. All modulations shown were statistically significant ($p < 0.001$, surrogate data), and are estimated from the entire planning and execution time windows. Dark gray areas represent cortical regions for which electrode coverage was available but, where the modulations did not reach statistical significance. By convention, the left hemisphere (LH) is presented on the left.

areas are available in [Supplementary Fig. S2](#)). The bar plots above each panel depict the highest decoding accuracies obtained with each feature, and the Brodmann area in which it was observed. The bar plots on the right side of each panel indicate the number of significant features present in each Brodmann area. It is noteworthy that all three feature types (amplitude, phase and PAC) computed across many frequencies, and most of the probed brain areas, contained to variable degrees discriminant information on motor states (All results in [Fig. 6](#) represent statistically significant results using permutation tests corrected for multiple comparisons using maximum statistics, $p < 0.05$).

3.4.1. Classification similarities between Execution vs Rest and Intention vs Rest

The highest percentages of correct prediction were obtained when classifying Rest vs. Execution (as high as 94.9% on single-trial classifications). A high degree of similarity was observed between the decoding patterns for Execution vs Rest ([Fig. 6A](#)), and Intention vs Rest ([Fig. 6B](#)). In principle, most of the features that provided significant decoding of Execution vs Rest, also provided significant decoding of Intention vs Rest. The features that yielded the highest discrimination in both cases primarily involved spectral power in beta, low-gamma, and broadband gamma bands, in addition to phase in the very low frequency range (< 1.5 Hz). The most prominent brain areas involved in both types of decoding were BA 4, 6, 8, 9, 13, 32 and 40. These were the regions that contained the highest number of significantly decoding sites (gray histograms), and these were also the BAs that yielded the highest levels of decoding accuracy. The highest DA performances reached 94.9% (VLFC phase in premotor cortex) when decoding Execution vs Rest, and 91.3% (VLFC phase in inferior parietal cortex, BA40) in the case of Intention vs Rest.

3.4.2. Classification differences between Execution vs Rest and Intention vs Rest

Although the decoding matrices in [Fig. 6A](#) (Execution vs Rest) and [6B](#) (Intention vs Rest) show a high degree of similarity, there were also notable differences between the two. First, many more significant sites were obtained in the case of Execution vs Rest (cf. gray bar plots on the

right of each matrix). Furthermore, the low frequency phase in BA 40 played a more central role in distinguishing intention from rest, than movement execution from rest. This could be in line with what is known about the role of the inferior parietal lobe in motor planning ([Caspers et al., 2008](#); [Mattingley et al., 1998](#); [Rushworth et al., 2001](#)). In addition, power-based classification results showed that, while primary motor (BA4) and premotor areas (BA6) were important in predicting execution, the most noticeable intention-decoding power features were recorded from BA9, 13 and 32 (top histogram in Panel 6B). This is consistent with what is reported in the literature on the involvement of BA9 in the processes of working memory, executive planning and behavioral inhibition. Among other things, the important decoding levels in BA13 could be related to the role of insular cortex in motor planning ([Lacourte et al., 2005](#); [Stephan et al., 1995](#)). Remarkably, the best PAC-mediated decoding of Intention vs Rest was obtained with delta-gamma coupling in the dorsal anterior cingulate cortex (dACC, BA32, DA=70,1%). This prominent PAC in ACC is consistent with the role of this brain region in motor inhibition, visuo-spatial attention, motor planning and imagery ([Cheyne et al., 2012](#); [Jahanshahi et al., 1995](#); [Paus, 2001](#)). To the best of our knowledge, this is the first report of a motor-related delta-gamma PAC modulation in ACC.

3.4.3. Classification of Execution versus Intention

The decoding matrix of [sub-Fig. 6C](#) shows a markedly different pattern for Execution vs Intention. Because of the similarity between the neuronal responses in both conditions (cf. previous sections), the differences are subtle and, as a result, the decoding performances are lower than those reported for the Execution vs Rest, or Intention vs Rest classifications. In addition, in this comparison, fewer brain areas allowed for statistically significant decoding. Interestingly, among all features, the one that provided the best discrimination between the intention period and the execution period was the phase of the very low frequency range (< 1.5 Hz), which yielded a decoding peak of 88.9% in BA 8. The second best decoding feature was broadband gamma with a decoding that peaked at 83.5% in the posterior insula (BA13). Moreover, the premotor cortex (BA6) appeared to be overall the most

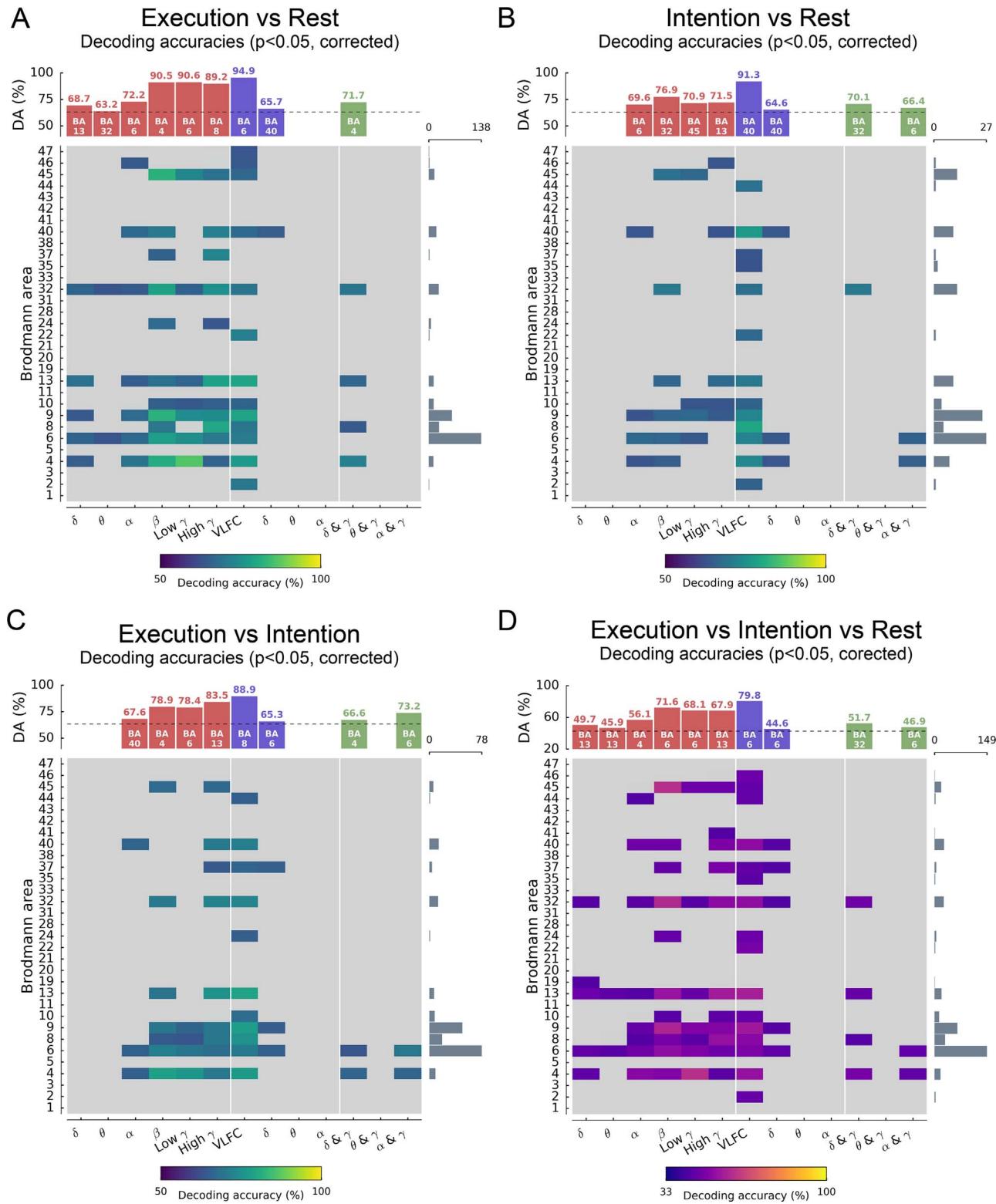


Fig. 6. Single-trial classification of motor states (pre-stimulus rest, intention and execution) using the features of power, phase and PAC. **(A)** Execution vs Rest, **(B)** Intention vs Rest, **(C)** Execution vs Intention, **(D)** Execution vs Intention vs Rest. The decoding matrices depict the mean percent decoding accuracies (DA) for all statistically significant features, across all frequency bands (x-axes) and across all Brodmann Areas in which recordings were available (y-axes). Yellow on the color-bar indicates 100% DA, the lower bounds of the color bars correspond to the chance levels of 50% in the pairwise decoding (A-C) and 33% in the 3-class decoding (D). Only statistically significant decoding accuracies are reported ($p < 0.05$, permutation test, corrected via maximum statistics). The bar plots above each panel show the highest DA obtained with each feature (red: power, blue: phase, and green: PAC), as well as the Brodmann area where it was observed. The gray bar plots on the right side of each panel indicate how many statistically significant features were found in each Brodmann area (note that the upper bounds on the y-axes of these bar plots differ across panels). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

prominent area for Execution vs Intention decoding, providing the largest number of statistically significant sites and features. This is in line with the established involvement of the premotor cortex in motor execution, planning and imagery, as well as visuomotor and visuospatial attention (Ball et al., 1999; Gallivan et al., 2013; Hanakawa et al., 2008; Miller et al., 2010). Finally, the data showed that delta-gamma and alpha-gamma provided the best PAC decoding (respectively in BA4 and BA6) cortices (Fig. 6C), while theta-gamma coupling did not provide significant decoding. This finding is best explained by the significant decoding accuracy (in these areas) of delta-gamma and alpha-gamma observed in Execution vs Rest (Fig. 6A) and Intention vs Rest (Fig. 6B), respectively.

3.4.4. Three-class decoding: Execution versus Intention versus Rest

Three-class decoding results (Fig. 6D) provide results that are very similar to those observed in the pair-wise comparisons. The peak decoding was obtained with the VLCF feature in BA6 (79.8%, with a theoretical chance level at 33%). This result confirms the decoding results observed in the pair-wise decoding (Fig. 6A-C). The second and third best results also originate from sites in premotor area (BA6), respectively with the beta and low-gamma power.

4. Discussion

4.1. Summary of findings

Previous reports have reported complex patterns of overlap and segregation between the networks of brain areas involved in action representation, planning and execution (e.g. Jeannerod, 1994; Stephan et al., 1995; Hanakawa et al., 2008; Guillot et al., 2012;). So far, electrophysiological explorations of these patterns in humans have focused on modulations of rhythmic activity, primarily measured through spectral power modulations. Using direct recordings in humans performing a delayed motor task, we provide by the present article an intracranial investigation of the similarities and discrepancies between activations in areas involved in the preparation of action, and those that actually mediate movement execution. To the best of our knowledge, this study provides the first account of these key brain dynamics through a systematic investigation of the roles of phase, amplitude and phase-amplitude coupling across widely distributed brain areas (extending beyond primary motor areas), and across a wide range of frequencies (up to 200 Hz). Furthermore, in addition to statistical comparisons, an important addition of this study was the use of supervised learning and a single-trial classification framework as ways to assess the distinct and overlapping information content of these features.

4.2. Planning and execution are associated with prominent phase, amplitude and PAC modulations

Our analyses of task-based modulations revealed spatially distributed patterns of statistically significant changes in power, phase and PAC, both during the delay period and following the execution cue. Large amounts of overlap were uncovered between the significant modulations observed in the planning processes and those in movement execution. In task-based power changes, significant increases in the delta, theta and broadband gamma bands were present during the planning phase and were further enhanced at the time of execution. By contrast, alpha, beta and low-gamma power showed a rather consistent pattern of inversion, from increases following the preparation cue to decreases following the execution cue. These activation pattern findings in intention and execution echo previously reported similarities and differences between motor imagery and motor execution. In particular, action representation and motor inhibition were likely to be common to, both, the delay period of the task, and to motor imagery tasks (Stephan et al., 1995). Phase and PAC also displayed similar patterns of

responses for action preparation and execution, although they were also indicative of clear discrepancies in motor areas, such as prominent alpha-gamma coupling during planning, but delta-gamma coupling during execution. Although not spatially exhaustive, the reported pre- and peri-movement modulations in power, phase and PAC were observed over large parts of parietal, frontal and prefrontal areas, including medial areas, such as the dACC. Moreover, while the reported power modulations were largely consistent with previous iEEG reports (Crone, 1998; Crone et al., 2006, 1998; Pfurtscheller and Lopes da Silva, 1999), the wide-spread significant modulations of phase and PAC effects during the delay-period and movement execution are, to our knowledge, novel findings that extend the current understanding of their role in mediating motor behavior.

It is noteworthy that the PAC observations reported here are partly consistent with those of an earlier ECoG study (Yanagisawa et al. 2012). By computing cross-frequency coupling in electrodes implanted in motor areas, the authors found that high-gamma amplitude during waiting was strongly coupled with alpha phase. However, this alpha-gamma PAC was not predictive of movement type and was strongly attenuated towards the timing of motor execution. Similarly, we found prominent motor alpha-gamma PAC during the delay period, which then disappeared during movement execution. Interestingly, in our dataset, the alpha-gamma coupling during the planning was replaced by a delta-gamma coupling at execution. In addition, our data provide evidence for PAC modulations across wider brain areas, beyond primary motor areas.

Future studies might not only benefit from investigating brain activation dynamics through the role of phase, amplitude and phase-amplitude coupling during motor intention and execution, but also during explicit mental simulation of the same movement.

4.3. Decoding patterns reveal functional overlap and discrepancies across phase, amplitude and PAC

The machine learning framework allowed for deeper investigation of the role of our features through the assessment of their decoding accuracy. Most importantly, the classification strategy extended the standard statistical analyses by switching from comparisons of means to the evaluation via cross-validation of the predictive power of the data computed from single-trials; once trained, the classifier was individually applied to each single-trial of the test set. In other words, the mean decoding performances reflected the ability of each feature to discriminate between the conditions (Rest, Intention, Execution) on single samples of data. Taken together, the decoding analyses provide a rich, multi-dimensional, exploration of the functional involvement of phase, amplitude and phase-amplitude in motor behavior. The main findings can be summarized as follows. First, the slow frequency phase achieved the highest DAs across all classifications; Our results revealed its prominent role primarily in inferior parietal areas (DA > 90%). Second, we provide evidence for significant decoding using phase-amplitude coupling in dACC (DA > 70%) during the delay-period preceding motor execution. Third, the performance of power features dropped substantially in the condition Intention vs Rest (below 77%) compared to Execution vs Rest (ca. 90%). During planning, the best power-based decoding levels were observed in prefrontal regions (BA9, 13, 32, 40 and 45), while execution decoding using power revealed strong classifications in BA4, 6, 8 and 9, including precentral and premotor regions. Finally, we showed that when it comes to directly distinguishing the neuronal correlates of execution from those mediating intention, the highest classifications (among all explored regions) were achieved either with the phase of the very low frequency range (< 1.5 Hz) in BA8 (DA > 88%), or with broadband gamma power (60–200 Hz) in the posterior insula (DA > 83%).

4.4. Relationship with previous electrophysiological findings

Overall, the present analysis of intracranial human data provides strong evidence for a key role of phase and phase-amplitude relationships in motor goal encoding, and execution. These original findings extend previous invasive electrophysiological studies in humans that have investigated the correlates of motor intention and/or execution. For the large part, such studies were primarily based on the estimation of cortical power modulations in multiple frequency components of the LFP, including beta and gamma range power modulations (Leuthardt et al., 2004; Crone et al., 2006; Miller et al., 2007; Gunduz et al., 2016). Nevertheless, there is some evidence suggesting a critical role for low-frequency phase and amplitude information in mediating movement parameters (Hammer et al., 2016, 2013; Jerbi et al., 2011, 2007; Milekovic et al., 2012; Miller et al., 2012; Waldert et al., 2009, 2008; Kajihara et al., 2015; Liu et al., 2011). Using ECoG recordings in epilepsy patients, Hammer and colleagues (2013) directly addressed the role of phase information in decoding movement kinematics (i.e. position, velocity, and acceleration). By separately exploring kinematics decoding with spectral amplitudes and phase of the low frequency component (LFC), the authors came to the conclusion that the ECoG LFC phase was in fact much more informative than amplitude (Hammer et al., 2013).

In theory, it is conceivable that our single-trial decoding in the low frequency range (e.g. VLFC decoding) is driven at least in part by the presence of phase-locking across trials.

For instance, our decoding results may be related to the presence of a Bereitschaftspotential (BP), since the latter varies with movement states and parameters (Birbaumer et al., 1990; Shibasaki and Hallett, 2006). A consistent phase-locking across trials related to such slow cortical potentials may substantially contribute for instance to the high VLFC phase decoding we observed. A key question is whether this VLFC phase feature provides higher decoding than the envelope or the low-pass filtered raw signal in the same frequency range. To address this interesting question, we conducted additional decoding analyses where we directly compared decoding performance (across all available data) with the 3 following types of features (a) low-pass filtered [0.1 1.5 Hz] raw signal, (b) [0.1 1.5 Hz] phase (VLFC) and the (c) [0.1 1.5 Hz] envelope. The results (Fig. 7) suggest that the phase of the low-frequency signal contains more relevant information than its envelope in terms of decoding. When contrasted to the low-pass filtered signal, the phase provided either comparable or higher (cf. planning vs rest) motor state decoding. Note that these results were obtained in single trial classification (as is the case with all the rest of our analyses). These findings imply that single-trial low-frequency components (which contribute, when averaged, to generating motor-related potentials) provide significant motor state decoding performances. Our results (Fig. 7) indicate that this low-frequency decoding appears to be, on a trial-by-trial basis, primarily driven by the phase of the low-pass signal, rather than its envelope.

A recent study demonstrated that the LFC signal may be more adapted to speed decoding than to velocity decoding (Hammer et al., 2016). These findings are globally in line with reports of low frequency range coherence (< 4 Hz) between limb speed and the activity of the primary motor cortex during continuous movements (Jerbi et al., 2007). Moreover, movement direction classification has been achieved with significant success using the LFC component, both, with invasive and non-invasive brain recordings (Jerbi et al., 2011; Waldert et al., 2009, 2008).

Our observation of the efficacy of low frequency brain signals in motor decoding is consistent with a few animal studies that have revealed a prominent role for frequencies (< 5 Hz) LFP components in decoding motor movements via recordings in primary motor and parietal cortices (e.g. Mehring et al., 2004; Averbeck et al., 2005; Rickert, 2005).

The results of the current study differ from the above reports in

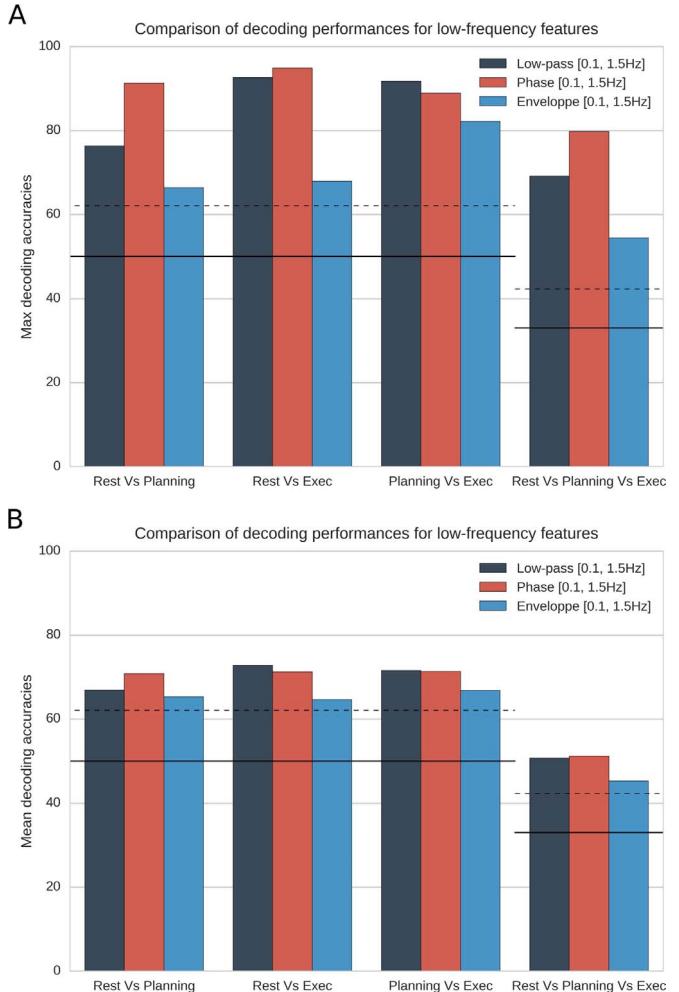


Fig. 7. Comparison of maximum decoding performance using 3 features in the same low frequency band (0.5–1.5 Hz): Low-pass filtered signal (deep blue), low-pass phase (red) and low-pass envelope (light blue). (A) Maximum decoding achieved with each feature type, and across the 4 classification problems. (B) Same as in (A) but here mean decoding across all statistically significant sites. In both panels, theoretical chance levels (50% or 33%) are depicted with continuous lines and the statistical thresholds ($p < 0.05$, corrected) are indicated by dashed lines. The low-frequency phase features provide higher classifications than the low-frequency envelope. The entire low-pass signal (deep blue) achieves decoding levels closer to those of the phase component. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

several ways. Firstly, while most studies typically focus on movement execution, the delayed-motor task used here allowed us to examine the role of phase information in the motor encoding and planning phase preceding actual movement execution. Secondly, beyond phase or amplitude, we performed a systematic exploration of PAC effects and found that it yielded significant modulations and motor state decoding. Thirdly, the intracerebral recordings of our subject pool provided a different spatial sampling compared to most previous electrocorticographic studies. In particular, the recording sites covered widely distributed brain areas beyond primary and secondary motor areas (such as parietal, prefrontal and insular cortices), and included some medial brain areas (e.g. dACC). Moreover, most of the previous investigations into the role of low frequencies (< 4 Hz) in movement decoding relied on the low-pass filtered brain signals. By contrast, we specifically focused on the phase at these low frequencies, but we also examined the difference with using the envelope or low-pass filtered signals (Fig. 7). Therefore, taken together, our exploration of intracranial recording during a delayed motor task confirms and extends the growing body of evidence for the role of phase and phase-related

measures in encoding motor processes, from intention to execution.

Invasive recordings in patients with motor disorders have been reported both with ECoG and passive recordings from deep brain stimulation (DBS) electrodes, in particular in the case of Parkinson's disease (McIntyre and Thakor, 2002; Litvak et al. 2011; Florin et al. 2013; Beudel et al. 2015; Rowland et al. 2015; Guridi and Alegre, 2016). For instance, the recent intracranial study by Rowland et al. (2015) bares some similarities with our study since it set out to explore beta and gamma power changes across rest, movement preparation and execution in Parkinson's patients. In particular, the authors used DBS recordings to monitor event-related synchronization and desynchronization associated with rigidity or akinesia in these patients. The study showed that compared to patients with essential tremor (who do not show rigidity or akinesia), Parkinson's patients had larger beta desynchronization in early motor preparation as well as enhanced cortical broadband gamma power during both rest and task. The authors interpret the findings as an indication that the dynamic profile of sensorimotor cortex oscillations in Parkinson's is in opposition with the anti-kinetic activity of the basal ganglia.

4.5. A key role for phase coding in goal-directed motor behavior

Currently, the overwhelming body of research on the oscillatory brain dynamics that mediate motor behavior continues to explore local activations in terms of amplitude modulations, while phase information is primarily used in the context of inter-regional connectivity assessments (da Silva, 2006; Lachaux et al., 1999; Le Van Quyen et al., 2001; Roach and Mathalon, 2008). Our findings support the view that phase carries critical information that is often overlooked, and can provide more task-specific information than spectral amplitude. Thus, one may wonder about the underlying physiological phenomenon that could explain the relevance of phase. One potential explanation is that phase modulations might be closely related to neuronal firing patterns. Evidence for this hypothesis has been found in non-human primates, for instance, in auditory and visual cortices (Ng et al., 2013; Montemurro et al., 2008). In the auditory cortex, stimulus selective firing patterns have been found to imprint on the phase, rather than the amplitude, of theta oscillations within both LFPs and EEG data (Ng et al. 2013). By applying a stimulus decoding technique to intracortical LFPs and single cell recordings in macaque auditory cortex, the authors found that the stimuli, which were successfully discriminated by firing rates, were also discriminated by phase patterns but not by oscillation amplitude. In visual cortex, information theoretical approaches have also shown that the combination of spikes and low frequency (1–4 Hz) LFP phase provide more information on visual stimuli than spikes alone (Latham and Lengyel, 2008; Montemurro et al., 2008). Similarly, the timing of action potentials relative to LFP theta phase in the hippocampus have been found to be more informative about position (i.e., higher precision) than what could be inferred from firing rate alone (Dragoi and Buzsáki, 2006; O'Keefe and Recce, 1993).

In the present study, the low-frequency phase decoding results could be theoretically explained by (and be consistent with) the hypothesis that the precise relationship between slow LFP phase and neuronal firing reported in other modalities (as discussed above), also operates in human motor brain areas during limb movement preparation and execution. This idea needs to be tested with microelectrode data allowing for simultaneous access to both LFP and neuronal firing in human participants. This was not possible with the SEEG data used for the purpose of this study. Furthermore, the significant modulations and decoding results reported here, using phase-amplitude coupling, could be suggestive of a specific mechanism by which phase, broadband gamma amplitude and neuronal firing are lined up in a precise fashion. Whittingstall and Logothetis (2009) observed such a phenomenon in macaque visual cortex, where multi-unit firing responses were found to be strongest only when increases in EEG gamma power occurred during a specific phase of the delta (2–4 Hz) wave. It is however

obvious that, the tempting speculation that this may also be at play in areas where we found significant PAC decoding, cannot be confirmed without simultaneous monitoring of neuronal firing.

While the notion of phase coding is not new *per se*, the presented results extend an emerging body of literature suggesting that phase coding may play a more important role in motor behavior than previously assumed. In particular, we found that phase-based motor state decoding (i) is widely distributed, extending beyond primary motor areas, (ii) occurs both during action planning and execution, and (iii) that phase-amplitude coupling could be an additional phase-based key feature involved in the neuronal coding of goal-directed behavior.

4.6. Limitations and open questions

The results of this study have a number of limitations. Our data, just like all previous intracranial EEG studies, have limited spatial sampling of the involved networks. The over 500 intracerebral sites used here provide only a partial spatial coverage of the brain, with a more dense coverage of right frontal than left frontal regions (Fig. 1A). Additionally, a reliable comparison between neural responses contra and ipsilateral to the moving hand was not possible because 4 patients out of 6 had uni-lateral implantations, and even for those with bilateral implantation, the electrodes were not located in exact homologous brain areas in both hemispheres. This is a typical limitation of invasive human data. Clearly, investigating the lateralization of amplitude, phase and PAC phenomena reported here, would greatly benefit from further investigations with full head coverage using MEG or EEG and similar delayed-motor paradigms. This said, patients were selected primarily based on whether they had electrodes implanted in frontal, prefrontal or parietal areas, thus providing a reasonable coverage of the targeted networks. Moreover, as in all previous human invasive recordings, participants suffered from drug-resistant epilepsy, which could limit the generalizability of the findings to healthy subjects. To address this in the best possible way, our standard procedure (e.g. Jerbi et al., 2009) is to exclude electrodes that display pathological activity (such as epileptic spikes), and to focus on task-related changes and multi-trial analyses, thereby reducing the impact of neuronal activations that are spurious or unrelated to the task. Therefore, our findings could benefit from future replication using non-invasive recordings in healthy controls.

Note that all the decoding throughout this study was performed on single-features in single-trial mode, individually in each subject. We did not use multi-feature classifications, and did not pool all participants in the decoder. The nature of the individualized SEEG implantation precludes the possibility to reliably combine data from multiple subjects into the same cross-validation decoding framework. We chose to provide an overview of the single-feature findings across BA (mean and max significant decoding accuracies, Fig. 6) as a pragmatic way to pool the multi-subject results. Of course, the decoding accuracies would have likely been even higher had we chosen to run multi-feature classification at least within individual subjects. Yet, our aim was not to achieve the highest decoding rate, rather to use the decoding approach to unravel and compare the distinct contribution of each feature individually to motor planning and execution processes.

Furthermore, it is important to acknowledge that our task design did not allow for a fine-grained disambiguation of the various, distinct, processes that are expected to take place during the delay period. Clearly, the delay period encompasses a wide range of processes, such as stimulus encoding, visuomotor transformations, motor imagery, action selection, motor preparation, as well as working memory and maintenance processes. To limit any bias towards visual stimulus encoding when analyzing the planning/intention period, we deliberately centered our phase and amplitude analyses on 500 ms after target onset. In future investigations, this limitation should be addressed with a specific experimental design aimed at disentangling various motor related processes in the delay period.

In addition, the Go cue is likely to be immediately followed both by preparatory and execution signals. Thus, a more detailed analysis of motor execution would have been possible if we were able to align the data to movement onset rather than the Go cue. Unfortunately, for technical reasons movement onset was not available. To limit the impact of this limitation, we deliberately considered a window in the execution phase between 500 ms and 1000 ms after the Go Cue, and thus ignored the 500 ms post-Go period where most of the preparation is expected to take place. More importantly, because we relied on a decoding approach to disentangle execution from preparation, we hypothesized that processes (or feature modulations) common to both conditions would not allow for significant decoding, and thus by definition, features with high decoding should reflect differences between the two states.

Moreover, the physiological interpretations made in the current study also bare their share of limitations. The way the explored features (amplitude, phase and PAC) explicitly relate to the notion of information processing in the human brain is still poorly understood. Rather than identifying the precise functional relevance of each feature, our results emphasized their task-specific modulations and their ability to successfully predict the state of the system in single-trial data (i.e. Rest, motor Planning or motor Execution).

Despite the above limitations, we feel that access to intracerebral depth EEG recordings in human subjects, provides privileged insight into the neural dynamics mediating human cognition, with superior spatial, temporal and spectral precision. In the long run, this type of data can help bridge the gap between neuroimaging studies and electrophysiological recordings in non-human primates.

5. Conclusion

The findings of this study provide novel experimental evidence for the role of oscillatory phase and amplitude properties in motor planning and execution. In particular, the evidence for phase and PAC-based coding are a compelling support for the key role of phase in encoding motor representations and mediating movement execution across widely distributed brain areas. We are confident that these results will pave the way for a better understanding of, and novel hypotheses about, the roles of phase, amplitude, and the coordination between the two, in goal-directed motor behavior in humans.

Conflict of interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

Etienne Combrisson was supported in part by a Ph.D. Scholarship awarded by the Ecole Doctorale Inter-Disciplinaire Sciences-Santé (EDISS), Lyon, France, and by Ph.D. funding from the Natural Sciences and Engineering Research Council of Canada (NSERC). This work was partly performed within the framework of the LABEX CORTEX (ANR-11-LABX-0042) of Université de Lyon, within the program ANR-11-IDEX-0007, and by the ANR (project FORCE, ANR-13-TECS-0013-01), SEMAINE, the Human Brain Project (Human intracerebral database, Grant number 604102) to J.-P. Lachaux; We acknowledge support from the Brazilian Ministry of Education (CAPES grant 1719-04-1), the Foundation pour la Recherche Médicale (FRM, France) and the Fulbright Commission to J.L.P. Soto, and funding from the Canada Research Chairs program and a Discovery Grant (RGPIN-2015-04854) awarded by the Natural Sciences and Engineering Research Council of Canada to K.J. The authors are grateful for the collaboration of the patients and clinical staff at the Epilepsy Department of the Grenoble University Hospital.

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at <http://dx.doi.org/10.1016/j.neuroimage.2016.11.042>.

References

- Ariani, G., Wurm, M.F., Lingnau, A., 2015. Decoding internally and externally driven movement plans. *J. Neurosci. Off. J. Soc. Neurosci.* 35, 14160–14171. <http://dx.doi.org/10.1523/JNEUROSCI.0596-15.2015>.
- Aru, J., Aru, J., Priesemann, V., Wibral, M., Lana, L., Pipa, G., Singer, W., Vicente, R., 2015. Untangling cross-frequency coupling in neuroscience. *Curr. Opin. Neurobiol. SI: Brain Rhythm. Dyn. Coord.* 31, 51–61. <http://dx.doi.org/10.1016/j.conb.2014.08.002>.
- Averbbeck, B.B., Chafee, M.V., Crowe, D.A., Georgopoulos, A.P., 2005. Parietal representation of hand velocity in a copy task. *J. Neurophysiol.* 93, 508–518.
- Babiloni, C., Del Percio, C., Vecchio, F., Sebastian, F., Di Gennaro, G., Quarato, P.P., Morace, R., Pavone, L., Soricelli, A., Noce, G., Esposito, V., Rossini, P.M., Gallesse, V., Mirabella, G., 2016. Alpha, beta and gamma electrocorticographic rhythms in somatosensory, motor, premotor and prefrontal cortical areas differ in movement execution and observation in humans. *Clin. Neurophysiol. Off. J. Int. Fed. Clin. Neurophysiol.* 127, 641–654. <http://dx.doi.org/10.1016/j.clinph.2015.04.068>.
- Babiloni, C., Babiloni, F., Carducci, F., Cincotti, F., Rosciarelli, F., Arendt-Nielsen, L., Chen, A.C.N., Rossini, P.M., 2002. Human brain oscillatory activity phase-locked to painful electrical stimulations: a multi-channel EEG study. *Hum. Brain Mapp.* 15, 112–123. <http://dx.doi.org/10.1002/hbm.10013>.
- Bahramisharif, A., van Gerven, M.A.J., Aarnoutse, E.J., Mercier, M.R., Schwartz, T.H., Foxe, J.J., Ramsey, N.F., Jensen, O., 2013. Propagating neocortical gamma bursts are coordinated by traveling alpha waves. *J. Neurosci.* 33, 18849–18854. <http://dx.doi.org/10.1523/JNEUROSCI.2455-13.2013>.
- Ball, T., Schreiber, A., Feige, B., Wagner, M., Lücking, C.H., Kristeva-Feige, R., 1999. The role of higher-order motor areas in voluntary movement as revealed by high-resolution EEG and fMRI. *Neuroimage* 10, 682–694.
- Ball, T., Demandt, E., Mutschler, I., Neitzel, E., Mehring, C., Vogt, K., Aertsen, A., Schulze-Bonhage, A., 2008. Movement related activity in the high gamma range of the human EEG. *Neuroimage* 41, 302–310.
- Bastin, J., Deman, P., David, O., Gueguen, M., Benis, D., Minotti, L., Hoffman, D., Combrisson, E., Kujala, J., Perrone-Bertolotti, M., Kahane, P., Lachaux, J.-P., Jerbi, K., 2016. Direct recordings from human anterior insula reveal its leading role within the error-monitoring network. *Cereb. Cortex Behav.* 352. <http://dx.doi.org/10.1093/cercor/bhv352>.
- Berens, P., et al., 2009. CircStat a MATLAB toolbox for circular statistics. *J. Stat. Softw.* 31, 1–21.
- Beudel, M., Little, S., Pogosyan, A., Ashkan, K., Foltyne, T., Limousin, P., Zrinzo, L., Hariz, M., Bogdanovic, M., Cheeran, B., others, 2015. Tremor reduction by deep brain stimulation is associated with gamma power suppression in Parkinson's disease. *Neuromodul. Technol. Neural Interface* 18, 349–354.
- Birbaumer, N., Elbert, T., Canavan, A.G., Rockstroh, B., 1990. Slow potentials of the cerebral cortex and behavior. *Physiol. Rev.* 70, 1–41.
- Brovelli, A., Lachaux, J.-P., Kahane, P., Boussaoud, D., 2005. High gamma frequency oscillatory activity dissociates attention from intention in the human premotor cortex. *Neuroimage* 28, 154–164. <http://dx.doi.org/10.1016/j.neuroimage.2005.05.045>.
- Bundy, D.T., Pawha, M., Szrama, N., Leuthardt, E.C., 2016. Decoding three-dimensional reaching movements using electrocorticographic signals in humans. *J. Neural Eng.* 13, 026021. <http://dx.doi.org/10.1088/1741-2560/13/2/026021>.
- Canolty, R.T., 2006. High gamma power is phase-locked to theta. *Science* 313, 1128115.
- Canolty, R.T., Knight, R.T., 2010. The functional role of cross-frequency coupling. *Trends Cogn. Sci.* 14 (11), 506–515. <http://dx.doi.org/10.1016/j.tics.2010.09.001>.
- Caspers, S., Eickhoff, S.B., Geyer, S., Schepersjans, F., Mohlberg, H., Zilles, K., Amunts, K., 2008. The human inferior parietal lobule in stereotaxic space. *Brain Struct. Funct.* 212, 481–495. <http://dx.doi.org/10.1007/s00429-008-0195-z>.
- Cheyne, D.O., Ferrari, P., Cheyne, J.A., 2012. Intended actions and unexpected outcomes: automatic and controlled processing in rapid motor task. *Front. Hum. Neurosci.* 6, 237. <http://dx.doi.org/10.3389/fnhum.2012.00237>.
- Cheyne, D., Bells, S., Ferrari, P., Gaetz, W., Bostan, A.C., 2008. Self-paced movements induce high-frequency gamma oscillations in primary motor cortex. *Neuroimage* 42, 332–342. <http://dx.doi.org/10.1016/j.neuroimage.2008.04.178>.
- Cohen, M.X., Elger, C.E., Fell, J., 2008. Oscillatory activity and phase-amplitude coupling in the human medial frontal cortex during decision making. *J. Cogn. Neurosci.* 21, 390–402.
- Combrisson, E., Jerbi, K., 2015. Exceeding chance level by chance: the caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy. *J. Neurosci. Methods* 250, 126–136. <http://dx.doi.org/10.1016/j.jneumeth.2015.01.010>.
- Crone, N., 1998. Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis I. Alpha and beta event-related desynchronization. *Brain* 121, 2271–2299. <http://dx.doi.org/10.1093/brain/121.12.2271>.
- Crone, N.E., Miglioretti, D.L., Gordon, B., Lesser, R.P., 1998. Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis II. Event-related synchronization in the gamma band. *Brain*, 2301–2315.

- Crone, N.E., Sinai, A., Korzeniewska, A., 2006. High-frequency gamma oscillations and human brain mapping with electrocorticography. In: *Progress in Brain Research*. Elsevier, pp. 275–295.
- da Silva, F.H.L., 2006. Event-related neural activities: what about phase? *Prog. Brain Res.* 159, 3–17.
- Delorme, A., Makeig, S., 2004. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J. Neurosci. Methods* 134, 9–21.
- Desmurget, M., Sirigu, A., 2009. A parietal-premotor network for movement intention and motor awareness. *Trends Cogn. Sci.* 13, 411–419. <http://dx.doi.org/10.1016/j.tics.2009.08.001>.
- Dragoi, G., Buzsáki, G., 2006. Temporal encoding of place sequences by hippocampal cell assemblies. *Neuron* 50, 145–157. <http://dx.doi.org/10.1016/j.neuron.2006.02.023>.
- Drewes, J., VanRullen, R., 2011. This is the rhythm of your eyes: the phase of ongoing electroencephalogram oscillations modulates saccadic reaction time. *J. Neurosci.* 31, 4698–4708. <http://dx.doi.org/10.1523/JNEUROSCI.4795-10.2011>.
- Dugue, L., Marque, P., VanRullen, R., 2011. The phase of ongoing oscillations mediates the causal relation between brain excitation and visual perception. *J. Neurosci.* 31, 11889–11893. <http://dx.doi.org/10.1523/JNEUROSCI.1161-11.2011>.
- Florin, E., Erasmi, R., Reck, C., Maarouf, M., Schnitzler, A., Fink, G., Timmermann, L., 2013. Does increased gamma activity in patients suffering from Parkinson's disease counteract the movement inhibiting beta activity? *Neuroscience* 237, 42–50.
- Foster, B.L., Parvizi, J., 2012. Resting oscillations and cross-frequency coupling in the human posteromedial cortex. *Neuroimage* 60, 384–391. <http://dx.doi.org/10.1016/j.neuroimage.2011.12.019>.
- Gallivan, J.P., McLean, D.A., Flanagan, J.R., Culham, J.C., 2013. Where one hand meets the other: limb-specific and action-dependent movement plans decoded from preparatory signals in single human frontoparietal brain areas. *J. Neurosci.* 33, 1991–2008.
- Golland, P., Fischl, B., 2003. Permutation tests for classification: towards statistical significance in image-based studies. In: *Proceedings of the 18th Conference of Information Processing in Medical Imaging*, pp. 330–341.
- Guillot, A., Di Rienzo, F., MacIntyre, T., Moran, A., Collet, C., 2012. Imagining is not doing but involves specific motor commands: a review of experimental data related to motor inhibition. *Front. Hum. Neurosci.* 6, 247.
- Gunduz, A., Brunner, P., Sharma, M., Leuthardt, E.C., Ritaccio, A.L., Pesaran, B., Schalk, G., 2016. Differential roles of high gamma and local motor potentials for movement preparation and execution. *Brain-Computer Interfaces* 3, 88–102. <http://dx.doi.org/10.1080/2326263X.2016.1179087>.
- Guridi, J., Alegre, M., 2016. Oscillatory activity in the basal ganglia and deep brain stimulation. *Mov. Disord.*
- Hamamé, C.M., Vidal, J.R., Perrone-Bertolotti, M., Ossandon, T., Jerbi, K., Kahane, P., Bertrand, O., Lachaux, J.-P., 2014. Functional selectivity in the human occipitotemporal cortex during natural vision: evidence from combined intracranial EEG and eye-tracking. *Neuroimage* 95, 276–286.
- Hammer, J., Fischer, J., Ruescher, J., Schulze-Bonhage, A., Aertsen, A., Ball, T., 2013. The role of ECoG magnitude and phase in decoding position, velocity, and acceleration during continuous motor behavior. *Front. Neurosci.* 7. <http://dx.doi.org/10.3389/fnins.2013.00200>.
- Hammer, J., Pistoohl, T., Fischer, J., Krsek, P., Tomášek, M., Marusič, P., Schulze-Bonhage, A., Aertsen, A., Ball, T., 2016. Predominance of movement speed over direction in neuronal population signals of motor cortex: intracranial EEG data and a simple explanatory model. *Cereb. Cortex* 26, 2863–2881. <http://dx.doi.org/10.1093/cercor/bhw033>.
- Hanakawa, T., Dimyan, M.A., Hallett, M., 2008. Motor planning, imagery, and execution in the distributed motor network: a time-course study with functional MRI. *Cereb. Cortex* 18, 2775–2788. <http://dx.doi.org/10.1093/cercor/bhn036>.
- Hempinne, C., de Ryapolova-Webb, E.S., Air, E.L., Garcia, P.A., Miller, K.J., Ojemann, J.G., Ostrem, J.L., Galifanakis, N.B., Starr, P.A., 2013. Exaggerated phase–amplitude coupling in the primary motor cortex in Parkinson disease. *Proc. Natl. Acad. Sci. USA*. <http://dx.doi.org/10.1073/pnas.1214546110>.
- Hyafil, A., Giraud, A.-L., Fontolan, L., Gutkin, B., 2015. Neural cross-frequency coupling: connecting architectures, mechanisms, and functions. *Trends Neurosci.* 38, 725–740. <http://dx.doi.org/10.1016/j.tins.2015.09.001>.
- Jahanshahi, M., Jenkins, I.H., Brown, R.G., Marsden, C.D., Passingham, R.E., Brooks, D.J., 1995. Self-initiated versus externally triggered movements. I. An investigation using measurement of regional cerebral blood flow with PET and movement-related potentials in normal and Parkinson's disease subjects. *Brain J. Neurol.* 118 (Pt 4), 913–933.
- Jenkinson, N., Kühn, A.A., Brown, P., 2013. Gamma oscillations in the human basal ganglia. *Exp. Neurol.* 245, 72–76. <http://dx.doi.org/10.1016/j.expneurol.2012.07.005>.
- Jeannerod, M., 1994. The representing brain: neural correlates of motor intention and imagery. *Behav. Brain Sci.* 17, 187–202.
- Jensen, O., Colgin, L.L., 2007. Cross-frequency coupling between neuronal oscillations. *Trends Cogn. Sci.* 11, 267–269. <http://dx.doi.org/10.1016/j.tics.2007.05.003>.
- Jensen, O., Gips, B., Bergmann, T.O., Bonnefond, M., 2014. Temporal coding organized by coupled alpha and gamma oscillations prioritize visual processing. *Trends Neurosci.* 37, 357–369. <http://dx.doi.org/10.1016/j.tins.2014.04.001>.
- Jerbi, K., Lachaux, J.-P., Karim, N., Pantazis, D., Leahy, R.M., Garnero, L., Baillet, S., others, 2007. Coherent neural representation of hand speed in humans revealed by MEG imaging. *Proc. Natl. Acad. Sci.* 104, 7676–7681.
- Jerbi, K., Ossandon, T., Hamamé, C.M., Senova, S., Dalal, S.S., Jung, J., Minotti, L., Bertrand, O., Berthoz, A., Kahane, P., Lachaux, J.-P., 2009. Task-related gamma-band dynamics from an intracerebral perspective: review and implications for surface EEG and MEG. *Hum. Brain Mapp.* 30, 1758–1771. <http://dx.doi.org/10.1002/hbm.20750>.
- Jerbi, K., Vidal, J.R., Mattout, J., Maby, E., Lecaingard, F., Ossandon, T., Hamamé, C.M., Dalal, S.S., Bouet, R., Lachaux, J.-P., Leahy, R.M., Baillet, S., Garnero, L., Delpuech, C., Bertrand, O., 2011. Inferring hand movement kinematics from MEG, EEG and intracranial EEG: from brain-machine interfaces to motor rehabilitation. *IRBM* 32, 8–18. <http://dx.doi.org/10.1016/j.irbm.2010.12.004>.
- Jerbi, K., Vidal, J.R., Ossandon, T., Dalal, S.S., Jung, J., Hoffmann, D., Minotti, L., Bertrand, O., Kahane, P., Lachaux, J.-P., 2010. Exploring the electrophysiological correlates of the default-mode network with intracerebral EEG. *Front. Syst. Neurosci.* 4, 27. <http://dx.doi.org/10.3389/fnsys.2010.00027>.
- Jurkiewicz, M.T., Gaetz, W.C., Bostan, A.C., Cheyne, D., 2006. Post-movement beta rebound is generated in motor cortex: evidence from neuromagnetic recordings. *Neuroimage* 32, 1281–1289. <http://dx.doi.org/10.1016/j.neuroimage.2006.06.005>.
- Kahane, P., Landré, E., Minotti, L., Francione, S., Rylvil, P., 2006. The Bancaud and Talairach view on the epileptogenic zone: a working hypothesis. *Epileptic Disord. Int. Epilepsy J. Videotape* 8, S16–S26.
- Kajihara, T., Anwar, M.N., Kawasaki, M., Mizuno, Y., Nakazawa, K., Kitajo, K., 2015. Neural dynamics in motor preparation: from phase-mediated global computation to amplitude-mediated local computation. *Neuroimage* 118, 445–455. <http://dx.doi.org/10.1016/j.neuroimage.2015.05.032>.
- Kalaska, J.F., 2009. From intention to action: motor cortex and the control of reaching movements. In: Sternad, D. (Ed.), *Progress in Motor Control*. Springer US, Boston, MA, 139–178.
- Klimesch, W., Freunberger, R., Sauseng, P., Gruber, W., 2008. A short review of slow phase synchronization and memory: evidence for control processes in different memory systems? *Brain Res. Brain Oscil. Cogn. Cogn. Disord.* 1235, 31–44. <http://dx.doi.org/10.1016/j.jbrainres.2008.06.049>.
- Klimesch, W., Sauseng, P., Hanslmayr, S., 2007. EEG alpha oscillations: the inhibition-timing hypothesis. *Brain Res. Rev.* 53, 63–88. <http://dx.doi.org/10.1016/j.jbrainres.2006.06.003>.
- Kramer, M.A., Tort, A.B.L., Kopell, N.J., 2008. Sharp edge artifacts and spurious coupling in EEG frequency comodulation measures. *J. Neurosci. Methods* 170, 352–357. <http://dx.doi.org/10.1016/j.jneumeth.2008.01.020>.
- Lachaux, J.-P., Rodriguez, E., Martinerie, J., Varela, F.J., et al., 1999. Measuring phase synchrony in brain signals. *Hum. Brain Mapp.* 8, 194–208.
- Lachaux, J.P., Rudrauf, D., Kahane, P., 2003. Intracranial EEG and human brain mapping. *J. Physiol. Paris* 97, 613–628. <http://dx.doi.org/10.1016/j.jphysparis.2004.01.018>.
- Lacourte, M.G., Orr, E.L.R., Cramer, S.C., Cohen, M.J., 2005. Brain activation during execution and motor imagery of novel and skilled sequential hand movements. *Neuroimage* 27, 505–519. <http://dx.doi.org/10.1016/j.neuroimage.2005.04.025>.
- Lakatos, P., 2005. An oscillatory hierarchy controlling neuronal excitability and stimulus processing in the auditory cortex. *J. Neurophysiol.* 94, 1904–1911. <http://dx.doi.org/10.1152/jn.00263.2005>.
- Latham, P.E., Lengyel, M., 2008. Phase coding: spikes get a boost from local fields. *Curr. Biol.* 18, R349–R351. <http://dx.doi.org/10.1016/j.cub.2008.02.062>.
- Lau, H.C., 2004. Attention to intention. *Science* 303, 1208–1210. <http://dx.doi.org/10.1126/science.1090973>.
- Le Van Quyen, M., Foucher, J., Lachaux, J.-P., Rodriguez, E., Lutz, A., Martinerie, J., Varela, F.J., 2001. Comparison of Hilbert transform and wavelet methods for the analysis of neuronal synchrony. *J. Neurosci. Methods* 111, 83–98.
- Lee, J., Jeong, J., 2013. Correlation of risk-taking propensity with cross-frequency phase–amplitude coupling in the resting EEG. *Clin. Neurophysiol.* <http://dx.doi.org/10.1016/j.clinph.2013.05.007>.
- Lemm, S., Blankertz, B., Dickhaus, T., Müller, K.-R., 2011. Introduction to machine learning for brain imaging. *Neuroimage* 56, 387–399. <http://dx.doi.org/10.1016/j.neuroimage.2010.11.004>.
- Leuthardt, E.C., Schalk, G., Wolpaw, J.R., Ojemann, J.G., Moran, D.W., 2004. A brain-computer interface using electrocorticographic signals in humans. *J. Neural Eng.* 1, 63. <http://dx.doi.org/10.1088/1741-2560/1/2/001>.
- Litvak, V., Jha, A., Eusebio, A., Oostenveld, R., Foltyne, T., Limousin, P., Zrinzo, L., Hariz, M.I., Friston, K., Brown, P., 2011. Resting oscillatory cortico-subthalamic connectivity in patients with Parkinson's disease. *Brain* 134, 359–374. <http://dx.doi.org/10.1093/brain/awq332>.
- Liu, J., Perdoni, C., He, B., 2011. Hand movement decoding by phase-locking low frequency EEG signals. In: *Proceedings of the Annual International Conference IEEE Engineering in Medicine and Biology Society*. doi:10.1109/IEMBS.2011.6091564. pp. 6335–6338.
- Maris, E., van Vugt, M., Kahana, M., 2011. Spatially distributed patterns of oscillatory coupling between high-frequency amplitudes and low-frequency phases in human iEEG. *Neuroimage* 54, 836–850.
- Mattingley, J.B., Husain, M., Rorden, C., Kennard, C., Driver, J., 1998. Motor role of human inferior parietal lobe revealed in unilateral neglect patients. *Nature* 392, 179–182. <http://dx.doi.org/10.1038/32413>.
- McIntyre, C.C., Thakor, N.V., 2002. Uncovering the mechanisms of deep brain stimulation for Parkinson's disease through functional imaging, neural recording, and neural modeling. *Crit. Rev. Biomed. Eng.*, 30.
- Mehring, C., Nawrot, M.P., de Oliveira, S.C., Vaadia, E., Schulze-Bonhage, A., Aertsen, A., Ball, T., 2004. Comparing information about arm movement direction in single channels of local and epicortical field potentials from monkey and human motor cortex. *J. Physiol. Paris* 98, 498–506. <http://dx.doi.org/10.1016/j.jphosphars.2005.09.016>.
- Meyers, E., Kreiman, G., 2012. Tutorial on pattern classification in cell recording. *Vis. Popul. Codes*, 517–538.
- Milekovic, T., Fischer, J., Pistoohl, T., Ruescher, J., Schulze-Bonhage, A., Aertsen, A., Rickert, J., Ball, T., Mehring, C., 2012. An online brain–machine interface using

- decoding of movement direction from the human electrocorticogram. *J. Neural Eng.* 9, 046003. <http://dx.doi.org/10.1088/1741-2560/9/4/046003>.
- Miller, K.J., denNijs, M., Shenoy, P., Miller, J.W., Rao, R.P.N., Ojemann, J.G., 2007. Real-time functional brain mapping using electrocorticography. *Neuroimage* 37, 504–507. <http://dx.doi.org/10.1016/j.neuroimage.2007.05.029>.
- Miller, K.J., Schalk, G., Fetz, E.E., den Nijs, M., Ojemann, J.G., Rao, R.P., 2010. Cortical activity during motor execution, motor imagery, and imagery-based online feedback. *Proc. Natl. Acad. Sci.* 107, 4430–4435.
- Miller, K.J., Hermes, D., Honey, C.J., Hebb, A.O., Ramsey, N.F., Knight, R.T., Ojemann, J.G., Fetz, E.E., 2012. Human motor cortical activity is selectively phase-entrained on underlying rhythms. *PLoS Comput. Biol.* 8, e1002655. <http://dx.doi.org/10.1371/journal.pcbi.1002655>.
- Montemurro, M.A., Rasch, M.J., Murayama, Y., Logothetis, N.K., Panzeri, S., 2008. Phase-of-firing coding of natural visual stimuli in primary visual cortex. *Curr. Biol.* 18, 375–380. <http://dx.doi.org/10.1016/j.cub.2008.02.023>.
- Muthukumaraswamy, S.D., 2010. Functional properties of human primary motor cortex gamma oscillations. *J. Neurophysiol.* 104, 2873–2885.
- Muthukumaraswamy, S.D., 2011. Temporal dynamics of primary motor cortex gamma oscillation amplitude and piper corticomuscular coherence changes during motor control. *Exp. brain Res.* 212, 623–633.
- Nakhnikian, A., Ito, S., Dwiel, L.L., Grasse, L.M., Rebec, G.V., Lauridsen, L.N., Beggs, J.M., 2016. A novel cross-frequency coupling detection method using the generalized morse wavelets. *J. Neurosci. Methods* 269, 61–73. <http://dx.doi.org/10.1016/j.jneumeth.2016.04.019>.
- Newman, E.L., Gillet, S.N., Climer, J.R., Hasselmo, M.E., 2013. Cholinergic blockade reduces theta-gamma phase amplitude coupling and speed modulation of theta frequency consistent with behavioral effects on encoding. *J. Neurosci.* 33, 19635–19646. <http://dx.doi.org/10.1523/JNEUROSCI.2586-13.2013>.
- Ng, B.S.W., Logothetis, N.K., Kayser, C., 2013. EEG phase patterns reflect the selectivity of neural firing. *Cereb. Cortex* 23, 389–398. <http://dx.doi.org/10.1093/cercor/bhs031>.
- O'Keefe, J., Recce, M.L., 1993. Phase relationship between hippocampal place units and the EEG theta rhythm. *Hippocampus* 3, 317–330.
- Ojala, M., Garriga, G.C., 2010. Permutation tests for studying classifier performance. *J. Mach. Learn. Res.* 11, 1833–1863.
- Ossandon, T., Jerbi, K., Vidal, J.R., Bayle, D.J., Henaff, M.-A., Jung, J., Minotti, L., Bertrand, O., Kahane, P., Lachaux, J.-P., 2011. Transient suppression of broadband gamma power in the default-mode network is correlated with task complexity and subject performance. *J. Neurosci.* 31, 14521–14530. <http://dx.doi.org/10.1523/JNEUROSCI.2483-11.2011>.
- Ossandon, T., Vidal, J.R., Ciomas, C., Jerbi, K., Hamame, C.M., Dalal, S.S., Bertrand, O., Minotti, L., Kahane, P., Lachaux, J.-P., 2012. Efficient “pop-out” visual search elicits sustained broadband gamma activity in the dorsal attention network. *J. Neurosci.* 32, 3414–3421. <http://dx.doi.org/10.1523/JNEUROSCI.6048-11.2012>.
- Özkurt, T.E., 2012. Statistically reliable and fast direct estimation of phase-amplitude cross-frequency coupling. *Biomed. Eng. IEEE Trans.* 59, 1943–1950.
- Özkurt, T.E., Schnitzler, A., 2011. A critical note on the definition of phase–amplitude cross-frequency coupling. *J. Neurosci. Methods* 201, 438–443. <http://dx.doi.org/10.1016/j.jneumeth.2011.08.014>.
- Palva, S., Palva, J.M., 2007. New vistas for α-frequency band oscillations. *Trends Neurosci.* 30, 150–158. <http://dx.doi.org/10.1016/j.tins.2007.02.001>.
- Paus, T., 2001. Primate anterior cingulate cortex: where motor control, drive and cognition interface. *Nat. Rev. Neurosci.* 2, 417–424. <http://dx.doi.org/10.1038/35077500>.
- Pfurtscheller, M., Kujala, J., Vidal, J.R., Hamame, C.M., Ossandon, T., Bertrand, O., Minotti, L., Kahane, P., Jerbi, K., Lachaux, J.-P., 2012. How silent is silent reading? Intracerebral evidence for top-down activation of temporal voice areas during reading. *J. Neurosci.* 32, 17554–17562.
- Pfurtscheller, G., Lopes da Silva, F.H., 1999. Event-related EEG/MEG synchronization and desynchronization: basic principles. *Clin. Neurophysiol.* 110, 1842–1857.
- Pfurtscheller, G., Graimann, B., Huggins, J.E., Levine, S.P., Schuh, L.A., 2003. Spatiotemporal patterns of beta desynchronization and gamma synchronization in corticographic data during self-paced movement. *Clin. Neurophysiol.* 114, 1226–1236. [http://dx.doi.org/10.1016/S1388-2457\(03\)00067-1](http://dx.doi.org/10.1016/S1388-2457(03)00067-1).
- Pittman-Polletta, B., Hsieh, W.-H., Kaur, S., Lo, M.-T., Hu, K., 2014. Detecting phase-amplitude coupling with high frequency resolution using adaptive decompositions. *J. Neurosci. Methods* 226, 15–32. <http://dx.doi.org/10.1016/j.jneumeth.2014.01.006>.
- Ray, S., Maunsell, J.H., 2011. Different origins of gamma rhythm and high-gamma activity in macaque visual cortex. *PLoS Biol.* 9, e1000610.
- Rektor, I., Sochýrková, D., Bocková, M., 2006. Intracerebral ERD/ERS in voluntary movement and in cognitive visuomotor task. *Prog. Brain Res.* 159, 311–330. [http://dx.doi.org/10.1016/S0079-6123\(06\)59021-1](http://dx.doi.org/10.1016/S0079-6123(06)59021-1).
- Rickert, J., 2005. Encoding of movement direction in different frequency ranges of motor cortical local field potentials. *J. Neurosci.* 25, 8815–8824. <http://dx.doi.org/10.1016/j.jneurosci.2005.05.2005>.
- Roach, B.J., Mathalon, D.H., 2008. Event-related EEG time-frequency analysis: an overview of measures and an analysis of early gamma band phase locking in schizophrenia. *Schizophr. Bull.* 34, 907–926.
- Rowland, N.C., De Hemptinne, C., Swann, N.C., Qasim, S., Miocinovic, S., Ostrem, J.L.,
- Knight, R.T., Starr, P.A., 2015. Task-related activity in sensorimotor cortex in Parkinson's disease and essential tremor: changes in beta and gamma bands. *Front. Hum. Neurosci.* 9. <http://dx.doi.org/10.3389/fnhum.2015.00512>.
- Rushworth, M.F., Krams, M., Passingham, R.E., 2001. The attentional role of the left parietal cortex: the distinct lateralization and localization of motor attention in the human brain. *J. Cogn. Neurosci.* 13, 698–710. <http://dx.doi.org/10.1162/089892901750363244>.
- Saleh, M., Reimer, J., Penn, R., Ojakangas, C.L., Hatsopoulos, N.G., 2010. Fast and slow oscillations in human primary motor cortex predict oncoming behaviorally relevant cues. *Neuron* 65, 461–471. <http://dx.doi.org/10.1016/j.neuron.2010.02.001>.
- Sauseng, P., Klimesch, W., 2008. What does phase information of oscillatory brain activity tell us about cognitive processes? *Neurosci. Biobehav. Rev.* 32, 1001–1013. <http://dx.doi.org/10.1016/j.neubiorev.2008.03.014>.
- Schnitzler, A., Gross, J., 2005. Normal and pathological oscillatory communication in the brain. *Nat. Rev. Neurosci.* 6, 285–296. <http://dx.doi.org/10.1038/nrn1650>.
- Schwartz, A.B., 2016. Movement: how the brain communicates with the world. *Cell* 164, 1122–1135. <http://dx.doi.org/10.1016/j.cell.2016.02.038>.
- Sherman, M.T., Kanai, R., Seth, A.K., VanRullen, R., 2016. Rhythmic influence of top-down perceptual priors in the phase of prestimulus occipital alpha oscillations. *J. Cogn. Neurosci.*.
- Shibasaki, H., Hallett, M., 2006. What is the Bereitschaftspotential? *Clin. Neurophysiol.* 117, 2341–2356. <http://dx.doi.org/10.1016/j.clinph.2006.04.025>.
- Snyder, L.H., Batista, A.P., Andersen, R.A., 1997. Coding of intention in the posterior parietal cortex. *Nature* 386, 167–170. <http://dx.doi.org/10.1038/386167a0>.
- Soto, J.L., Jerbi, K., 2012. Investigation of cross-frequency phase-amplitude coupling in visuomotor networks using magnetoencephalography. In: Proceedings of the Engineering in Medicine and Biology Society (EMBC), Annual International Conference of the IEEE, pp. 1550–1553.
- Staresina, B.P., Bergmann, T.O., Bonnefond, M., van der Meij, R., Jensen, O., Deuker, L., Elger, C.E., Axmacher, N., Fell, J., 2015. Hierarchical nesting of slow oscillations, spindles and ripples in the human hippocampus during sleep. *Nat. Neurosci.*
- Stephan, K.M., Fink, G.R., Passingham, R.E., Silbersweig, D., Ceballos-Baumann, A.O., Frith, C.D., Frackowiak, R.S., 1995. Functional anatomy of the mental representation of upper extremity movements in healthy subjects. *J. Neurophysiol.* 73, 373–386.
- Talairach, J., Tournoux, P., 1993. Referentially oriented cerebral MRI anatomy: an atlas of stereotaxic anatomical correlations for gray and white matter. Thieme.
- Tallon-Baudry, C., Bertrand, O., Delpuech, C., Pernier, J., 1996. Stimulus specificity of phase-locked and non-phase-locked 40 Hz visual responses in human. *J. Neurosci.* 16, 4240–4249.
- Tort, A.B.L., Komorowski, R., Eichenbaum, H., Kopell, N., 2010. Measuring phase-amplitude coupling between neuronal oscillations of different frequencies. *J. Neurophysiol.* 104, 1195–1210. <http://dx.doi.org/10.1152/jn.00106.2010>.
- van der Meij, R., Kahana, M., Maris, E., 2012. Phase-amplitude coupling in human electrocorticography is spatially distributed and phase diverse. *J. Neurosci.* 32, 111–123. <http://dx.doi.org/10.1523/JNEUROSCI.4816-11.2012>.
- VanRullen, R., Busch, N., Drewes, J., Dubois, J., 2011. Ongoing EEG phase as a trial-by-trial predictor of perceptual and attentional variability. *Front. Psychol.* 2, 60.
- Vidal, J.R., Freyermuth, S., Jerbi, K., Hamamé, C.M., Ossandon, T., Bertrand, O., Minotti, L., Kahane, P., Berthoz, A., Lachaux, J.-P., 2012. Long-distance amplitude correlations in the high gamma band reveal segregation and integration within the reading network. *J. Neurosci.* 32, 6421–6434.
- Vidal, J.R., Perrone-Bertolotti, M., Levy, J., De Palma, L., Minotti, L., Kahane, P., Bertrand, O., Lutz, A., Jerbi, K., Lachaux, J.-P., 2014. Neural repetition suppression in ventral occipito-temporal cortex occurs during conscious and unconscious processing of frequent stimuli. *Neuroimage* 95, 129–135.
- Voytek, B., 2010. Shifts in gamma phase–amplitude coupling frequency from theta to alpha over posterior cortex during visual tasks. *Front. Hum. Neurosci.* 4. <http://dx.doi.org/10.3389/fnhum.2010.00191>.
- Voytek, B., D'Esposito, M., Crone, N., Knight, R.T., 2013. A method for event-related phase/amplitude coupling. *Neuroimage* 64, 416–424. <http://dx.doi.org/10.1016/j.neuroimage.2012.09.023>.
- Walder, S., Pistohl, T., Braun, C., Ball, T., Aertsen, A., Mehring, C., 2009. A review on directional information in neural signals for brain-machine interfaces. *J. Physiol. Paris* 103, 244–254. <http://dx.doi.org/10.1016/j.jphysparis.2009.08.007>.
- Walder, S., Preissl, H., Demandt, E., Braun, C., Birbaumer, N., Aertsen, A., Mehring, C., 2008. Hand movement direction decoded from MEG and EEG. *J. Neurosci.* 28, 1000–1008. <http://dx.doi.org/10.1523/JNEUROSCI.5171-07.2008>.
- Weaver, K.E., Wander, J.D., Ko, A.L., Casimo, K., Grabowski, T.J., Ojemann, J.G., Darvas, F., 2016. Directional patterns of cross frequency phase and amplitude coupling within the resting state mimic patterns of fMRI functional connectivity. *Neuroimage* 128, 238–251.
- Whittingstall, K., Logothetis, N.K., 2009. Frequency-band coupling in surface EEG reflects spiking activity in monkey visual cortex. *Neuron* 64, 281–289. <http://dx.doi.org/10.1016/j.neuron.2009.08.016>.
- Yanagisawa, T., Yamashita, O., Hirata, M., Kishima, H., Saitoh, Y., Goto, T., Yoshimine, T., Kamitani, Y., 2012. Regulation of motor representation by phase–amplitude coupling in the sensorimotor cortex. *J. Neurosci.* 32, 15467–15475. <http://dx.doi.org/10.1523/JNEUROSCI.2929-12.2012>.

Quatrième partie

Étude 3 : Décodage des directions de mouvement pendant et avant l'exécution de mouvement de membres supérieurs

INTRODUCTION

Mon introduction

Cinquième partie

**Étude 4 : Tensorpac, logiciel
Python de calcul de
Phase-Amplitude Coupling**

INTRODUCTION

Mon introduction

Tensorpac: an open-source Python toolbox for tensor-based Phase-Amplitude Coupling measurement

Etienne Combrisson^{1,2}, Juan LP Soto, Aymeric Guillot, Karim Jerbi^{1,2,3}

¹ Psychology department, Université de Montréal, Canada

² Lyon 1 University, Lyon, France

³ DYCOG Lab, Lyon Neuroscience Research Center, Lyon, France

Corresponding authors:

Etienne Combrisson (e.combrisson@gmail.com)

Abstract

We present [Tensorpac](#)¹, a Python open-source toolbox dedicated to the calculation of Phase-Amplitude Coupling (PAC). We designed a flexible modular implementation of existing PAC methods, chance distribution evaluation and normalization types in a way that it is much easier to compare those methods. In addition, all of those methods are implemented using tensors and parallel computing and hence, perform much better on multidimensional datasets.

Keywords: phase-amplitude coupling, pac, tensor, python, parallel computing

Declarations

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflict of interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

¹ <https://github.com/EtienneCmb/tensorpac>

Introduction

For decades, neuroscientists try isolate task-related changes in brain oscillations by extracting spectral features such as power, amplitude or phase. Those changes are then used to understand brain behaviors and physiological involvements or in a Brain-computer Interface (BCI) to train the machine to recognized micro-events and then, associate a command to it. For example, hand motions imply a beta power desynchronization and a gamma synchronization in the sensorimotor cortex (Crone, 1998; Crone et al., 1998). Those rhythms in specific frequency bands can also interact together (Jensen and Colgin, 2007) and this is called Cross-Frequency Coupling (CFC). Publications report cross-frequency phase-phase coupling or phase synchronization (Belluscio et al., 2012; Darvas et al., 2009; Voytek, 2010), amplitude-amplitude coupling (Friston, 1997; Shirvalkar et al., 2010; Siegel et al., 2009) and Phase-Amplitude Coupling (PAC). The latter involves the synchronization between the phase of low-frequency oscillations and the amplitude of high-frequency oscillations. Over the last decade, the PAC receive a growing interest in a large variety of task (Bruns and Eckhorn, 2004; Cohen et al., 2008; Hemtinne et al., 2013; Lee and Jeong, 2013; Lega et al., 2016; Maris et al., 2011; Newman et al., 2013; Soto and Jerbi, 2012; van der Meij et al., 2012; Watrous et al., 2015; Weaver et al., 2016; Yanagisawa et al., 2012). While the role and the mechanism of the PAC remain elusive (Canolty and Knight, 2010), a relatively large number of methodologies have been proposed (Canolty et al., 2006; Cohen, 2008; Dvorak and Fenton, 2014; Lakatos, 2005; Nakhnikian et al., 2016; Ozkurt, 2012; Penny et al., 2008; Tort et al., 2010; Watrous et al., 2015) and compared (Penny et al., 2008; Tort et al., 2010). There is still no gold standard for the choice of the method as their performances in coupling detection depends on signal processing tools and data properties such as length, noise and amount of coupling. Furthermore, CFC can also be detected in absence of coupling (Aru et al., 2015). Assessing the PAC can be done in four steps : extract the phase and the amplitude, measure the degree of coupling between them, then generate a chance distribution and finally, correct the PAC with this chance distribution to minimize non-related PAC events. For those three steps, each publication provide a particular method which decrease the potential reproducibility. There is a few number of existing toolbox to compute PAC such as [Fieldtrip](#)² (Oostenveld et al., 2011) and [PACT](#)³ for EEGLAB (Delorme and Makeig, 2004) in Matlab or [pacpy](#)⁴ developed by for Voytek's research team in Python. Those toolbox re-implement existing methods but the implementation could be improved for large datasets as computing PAC can be very long, especially for the correction part. In this context, we developed an Python open-source toolbox called [Tensorpac](#)⁵, distributed under a BSD licence, for computing Phase-Amplitude Coupling using tensors and parallel computing. In addition, each step for assessing this coupling measure has been splitted so that the user can pick methods of his choice and then, have the most adapted combination considering data properties.

² http://www.fieldtriptoolbox.org/example/cross_frequency_analysis

³ <https://sccn.ucsd.edu/wiki/PACT>

⁴ <https://github.com/voytekresearch/pacpy>

⁵ <https://github.com/EtienneCmb/tensorpac>

Materials and Methods

As the PAC name suggest, it consist of measuring how the phase of slow waves are coupled with the amplitude higher frequency signal. This coupling measure is bidirectional i.e. is not possible to say if it is slow waves which lead amplitude rhythms or the opposite. That said, we denote by $f1 \leftrightarrow f2$ the PAC between a phase centered in $f1$ and the amplitude centered in $f2$.

At this point it is important to note that the PAC is usually computed using the phase and amplitude coming respectively from the same signal but it is also possible to compute distant PAC.

Synthetic signals

First of all, for the implementation and validation of coupling methods, we needed a signal with a controllable coupling frequencies. To this end, we included in the toolbox a *PacSignals* function that reproduce synthetic signals proposed by **Tort et al. 2010**.

- The coupling frequency pair of (phase, amplitude)
- The amount of coupling
- The amount of noise
- Data length and sampling frequency

To those controls, we add the possibility to generate multidimensional datasets and add an inter-trial variability variable. An example of such signals is shown in **Figure 1** and we also provide the code in the **Code snippet 1**.

PAC calculation procedure

Non-corrected PAC from extracted phase and amplitude

As shown in **Figure 2**, The first step is to extract the phase and the amplitude. This can be assessed either by filtering then taking the Hilbert transform of the filtered signals or using wavelets. Tensorac offers both possibilities and provides least squared filtered with a Python adaptation of EEGLAB (**Delorme and Makeig, 2004**), Butterworth or Bessel filters and morlet's wavelets (**Tallon-Baudry et al., 1996**). The phase and the amplitude are respectively obtained by taking the angle and modulus of complex decompositions provided by Hilbert transform or wavelets. Importantly, bandpass filtering can occurs frequency dependant phase shiftings and potentially destroy coupling. From a programming perspective, this is easily solved by using a forward high-pass filter the a backward low-pass filter and compensate delays. Finally, the PAC is computed using of the existing methodologies (Mean Vector Length, Kullback-Leibler Distance...). Tensorpac use this two-ways filter such as the recommended cycle number for the phase and amplitude filtering (**Bahramisharif et al., 2013**).

Chance distribution and PAC correction

As described by **Tort et al. 2010**, the absence of PAC in a signal could be related to several parameters. First, filtering artefacts could be observed. Then, each one of the proposed PAC methodologies present some advantages or limitations and might be not appropriate for all type of analysis. Those methods present more or less robustness to noise, to modulation width are could be not amplitude independent (**Tort et al., 2010**). In addition, PAC estimations might be biased by the length of data and longer epochs generally lead to a more trustful PAC.

Taken together, those limitations could be minimized by computing a chance distribution and then, by correcting the PAC value. To this end, several methods exist but all share this same idea as shown in **Figure 3** : introduce a small change in data such that PAC properties are conserved but this change should only reflect events that could happened by chance or, more generally, on any type of signals. Among those existing methods, **Canolty et al. 2006** introduce a time lag to the amplitude, while **Tort et al. 2010** swap amplitude and phase trials and **Bahramisharif et al., 2013** swap time blocks. Finally, the PAC estimation is corrected using the mean and sometimes the deviation of surrogates (see **Figure 4**). The code for computing the comodulogram on multidimensional data is provided in the **Code snippet 2**.

Modular implementation of existing methods

By setting aside the extraction of the phase and the amplitude, three steps are sufficient to compute the Phase-Amplitude Coupling :

1. Compute the non-corrected PAC
2. If needed, compute the chance distribution
3. Correct the PAC by the surrogates

With Tensorpac we propose a modular implementation of existing PAC and surrogates evaluation. When defining a *Pac* instance, we provide an *idpac* variable which consist of three integers each one respectively referring to the Pac method, to the surrogate method and how to normalize the PAC. Currently supported methods are presented in **Table 1**.

First digit : PAC methods	Second digit : Surrogate methods	Third digit : Normalization
1 - Mean Vector Length (MVL - <i>Canolty, 2006</i>)	0 - No surrogates	0 - No normalization
2 - Kullback-Leibler Distance (KLD - <i>Tort, 2010</i>)	1 - Swap phase/amplitude trials (<i>Tort, 2010</i>)	1 - $PAC - m$
3 - Height-ratio (HR - <i>Lakatos, 2005</i>)	2 - Swap amplitude time blocks (<i>Bahramisharif, 2013</i>)	2 - PAC / m
4 - Normalized Direct PAC (ndPAC - <i>Ozkurt, 2012</i>)	3 - Shuffle amplitude time-series	3 - $(PAC - m) / m$
5 - Phase Synchrony (PS - <i>Cohen, 2008</i>)	4 - Time-lag (<i>Canolty, 2006</i>)	4 - $(PAC - m) / std$

Table1 : implemented methods in Tensorpac toolbox. The *idpac* variable is a tuple of three integers referring to (PAC method, Surrogate method, Normalization). We denote by *PAC* the non-corrected coupling, *m* and *std* being respectively the mean and deviation of the chance distribution.

Note that the ndPAC include a statistical estimation and so surrogate evaluation is systematically ignored using it. Then, the Phase Synchrony (**Cohen et al. 2008**) is an adaptation of the Phase Locking Value (PLV) where the coupling is obtained by taking the phase of a second Hilbert transform of the amplitude of fastest oscillations.

Tensor implementation and parallel computing combination

Formulas of PAC in publications are always imply time series i.e. one dimensional signals. Hence, computing the PAC on several signals and in several frequency bands, such as a comodulogram, demand embedded loops. While C code is efficient with loops, higher-levels languages such as Python or Matlab are considerably slower and this is a huge limitation for computing coupling on a large amount of subjects/electrodes/trials.

That said, we adapted each methodology to be computed using tensor with a contraction over the time axis. This implementation type have two major benefits:

1. Even on smaller datasets the execution time is faster using tensors. Note that this difference in execution time is then propagate when computing surrogates and the gain of time increase.
2. Using tensors, loops are avoided and there is no restriction on data shape as soon as the time axis location is provided.

In addition to this tensor implementation two steps of the PAC evaluation can be processed in parallel with a control of the number of cores to use :

1. Extracting phase/amplitude in multiple frequency bands
2. Computing surrogates

Depending on the number of cores and the available memory, the tensor implementation and parallel computing can both drastically decrease the computing time.

Results

Methods comparison

Comparison of PAC methods

In order to compare PAC methods, we generated 100 signals each one containing a 10 \leftrightarrow 100 hz phase-amplitude coupling. Then, we extracted the phase and amplitude from each signal. Finally, we computed the comodulogram on each signal and for each methodology and the final picture represent the mean over generated comodulograms. The result is presented in **Figure 5**. First, MVL and ndPac shared a similar methodology with the exception that the ndPac also include a statistical test that improve coupling localisation. The PS (also called *adapted PLV*) correctly identify the coupling but seems to be sensible to noise. Finally, KLD and HR provide very similar results as expected but, for shorter epochs, might present additional noise in slower frequencies.

Comparison of surrogate methods

As explained above, the PAC validity can be compromised in presence of noise or a low coupling degree, too short epochs or filtering artefacts. In **Figure 6**, we show an example of normalized PAC, using the MVL and compare how those procedure performs to retrieve the original 10 \leftrightarrow 100 hz coupling among the 100 generated trials. First, all of the implemented surrogate methods enable to retrieve the coupling with approximately the same acuity. One thing should noticed, the swapping proposed by **Tort et al. 2010** requires to precise the location of the axis dimension while other methods can be performed on a single trial.

Computing time comparison

One remaining question is how perform the tensor combine with the parallel computing implementation compare to the usual time-series one. In **Figure 7**, we show the execution time ratio of t_{Loop}/t_{Tensor} for an increasing number of datasets (A) and frequency pairs (phase, amplitude) (B). The KLD and HR PAC methods are those benefits most and for 150 signals the tensor implementation is 30 times faster (A). Other methods also performs better with tensor but the difference is smaller. The reason for such a difference is that both of the KLD and HR needs a binned amplitude according to phase values. This process slow down performances but it can be assessed with multidimensional arrays. For an increasing number of frequencies as this is the case of a comodulogram, the difference between the tensor and loop implementations follow a linear increase with a slightly different slope depending on the methods. The results for computing a non-corrected comodulogram of 100 signals using 14 phases and 13 amplitudes (182 frequency pairs in total) is presented in **Table 1**. For this particular case, tensorpac performs is in mean 38 times faster compared across methodologies compared to the basic time-series forms with a maximum for the KLD and HR which are 50 times faster.

	Execution time (loop)	Execution time (tensor)	Execution time ratio loop/tensor
Mean Vector Length	3,72	0,16	23,73
Kullback-Leibler divergence	35,83	0,65	55,03
Heights ratio	32,77	0,65	50,32
ndPac	6,60	0,17	39,19
Phase-Synchrony	5,12	0,23	22,23
Mean	16,81	0,37	38,10

Table 1 : execution time (in seconds) comparison between the loop and tensor implementations for computing the comodulogram of 100 signals. Note that this implementation only compare non-corrected PAC and hence, does not include the calcul of surrogates.

Even on small datasets as in **Table 1**, there is an important difference in computing time. For noisy data the PAC needs to be corrected and to this end, compute surrogates. Those variations in computing time are then propagated and as expected, the difference between the tensor and loop implementation is even larger as shown in **Table 2**. Computing a corrected comodulogram with Tensorpac and with only 100 surrogates is now closed to 90 times faster with the KLD and HR around 120 faster.

	Execution time (loop)	Execution time (tensor)	Execution time ratio loop/tensor
Mean Vector Length	551,65	9,90	55,74
Kullback-Leibler divergence	3779,93	30,64	123,38
Heights ratio	3475,99	30,49	114,01
Phase-Synchrony	688,90	12,14	56,75
Mean	2124,12	20,79	87,47

Table 2 : execution time (in seconds) comparison between the loop and tensor implementations for computing the corrected (using 100 permutations) comodulogram of 100 signals.

Discussion

In this paper we present an open-source Python toolbox with a modular and tensor based implementation of Phase-Amplitude coupling measures. The interest of such a toolbox is 3 folds:

1. The modular implementation is not restrictive and let to the user the possibility to combine existing methods. In addition, new existing methods can easily be incorporate in the toolbox.
2. We used NumPy tensor functions that first, considerably decrease the execution time and secondly are more appropriate to multidimensional datasets.
3. The toolbox also include a documentation, examples, several filtering and complex decomposition fonctions, utility tools such as generating synthetic coupled signals. Tensorpac also includes statistics and extended plotting functions based on Matplotlib

We found a lack of existing efficient PAC softwares, and, because all neuroscientists are not methodologist, Tensorpac is an appropriate solution to the recent Phase-Amplitude Coupling demands and we also invite other laboratories to provide a source code of their methods.

Future directions

Currently, Tensorpac includes 5 PAC methods and it would be interesting to add old and new ones (such as Amplitude PSD ([Cohen, 2008](#)), Correlation measure ([Penny et al., 2008](#)), Generalized Linear Model ([Penny et al., 2008](#)), coherence ([Colgin et al., 2009](#)) or using generalized Morse wavelets ([Nakhnikian et al., 2016](#))). All the presented measures are computed across time, which limit short event investigations. To this end, [Voytek et al., 2013](#), proposed the Event Related PAC (ERPAC) which examine the consistency across trials hence, conserve the time dimension. We started to incorporate the ERPAC but the tensor conversion needs further investigations.

Conclusion

Tensorpac is a promising toolbox for a fast and reliable estimation of Phase-Amplitude Coupling. The modular implementation considerably increase the number of possibilities but is also more suited to fit to the variety of data. The tensor and parallel computing combination drastically decrease the needed execution time which reinforce the probability to find the best combination, if such a combination exist.

Figures

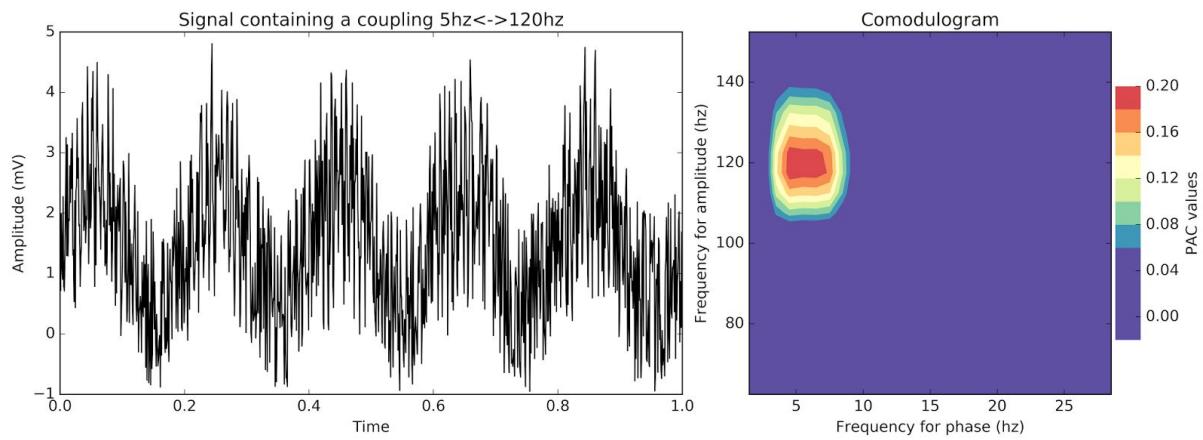


Figure 1 : example of a signal containing a 5<->120 hz coupling and corresponding comodulogram.

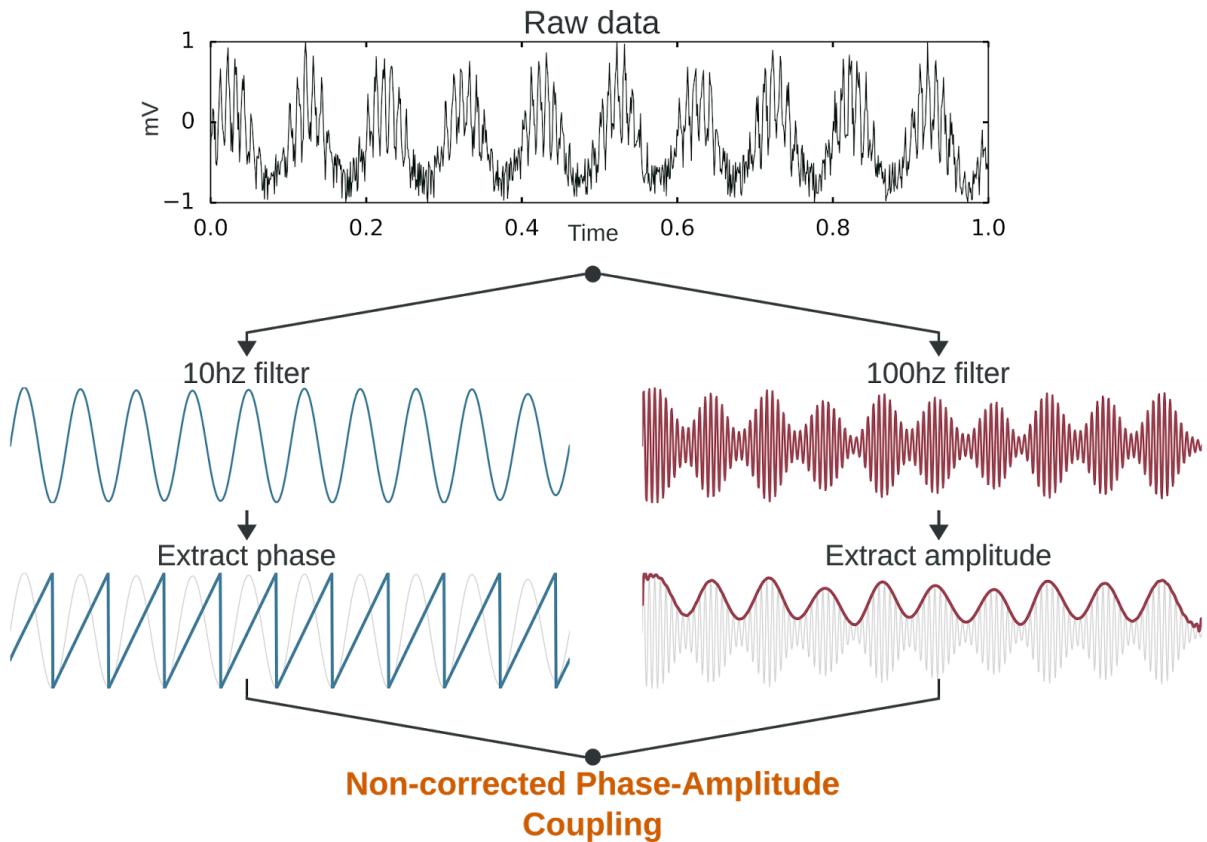


Figure 2 : estimation process of non-corrected 5<->100 hz. For the sake of the illustration, the raw data contains a coupling between a 10 hz phase and a 100 hz amplitude. First, the raw data is respectively filtered with frequencies centered on 10 hz and 100 hz. Then, each signals are passed to the complex domain using a Hilbert transform and on the first, only the phase is kept and the amplitude and the second. Finally, the PAC is obtained, from this phase and amplitude signals using of the existing measure.

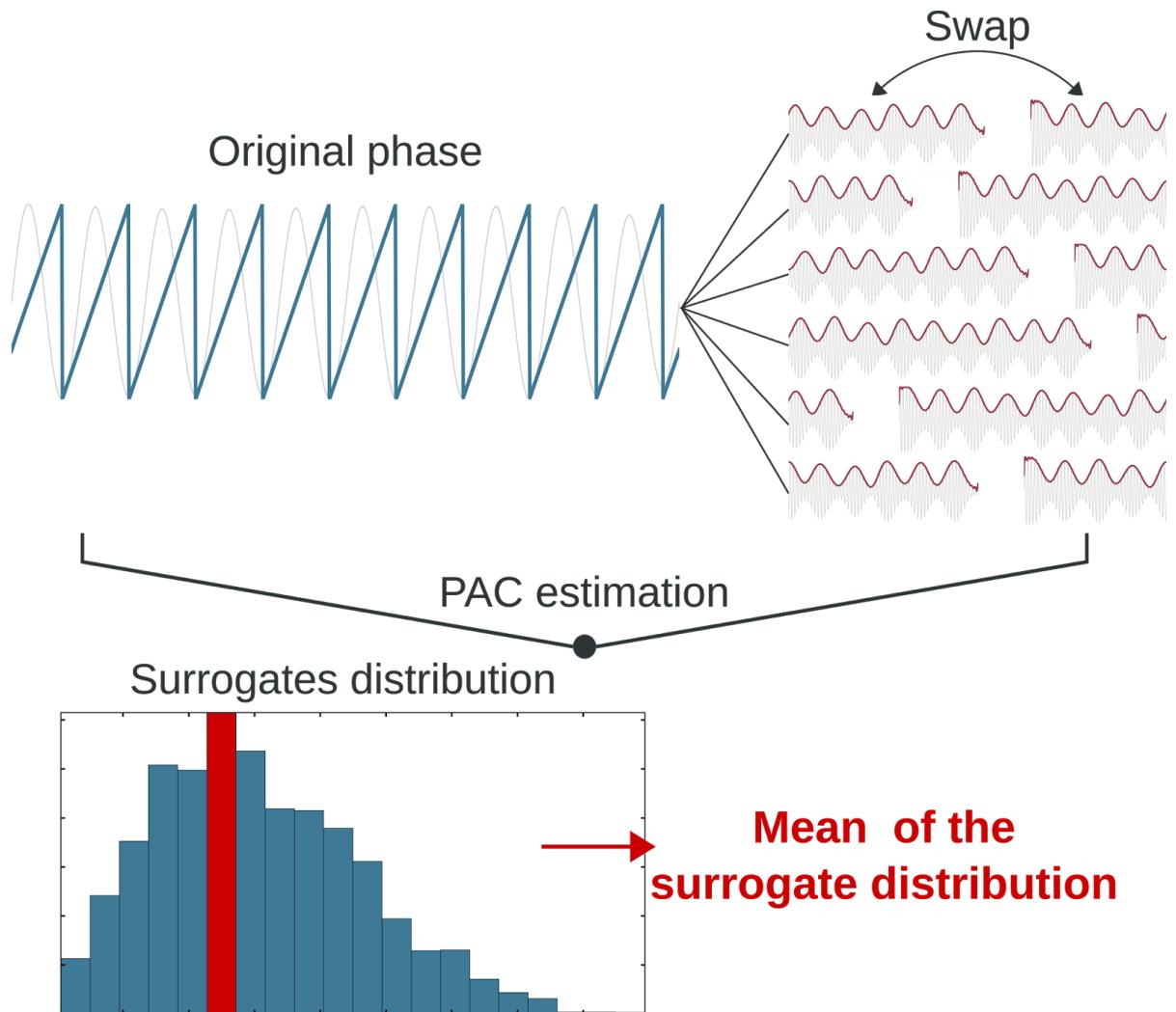


Figure 3 : example of surrogate distribution estimation by randomly swapping amplitude blocks. The amplitude is cut in half at a random time point and the two blocks are swapped. Then, the PAC measure is estimated using this swapped version of amplitude and the originally extracted phase. The distribution of surrogates is obtained by putting this process into a loop and varying the random cutting point.

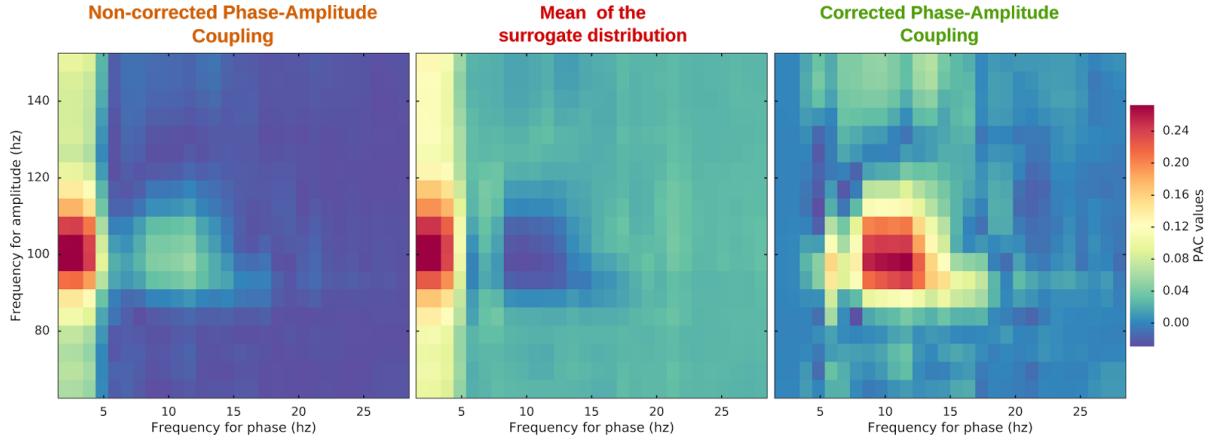


Figure 4 : example of PAC correction. First, the PAC is computed for several (phase, amplitude) pairs. Then, for each of those pairs, we estimate the distribution of surrogates. Both of the non-corrected PAC and surrogates shared a peak between the very low frequency phase and the 100hz amplitude. The 10->100 hz coupling is finally retrieved by subtracting the mean of the surrogate distribution to the non-corrected PAC.

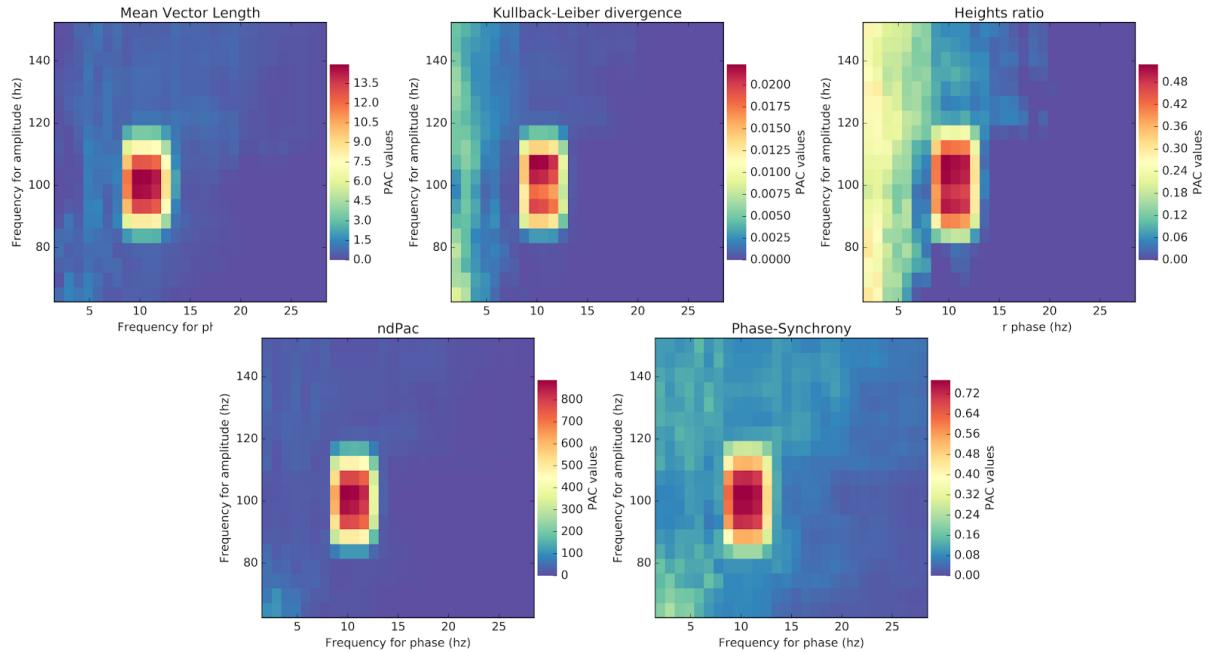


Figure 5 : comparison of implemented PAC methods. 100 10 <-> 100 hz synthetic coupled signals are generated. Then, we computed the comodulogram of such signal using the MVL, KLD, HR, ndPac, PS.

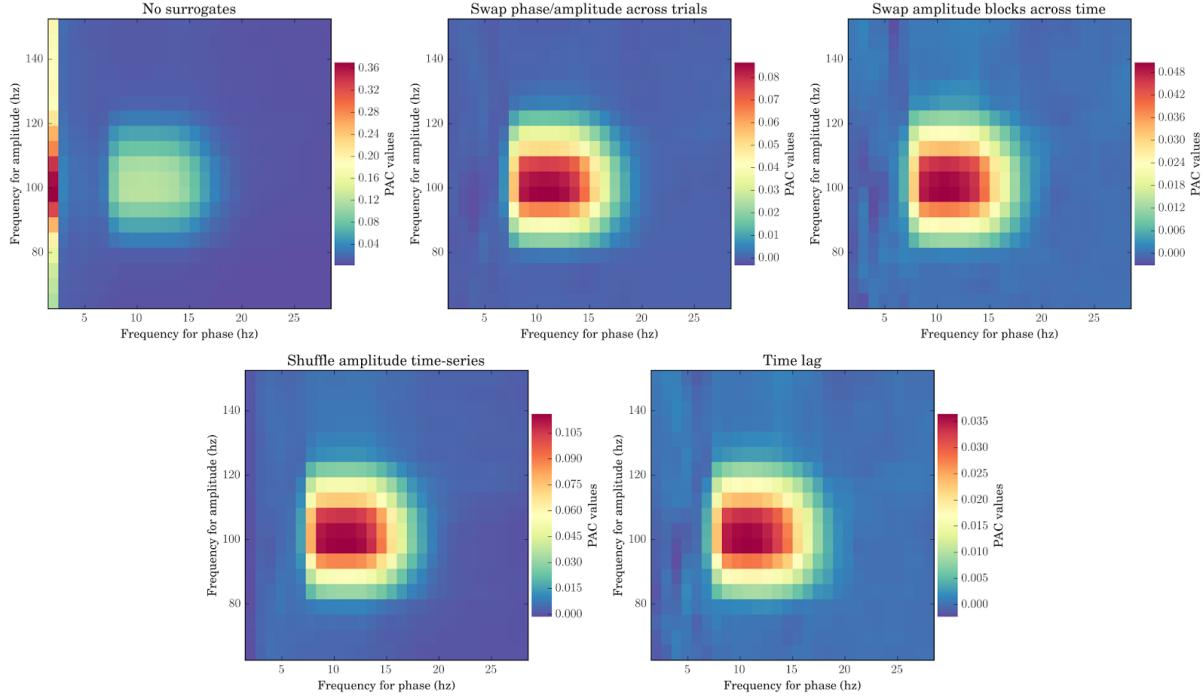


Figure 6 : comparison of implemented surrogate methods on 100 synthetic 10 \leftrightarrow 100 hz coupled signals using the MVL as a primary PAC evaluation method. This figure illustrate first non-corrected PAC then corrected PAC by swapping trials, swapping time blocks, shuffling amplitude time series and finally introducing a random time lag. Finally, the mean of surrogates is subtracted to the non-corrected PAC.

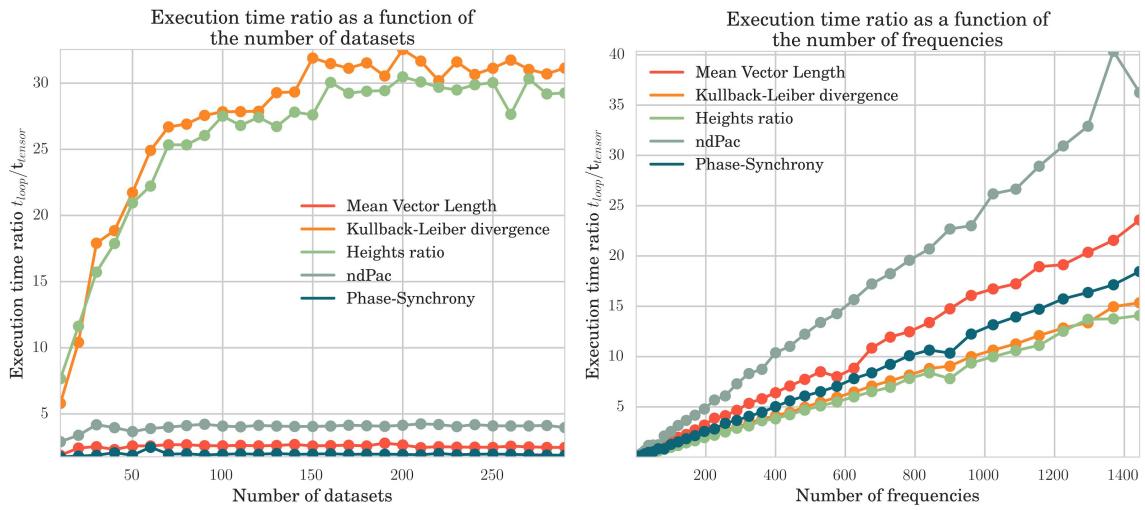


Figure 7 : execution time comparison between the tensor and loop implementations for (A) an increasing number of signals and (B) an increasing number of frequencies.

Code snippets

```
from tensorpac.utils import PacSignals

# Generate 100 signals with a 5<->120 hz coupling and a relatively large amount of noise :
data, time = PacSignals(ndatasets=100, fpha=5, famp=120, noise=3)
```

Code snippet 1 : generate synthetic signals containing a 5<->120 hz PAC coupling.

```
import matplotlib.pyplot as plt

from tensorpac.utils import PacSignals
from tensorpac import Pac

# Generate 100 signals with a 5<->120 hz coupling and a relatively large amount
# of noise. The data have a shape of (n_signals, n_points)
sf = 1024. # Sampling frequency
data, time = PacSignals(fpha=10, famp=100, noise=1, ndatasets=100, tmax=3,
                       dpha=10, damp=10, sf=sf)

# Define a Pac instance using the MVL. Then the chance distribution will be
# generated by randomly swap amplitude time blocks. Finally, the PAC is
# corrected by subtracting the mean of surrogates :
p = Pac(idpac=(1, 2, 1), fpha=(1, 30, 1, 1), famp=(60, 160, 5, 5),
        dcomplex='wavelet')

# Extract (phase, amplitude) using wavelets and compute corrected-PAC
# using 100 permutations :
pac, pvalue = p.filterfit(1024, data, data, axis=1, nperm=100)

# Finally, plot the comodulogram of the mean across trials :
p.comodulogram(pac.mean(-1), title='Comodulogram', cmap='Spectral_r')

plt.show()
```

Code snippet 2 : compute the comodulogram of 100 signals of 3 seconds length. Phases and amplitudes are extracted using Morlet's wavelets. Then, the PAC is computed using the MVL and is then corrected by swapping amplitude time blocks 100 times and subtracting the mean of surrogates.

References

- Aru, J., Aru, J., Priesemann, V., Wibral, M., Lana, L., Pipa, G., et al. (2015). Untangling cross-frequency coupling in neuroscience. *Current Opinion in Neurobiology* 31, 51–61. doi:10.1016/j.conb.2014.08.002.
- Bahramisharif, A., van Gerven, M. A. J., Aarnoutse, E. J., Mercier, M. R., Schwartz, T. H., Foxe, J. J., et al. (2013). Propagating Neocortical Gamma Bursts Are Coordinated by Traveling Alpha Waves. *Journal of Neuroscience* 33, 18849–18854. doi:10.1523/JNEUROSCI.2455-13.2013.
- Belluscio, M. A., Mizuseki, K., Schmidt, R., Kempter, R., and Buzsáki, G. (2012). Cross-frequency phase–phase coupling between theta and gamma oscillations in the hippocampus. *Journal of Neuroscience* 32, 423–435.
- Brunn, A., and Eckhorn, R. (2004). Task-related coupling from high-to low-frequency signals among visual cortical areas in human subdural recordings. *International Journal of Psychophysiology* 51, 97–116.
- Canolty, R. T., Edwards, E., Dalal, S. S., Soltani, M., Nagarajan, S. S., Kirsch, H. E., et al. (2006). High Gamma Power Is Phase-Locked to Theta Oscillations in Human Neocortex. *Science* 313, 1626–1628. doi:10.1126/science.1128115.
- Canolty, R. T., and Knight, R. T. (2010). The functional role of cross-frequency coupling. Available at: https://www.researchgate.net/profile/Ryan_Canolty/publication/47382016_The_functional_role_of_cross-frequency_coupling/links/548651f00cf2ef34478bf713.pdf [Accessed March 18, 2016].
- Cohen, M. X. (2008). Assessing transient cross-frequency coupling in EEG data. *Journal of Neuroscience Methods* 168, 494–499. doi:10.1016/j.jneumeth.2007.10.012.
- Cohen, M. X., Elger, C. E., and Fell, J. (2008). Oscillatory activity and phase–amplitude coupling in the human medial frontal cortex during decision making. *Journal of cognitive neuroscience* 21, 390–402.
- Colgin, L. L., Denninger, T., Fyhn, M., Hafting, T., Bonnevie, T., Jensen, O., et al. (2009). Frequency of gamma oscillations routes flow of information in the hippocampus. *Nature* 462, 353–357.

- Crone, N. (1998). Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis. I. Alpha and beta event-related desynchronization. *Brain* 121, 2271–2299. doi:10.1093/brain/121.12.2271.
- Crone, N. E., Miglioretti, D. L., Gordon, B., and Lesser, R. P. (1998). Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis. II. Event-related synchronization in the gamma band. *Brain* 121, 2301–2315.
- Darvas, F., Miller, K. J., Rao, R. P., and Ojemann, J. G. (2009). Nonlinear phase–phase cross-frequency coupling mediates communication between distant sites in human neocortex. *Journal of Neuroscience* 29, 426–435.
- Delorme, A., and Makeig, S. (2004). EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of neuroscience methods* 134, 9–21.
- Dvorak, D., and Fenton, A. A. (2014). Toward a proper estimation of phase–amplitude coupling in neural oscillations. *Journal of Neuroscience Methods* 225, 42–56. doi:10.1016/j.jneumeth.2014.01.002.
- Friston, K. J. (1997). Another neural code? *Neuroimage* 5, 213–220.
- Hempinne, C. de, Ryapolova-Webb, E. S., Air, E. L., Garcia, P. A., Miller, K. J., Ojemann, J. G., et al. (2013). Exaggerated phase–amplitude coupling in the primary motor cortex in Parkinson disease. *PNAS*. doi:10.1073/pnas.1214546110.
- Jensen, O., and Colgin, L. L. (2007). Cross-frequency coupling between neuronal oscillations. *Trends in cognitive sciences* 11, 267–269.
- Lakatos, P. (2005). An Oscillatory Hierarchy Controlling Neuronal Excitability and Stimulus Processing in the Auditory Cortex. *Journal of Neurophysiology* 94, 1904–1911. doi:10.1152/jn.00263.2005.
- Lee, J., and Jeong, J. (2013). Correlation of risk-taking propensity with cross-frequency phase–amplitude coupling in the resting EEG. *Clinical Neurophysiology*. doi:10.1016/j.clinph.2013.05.007.
- Lega, B., Burke, J., Jacobs, J., and Kahana, M. J. (2016). Slow-Theta-to-Gamma Phase–Amplitude Coupling in Human Hippocampus Supports the Formation of New Episodic Memories. *Cerebral Cortex* 26, 268–278. doi:10.1093/cercor/bhu232.

Maris, E., van Vugt, M., and Kahana, M. (2011). Spatially distributed patterns of oscillatory coupling between high-frequency amplitudes and low-frequency phases in human iEEG. *Neuroimage* 54, 836–850.

Nakhnikian, A., Ito, S., Dwiel, L. L., Grasse, L. M., Rebec, G. V., Lauridsen, L. N., et al. (2016). A novel cross-frequency coupling detection method using the generalized Morse wavelets. *Journal of Neuroscience Methods* 269, 61–73. doi:10.1016/j.jneumeth.2016.04.019.

Newman, E. L., Gillet, S. N., Climer, J. R., and Hasselmo, M. E. (2013). Cholinergic Blockade Reduces Theta-Gamma Phase Amplitude Coupling and Speed Modulation of Theta Frequency Consistent with Behavioral Effects on Encoding. *Journal of Neuroscience* 33, 19635–19646. doi:10.1523/JNEUROSCI.2586-13.2013.

Ozkurt, T. E. (2012). Statistically Reliable and Fast Direct Estimation of Phase-Amplitude Cross-Frequency Coupling. *Biomedical Engineering, IEEE Transactions on* 59, 1943–1950.

Penny, W. D., Duzel, E., Miller, K. J., and Ojemann, J. G. (2008). Testing for nested oscillation. *Journal of Neuroscience Methods* 174, 50–61. doi:10.1016/j.jneumeth.2008.06.035.

Shirvalkar, P. R., Rapp, P. R., and Shapiro, M. L. (2010). Bidirectional changes to hippocampal theta-gamma comodulation predict memory for recent spatial episodes. *PNAS* 107, 7054–7059. doi:10.1073/pnas.0911184107.

Siegel, M., Warden, M. R., and Miller, E. K. (2009). Phase-dependent neuronal coding of objects in short-term memory. *PNAS* 106, 21341–21346. doi:10.1073/pnas.0908193106.

Soto, J. L., and Jerbi, K. (2012). Investigation of cross-frequency phase-amplitude coupling in visuomotor networks using magnetoencephalography. in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE* (IEEE), 1550–1553. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6346238 [Accessed March 18, 2016].

Tallon-Baudry, C., Bertrand, O., Delpuech, C., and Pernier, J. (1996). Stimulus specificity of phase-locked and non-phase-locked 40 Hz visual responses in human. *The Journal of Neuroscience* 16, 4240–4249.

- Tort, A. B. L., Komorowski, R., Eichenbaum, H., and Kopell, N. (2010). Measuring Phase-Amplitude Coupling Between Neuronal Oscillations of Different Frequencies. *Journal of Neurophysiology* 104, 1195–1210. doi:10.1152/jn.00106.2010.
- van der Meij, R., Kahana, M., and Maris, E. (2012). Phase-Amplitude Coupling in Human Electrocorticography Is Spatially Distributed and Phase Diverse. *Journal of Neuroscience* 32, 111–123. doi:10.1523/JNEUROSCI.4816-11.2012.
- Voytek, B. (2010). Shifts in gamma phase–amplitude coupling frequency from theta to alpha over posterior cortex during visual tasks. *Frontiers in Human Neuroscience* 4. doi:10.3389/fnhum.2010.00191.
- Voytek, B., D'Esposito, M., Crone, N., and Knight, R. T. (2013). A method for event-related phase/amplitude coupling. *NeuroImage* 64, 416–424. doi:10.1016/j.neuroimage.2012.09.023.
- Watrous, A. J., Deuker, L., Fell, J., and Axmacher, N. (2015). Phase-amplitude coupling supports phase coding in human ECoG. *eLife* 4, e07886.
- Weaver, K. E., Wander, J. D., Ko, A. L., Casimo, K., Grabowski, T. J., Ojemann, J. G., et al. (2016). Directional patterns of cross frequency phase and amplitude coupling within the resting state mimic patterns of fMRI functional connectivity. *NeuroImage* 128, 238–251.
- Yanagisawa, T., Yamashita, O., Hirata, M., Kishima, H., Saitoh, Y., Goto, T., et al. (2012). Regulation of Motor Representation by Phase–Amplitude Coupling in the Sensorimotor Cortex. *J. Neurosci.* 32, 15467–15475. doi:10.1523/JNEUROSCI.2929-12.2012.

Sixième partie

Étude 5 : Visbrain, ensemble d'outils de visualisation de données neuroscientifiques

INTRODUCTION

Mon introduction

Visbrain : hardware-accelerated visualization suite for neuroscientific data in Python

Etienne Combrisson^{1,2}, [...], Aymeric Guillot, Karim Jerbi^{1,2,3}

¹ Psychology department, Université de Montréal, Canada

² Lyon 1 University, Lyon, France

³ DYCOG Lab, Lyon Neuroscience Research Center, Lyon, France

Corresponding authors:

Etienne Combrisson (e.combrisson@gmail.com)

Abstract	3
Introduction	4
Materials and Methods	5
Programming language and code guidelines	5
Graphics on the highway	5
Graphical interface and user interactions	6
Documentation and examples	6
Results	7
Ndviz : efficient data mining	7
Dispose your data into a grid	7
Plotting forms	7
Brain : visualization on a standard 3D MNI brain	8
MNI brains	8
Region of interest	8
Deep sources	8
Connectivity	9
High-definition screenshots	9
Class methods for a code-line control	9
Sleep : polysomnographic data visualization and edition	10
Data visualization	10
Hypnogram visualization and scoring	10
Automatic events detection and signal processing tools	11
GUI commodities	11
Figure : page layout of publication-ready complex figures	11
Discussion	12
Future directions	12
Conclusion	13
Figures	14
Code snippets	21
References	23

Abstract

We present a Python open-source package called Visbrain which proposes a coherent visualization suite for neuroscientific data. The current version of Visbrain is articulated around four modules respectively dedicated to 1) data mining and basic plotting functions (*NdViz* module), 2) visualizations on a 3D standard MNI brain (*Brain* module), 3) polysomnographic data visualization and sleep analysis (*Sleep* module) and finally, 3) a module for page layout and export of high-quality figures (*Figure* module). The three first modules come with modular and powerful graphical user interfaces built with PyQt. Each module has been developed in collaboration with neuroscientists and specialists of the field and provides therefore a coherent and exhaustive set of functionalities. Visbrain is developed on top of VisPy which provides high performance graphics by offloading rendering to the graphic card. This package is available on [Github](#)¹ and comes with an extensive [documentation](#)² and examples datasets.

Keywords: visualization, neuroscience, python, open-source, brain, MNI, GPU, opengl, sleep, data mining

Declarations

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflict of interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

¹ <https://github.com/EtienneCmb/visbrain>

² <http://etiennecmb.github.io/visbrain/>

Introduction

Generally, scientific visualizations are used to illustrate complex results and make the data speak. For publication, those illustrations must be clear, concise and understanding at the first glance. In practice, it is not always simple to handle very large datasets, especially with the dimension problem. Publication figures are statics, definitive and concentrate hundreds or thousands of pre-figures that have been upstream used to understand the data. Neuroscience is no exception and the several research fields imply a large range of possible representations. Moreover, this range is further expanded when account is taken of the diversity of neural recordings types.

To date, Matlab represents one of the most widely-used software for brain data analysis and visualization, notably thanks to several toolboxes such as SPM, [Brainstorm³](#) (Tadel et al., 2011), [EEGlab⁴](#) (Delorme and Makeig, 2004) or [Fieldtrip⁵](#) (Oostenveld et al., 2011). On the other hand, only few Python alternatives exist so far, among which are [MNE⁶](#) (Gramfort et al., 2013), [PySurfer⁷](#) or [Nilearn⁸](#) (Abraham et al., 2014).

In this context, we propose a Python open-source software named Visbrain, distributed under a BSD licence and dedicated to the visualization of neuroscientific data Visbrain is built on top of VisPy ([Campagnola et al., 2015](#)) package, and benefitiate therefore from high-performances graphical rendering offloaded to the graphic card. This high plotting performance is assessed using the VisPy ([Campagnola et al., 2015](#)) package. The goals of the Visbrain project are three folds : 1) Merge together several pure Python neuroscientific visualization tools, 2) Each module should comes with a graphical user interface (GUI) for non-python experts and should provide most used and extended functionalities, 3) Bring the community together to collaborate and adapt the software to larger needs.

³ <http://neuroimage.usc.edu/brainstorm/>

⁴ <https://sccn.ucsd.edu/eeglab/>

⁵ <http://www.fieldtriptoolbox.org/>

⁶ <http://mne-tools.github.io/stable/index.html#>

⁷ <https://pysurfer.github.io/>

⁸ <http://nilearn.github.io/>

Materials and Methods

The main philosophy behind Visbrain is to propose a package that centralize several visualization modules. Each module could come from very different fields but all should answer to one specific visualization type and must come with a modular and responsive GUI. In addition, neuroscientific datasets can potentially be large hence plotting methods should be efficient, with the ability to be updated in real-time through the GUI. This package does not intend to provide functions to analyse data except if those functions lead to a potentially useful illustration.

Programming language and code guidelines

As any new programming project, the choice of the language was the first interrogation. We naturally consider Matlab ([MathWorks, 2012](#)) or Julia ([Bezanson et al., 2017](#)) as they both are high-level languages. We finally chose Python as this mature language benefits from a large range of high-quality packages with a clear syntax, a huge community and documentation and, even more importantly, because we try to share this idea of open science and free software solution. In addition, Python software are portables and cross-platform and are easily distributed.

From a programming perspective, we did our best to avoid deep data copy and be able to load and process big data. The full Visbrain package is hosted on Github and is described in a NumPy like documentation and we further provide examples and datasets to improve understanding of those functions. Code blocks are well commented and we used static analysis tools as recommended by PEP8.

Graphics on the highway

As said before, neuroscientific data could be large, very large. For example, eighth hours of sleep data EEG recordings could weigh several gigabits according to the sample frequency. Hence, the visualization solution should be very efficient and, worst, should be able to be updated in real-time. For this latter point, Matplotlib ([Hunter, 2007](#)) has been discarded as this library is primarily designed to provide publication quality figure and is not well suited for handling large data and user interactions.

For those reasons, we dugged into the VisPy package ([Campagnola et al., 2015](#)) to find if this software could be a solution and indeed, it was. Basically, VisPy make a bridge between the comfortable syntax of Python and OpenGL which, in short, is really performant but not as intuitive as Python. In addition to those performances, VisPy is build on four levels, from the highest level very close to the Matplotlib style but suffering from weakest performances, to the lowest level (OpenGL Object Oriented), more difficult to manage and configure but providing the best graphics rendering. As Visbrain evolve, we went down from the highest to the lowest level for certain modules that do not rely on the basic objects, already implemented into VisPy. As a consequence, because graphic rendering are offloaded to the GPU and with the improvements of recent graphical cards, any modern laptop should benefit from this graphic treatment.

Graphical interface and user interactions

GUI are not a necessity, at least not for confirmed Python user but non-expert usually appreciate manual interactions into a dedicated interface. The fact is that it's easier for expert programmers to use an interface that for non-programmers to use command line functions and this is the primarily reason of interfaces based modules.

At this point of the development, we had to make a choice of how building those graphical user interface and, even more critically, if it was possible to embedded VisPy graphics in it. The cross-platform GUI toolkit [Qt⁹](#) provides Python binding tools, such as PyQt to the C++ Qt toolkit and therefore make possible to convert into a Python compatible GUI. In addition, the Qt designer has been extensively used to build modular and responsive GUI of each module.

Documentation and examples

In addition to the Python package, we provide a detailed step-by-step documentation build with [Sphinx¹⁰](#) and [hosted on github¹¹](#). This documentation describe how to install Visbrain and use modules. We also described each GUI panel, each functionalities and inputs of each class module. Moreover, inside those interfaces, we also provide a description of each graphical element using tooltips which appear on mouseover.

⁹ <https://www.qt.io/>

¹⁰ <http://www.sphinx-doc.org/en/stable/>

¹¹ <http://etiennecmb.github.io/visbrain/sleep.html>

Results

Currently, Visbrain architecture is based on four modules :

- *Ndviz* : for visualization of multidimensional data, basic plotting and data mining
- *Brain* : a 3D MNI standard brain centered module with a lot of controls and functionalities
- *Sleep* : this module is really specific to sleep scientist and propose to visualize polysomnographic recordings and hypnogram edition
- *Figure* : a Matplotlib based module that wrap together most used functions for publication-ready figures.

Ndviz : efficient data mining

A common first step when starting a new scientific study is to explore the dataset in order to inspect the shape of the neural signals or finding artefacts for trials that need to be rejected. And, for large datasets it is not always evident. In this Visbrain package, we included a module called [*Ndviz*¹²](#) dedicated to the visualization of multidimensional signals.

Dispose your data into a grid

This idea of data mining module was induced by one of the [VisPy example¹³](#) in which thousands of signals, each one having thousands of points, can be instantly plotted by the use of graphics rendering offloaded to the GPU. Those signals are disposed into a two dimensional grid and the user can scroll on each one of them. We thought that this idea has a great potential for inspecting neuroscientific data. For example, 10 sessions of EEG recordings using 128 contacts can be visualized in a 10 by 128 grid and on each point of this grid seat the corresponding time series.

We embedded this example in a GUI ([**Figure 1**](#)), providing more controls on which dimension to pick. In addition, we also provide several ways to color those signals.

Plotting forms

Then, each signal of the grid can also be visualized independently and in several forms ([**Figure 2**](#)). As a simple continuous line which is the first choice for time series. Then, as markers or cloud of points which, for example, might be useful for displaying features used in machine-learning in order to see if they are well separated. Thirdly, as an image as neuroscience representations frequently used them. Inspecting a large amount of time-frequency maps might be a dedicated scenario. Finally, it is also possible to compute the histogram or spectrogram and to visualize them. For also add basic “Previous” and “Next” buttons to quickly jump for a signal to another.

From a multidimensional signal, the user can pick the dimensions he wants to inspect the dataset, and visualize it either on a complex 2D grid or using simples to common

¹² <http://etiennecmb.github.io/visbrain/ndviz.html>

¹³ https://github.com/vispy/vispy/blob/master/examples/demo/gloo realtime_signals.py

representations. Thus, *Ndviz* is a good way to quickly inspect data, produce basics plot and familiarize with dimension switching. A small piece of Python code is provided in the **code snippet 1**.

Brain : visualization on a standard 3D MNI brain

The main idea behind the *Brain*¹⁴ module is to propose a GUI (**Figure 3**) able to perform complex visualizations using a 3D standard MNI brain. To this end, *Brain* is articulated around four distinct objects :

- The main brain template
- Deep sources
- Connectivity links between those sources
- Sub-structures

Importantly, those objects are independent from recording types can be added to the main canvas and can are highly configurables. We describe those objects in sections below.

MNI brains

With the installation of Visbrain comes three default templates containing or not the cerebellum and more or less smooth. This limitation of three templates can be extending by the use of input parameters to directly pass vertices and faces of user specific templates (**Figure 4**). Importantly, this visualization is not a volume i.e. it only display the brain surface. As a compensation of this latter point, the user can control the color of the surface. In addition to the provided templates, this module allows transparency control (from fully transparent to opaque). The user can also take brain slices over the (x, y, z) axis. This latter functionality can be interesting for medial wall inspection. Finally, *Brain* take the benefits of the VisPy cameras so that the brain can be rotate around the central point or in a airplane simulation way for deep visualization.

Region of interest

Secondly, we refer to region of interest (ROI), labelled volumes that can be add to the canvas. *Brain* can extract those structures from two atlases: brodmann areas and Automated Anatomical Labeling (AAL). As an example, the user can add brodmann areas 4 and 6, which respectively contains primary and premotor areas as well as the Supplementary Motor Area extracted from the AAL volume (**Figure 5**). Those volume are converted into surface with the possibility be smoothed before being displayed. Finally, those mesh inherits the same methods as the brain such as transparency and slice control.

Deep sources

Third object that can be add on the main canvas : sources. Here, the term sources can be understand as small balls inside the brain as, depending on the neuroscience field, it can be interpreted differently. For example, in MEG, those balls can be called source (source reconstruction) but insing intracranial data those sources are electrodes.

¹⁴ <http://etiennecmb.github.io/visbrain/brain.html>

We used the standard MNI coordinate system but *Brain* also includes functions for Talairach coordinates conversion (Talairach and Tournoux, 1993). We offer further source's parameters such as size, color, shape (i.e. disc, square, diamond...). A text can also be add to each one of them. More interestingly, it is possible to attach values to each source and, in this case, ball's radius will be proportional to this value. As an example, each site can materialize the beta power. Moreover, *Brain* contains methods that can detect and hide sites that are not contained into the brain template volume.

Taken together, color and shape can be useful for representing, for example, the intracranial implantation per subject. The proportional ball's size modulation according to the attached activity is an interesting feature that can give a first impression. But, the imposed planar representation of a laptop screen does not restitute the depth. As a complement, we provided the ability to project the activity on the cortical surface. Indeed, we look for every brain vertices contains in a 10mm (this radius is controllable) sphere around the source. Then, the activity attach to this source is finally set to the vertices within the sphere. In addition to the cortical projection, we also provide a cortical repartition which measure the number of contributing sources per vertex. Finally, all of those projections can also be applied on ROI. Those source's related functions are summarized in the **Figure 6**.

Connectivity

Last but not least object : the connectivity between sources can also be add and highly configured. An upper triangular connectivity array is passed to define the connectivity strength between those sources. In addition, using boolean mask on this array allow to control which connection need to be drawn. From the user side, line's width is also a parameter such as colors which can be either set by specifying one color per connectivity strength, or using colormap's based methods. Those methods automatically set a color according to the strength, to the number of connections per node or by density which look at the number of connections in a sphere of interest (**Figure 7**). A large number of connections often occurs a confused scene. In that case, *Brain* also provide the possibility to have a dynamic transparency where weaker connections are going to be more transparent.

High-definition screenshots

The last element of the chain is the possibility to save the scene into a compatible high-definition image (*.png, *.jpg, *.tif...). Note that some image file format conserve the transparency and, as a consequence, a transparent brain in the interface will also be transparent in the exported image. Finally, the exported image can be cropped and we also provide a method that automatically crop the image to the closest non-background pixel.

Class methods for a code-line control

We are frequently in the scenario where scientific figures needs several version and ajustements. While some of those figures are basics and do not take time to be redrawn, others might contains 20 brains, with the cortical projection on each one and finally organized in a grid. If for each brain, the GUI have to be opened, then run the projection and finally make a screenshot of it, thus the full process for having the final figure can be long. In this context, each methods or parameter can be set in command line and this figure process

can easily be embedded in a loop. All of those methods are referenced into the [documentation](#)¹⁵.

Taken together, Brain groups a relatively large amount of possibilities summarized in [Figure 8](#). We tried to write a clear [documentation](#)¹⁶ and also provide several basics and advanced [examples](#)¹⁷ and datasets.

Sleep : polysomnographic data visualization and edition

[Sleep](#)¹⁸ is the Visbrain module dedicated to the visualization and edition of polysomnographic data. It allows the user to load a variety of standard electrophysiological file formats (Brain vision *.eeg, European Data Format *.edf, Micromed *.trc, Elan *.eeg). For non-supported files, we also provide the possibility to directly pass NumPy arrays.

Data visualization

After loading the data, the GUI propose as many checkbox as the number of channels for controlling which one to display. In addition, the amplitude can either be set per channel or though all of them. By default, Sleep propose to visualize 30 seconds of polysomnographic data but this window length is configurable. For spectral investigations, we also embedded a highly configurable spectrogram and topographic representation. As any of the Visbrain module, Sleep inherit from this modularity and any panel can either be display or hide as shown in [Figure 9](#).

Hypnogram visualization and scoring

A standard procedure in the sleep domain consist of scoring full night recordings that means setting one the referenced sleep stage. We used the nomenclature of the American Academy of Sleep Medicine ([Iber, 2007](#)) which define 6 possible scoring states Artefact, Wake, Rapid-Eye Movement (REM) and non-REM stages 1, 2 and 3.

Sleep provide the ability to load an existing hypnogram or to start a new one. By default, those stage are displayed in the order : Art, REM, Wake, N1, N2, N3 but this order can be set when defining the Sleep instance, in the Python script. We also provide three scoring methods in the GUI:

- A drag and drop way : the user can directly interact with the hypnogram, insert new points by clicking on it and move those points. This method is more appropriate for shorter data as 8 hours night requires precision that is difficult to obtain with a mouse.
- Using a scoring table : the user can manually specify where the sleep stage start and finish in an interactive table.
- Using keyboard shortcuts : we associate a shortcut for each step. Pressing on the key set the stage on the hypnogram and the next time window is displayed.

¹⁵ <http://etiennecmb.github.io/visbrain/brain.html#user-functions>

¹⁶ <http://etiennecmb.github.io/visbrain/brain.html#>

¹⁷ <https://github.com/EtienneCmb/visbrain/tree/master/examples/brain>

¹⁸ <http://etiennecmb.github.io/visbrain/sleep.html>

At any moment, the hypnogram data can be saved or exported in a black and white publication-ready figure (**Figure 10**).

Automatic events detection and signal processing tools

Sleep data contains microstructural events that are specific to sleep stages. For example, the [12, 14hz] spindle bursts are frequently found in N2 stage. Therefore, those events represent an important scoring help. *Sleep* actually contains 6 detection types : spindles, k-complex, REM, slow-waves, muscle twitchings and a peaks detection that can be used to compute heart rate. Those detections can be runned on any channel, from the GUI, and are directly add into the channel and on the hypnogram (**Figure 9**).

In addition to those detection algorithms, we add further reversible signal processing tools that allows a real-time filtering and the ability to extract the amplitude, the phase or the power within specific frequency bands. Finally, the user can also re-referenced the data from the software or bipolarized, a frequent procedure for intracranial data (**Jerbi, 2009**)

GUI commodities

On top of the functionalities presented above, the entire GUI is controllable using keyboard shortcuts which enable a quick navigation. The list of those shortcuts can be opened from the Help menu of the interface.

Figure : page layout of publication-ready complex figures

At the end of this chain of modules, we also provide a Matplotlib based tools that we called *Figure*¹⁹ with the aim to simplify the layout of saved pictures. *Figure* do not provide any tools that is not already provided by Matplotlib, but instead wrap together functionalities that present a serious interest with the Visbrain modules and a full integration. In order to works, this module requires picture paths and then, can propose a simple grid representation on which it is off course possible to add titles, x and y labels and highly configurable colorbars that can either be set per figure or shared across subplots. The final produced figure can be exported in high-resolution with a controllable dpi level for publication integration. A layout example, produced with the **code snippet 3** is provided in **figure 11**.

¹⁹ <http://etiennecmb.github.io/visbrain/figure.html>

Discussion

The complexity of recent analysis procedures make neuroscientific visualizations demanding. We are currently trying to group a set of existing visualization tools, that should be transparent regardless to the recording types. This open-source Python package, Visbrain, benefits from graphics rendering offloaded to the GPU for efficient data plot.

This software actually includes 4 modules :

- *Ndviz* : for data mining and basic plotting. From a scientific pipeline perspective, *Ndviz* is the first step as it allows to quickly inspect large data, produce basic plotting (such as simple lines, cloud of points, images, histogram, spectrogram). Therefore, the multidimensional plotting capabilities enable to quickly identify bad signals and artefacts. We think that digging in fresh data with this software might be useful or for educational purposes.
- *Brain* : dedicated to visualizations using a 3D standard MNI brain such as intracranial or MEG source reconstruction representation. It is also possible to represent region of interest and to project source's activity on it or to illustrate 3D connectivity between those sources. We paid a very particular attention to color properties (colormaps, limits, colors under/over threshold...) as this highly declinable feature is usually the more explicit one in neuroscientific papers.
- *Sleep* : specific to polysomnographic data visualization and edition, this module is really addressed to sleep scientist. The main idea behind *Sleep* is to be able to visualize long night recordings and benefits from the OpenGL efficiency. We provide the ability to load standard file format, to configure which channel to display and provide further spectral tools such as spectrogram, filtering or topographic representations. Another main feature of *Sleep* is the possibility to score hypnogram, help by automatic events detection such as spindles or k-complex.
- *Figure* : for quality layout figure rendering, this tool close the chain of the Visbrain modules by offering a simple but easy to use way of organizing figures and quickly add additional labels, color background and colorbars.

If we set aside this last module, the three others comes with a modular and responsive graphical user interface designed with Qt. If both esthetic and ergonomics are not essentials, we still have paid attention to it, hence, provide comfortable softwares.

Considering the number of visualization possibilities, Visbrain is in its early stage of development but we think that the base is present and further programmers can contribute to make it grow and build a real Python open-source solution for Neuroscientific visualization needs.

Future directions

The first thing on our planning is to move from PyQt4 to PyQt5 as this new version of PyQt is needed for newer Python versions. Generally, each module could be improved especially for preserving memory as big data are frequent. Finally, we planned to add new modules especially for planar connectivity illustrations and topographic plotting.

Conclusion

The Python language brings portability and cross-platform installation to Visbrain. In addition, as researcher frequently move, we paid a particular attention to propose a solution that might be used on any computers. We try to build those modules with coherence, with intuitive interfaces and to provide consistent and extensive documentations.

Figures

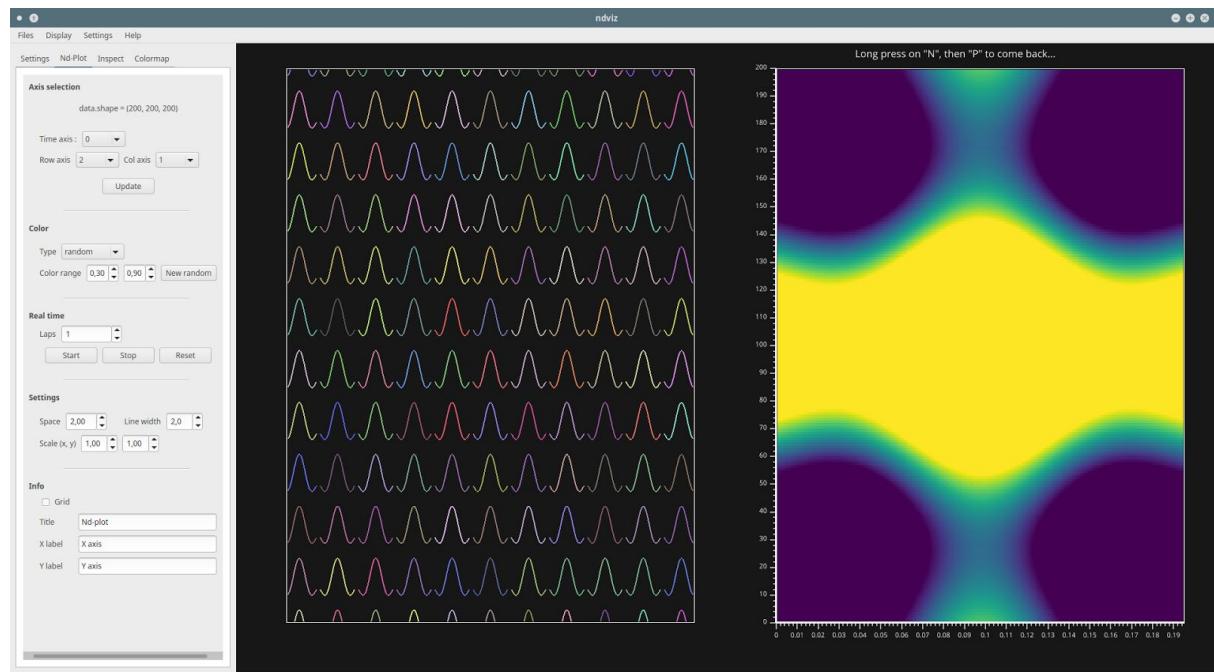


Figure 1 : Example of the graphical user interface of the *Ndviz* module. Leftmost is the setting panel, and side-by-side all of the sines disposed in a 2D grid and rightmost, the image formed by those sines.

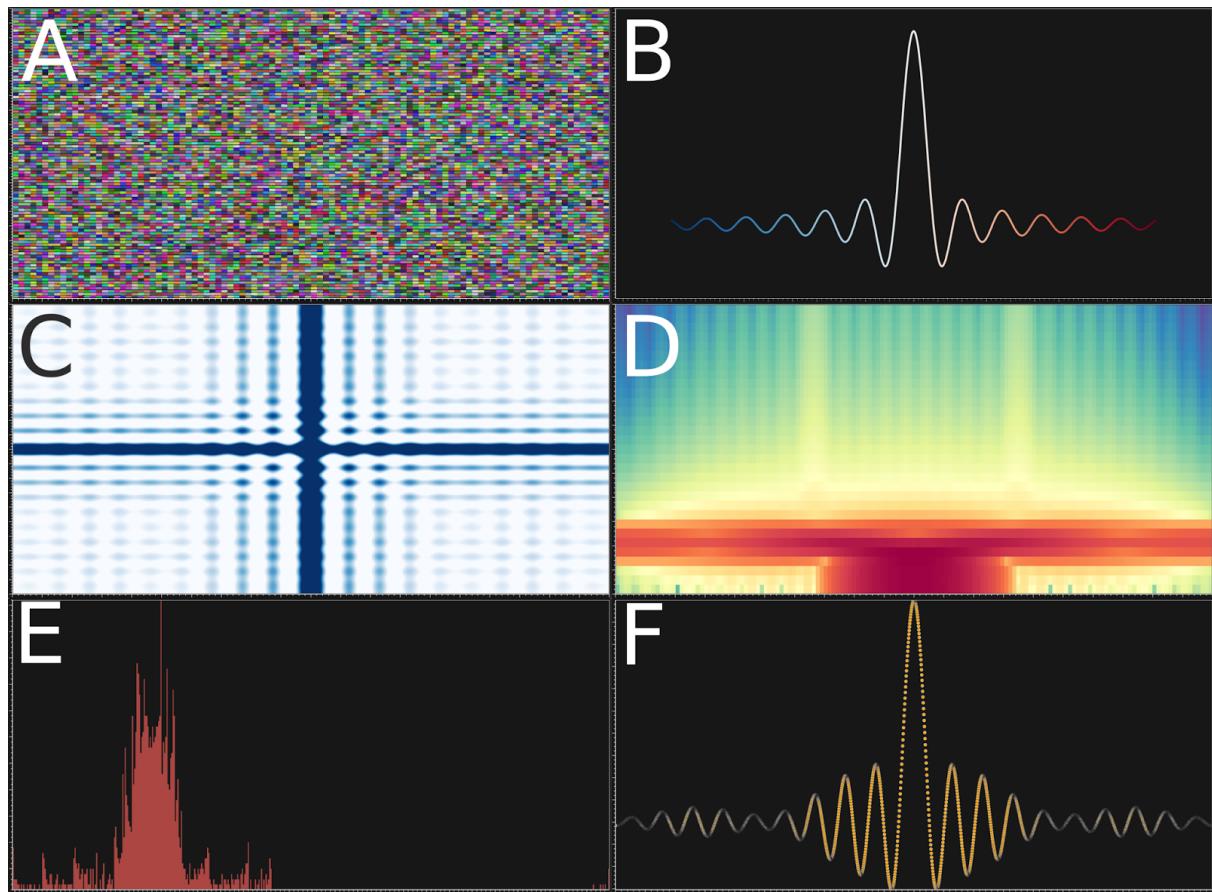


Figure 2 : Plotting functionalities from the *Ndviz* GUI, (A) Grid of 200x200 signals of 1000 time points each, visually differentiate with random color. Then, each one of those signals can be represented as a simple continuous line (A) as markers or points (F) or by picking two dimensions, the cardinal sine can be displayed as an image (C). Finally, it is also possible to compute and visualize the histogram (E) or the spectrogram (D).

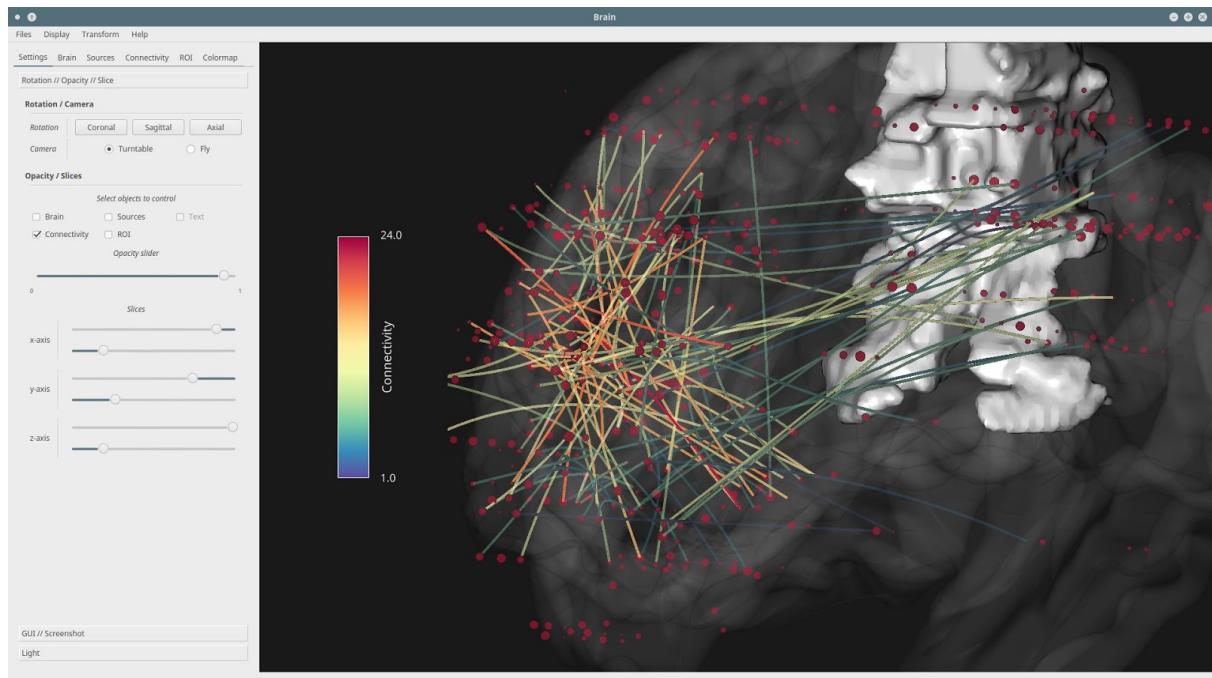


Figure 3 : Example of the graphical interface of the *Brain* module.

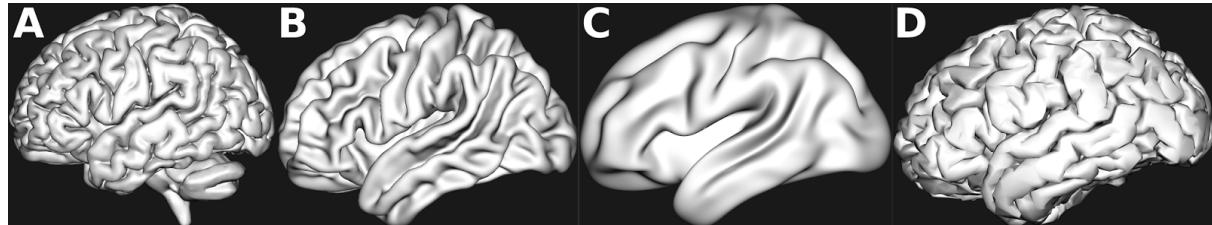


Figure 4 : Brain templates inside the *Brain* module. By default, *Brain* propose three templates (A, B and C) but using inputs parameters, the user can directly pass faces and vertices of his own template (D)

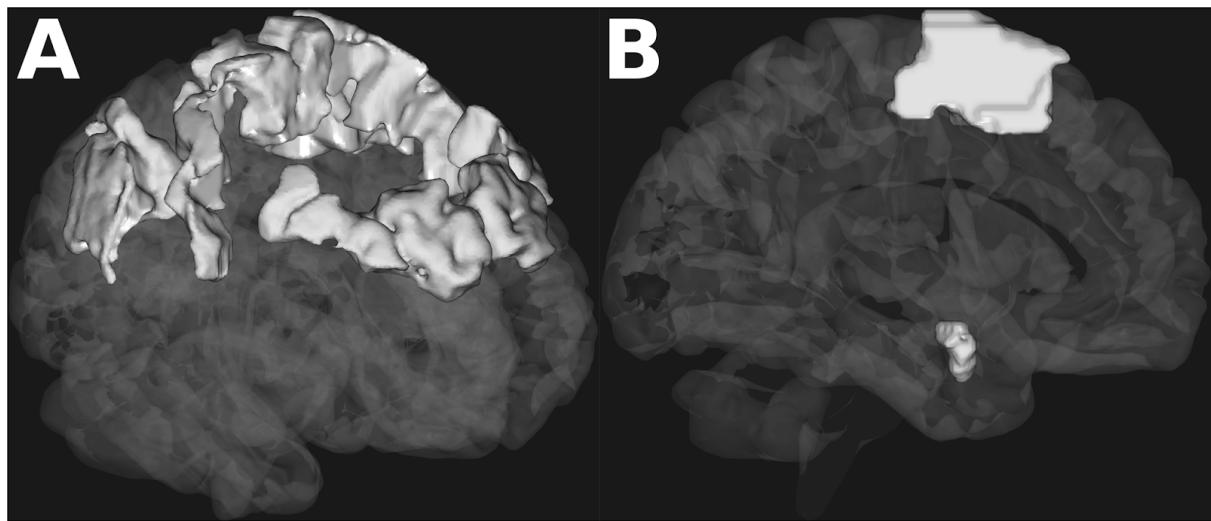


Figure 5 : Examples of region of interest displayed, (A) Brodmann area's 4, 9 and 40, (B) Supplementary motor area and amygdala only for the left hemisphere.

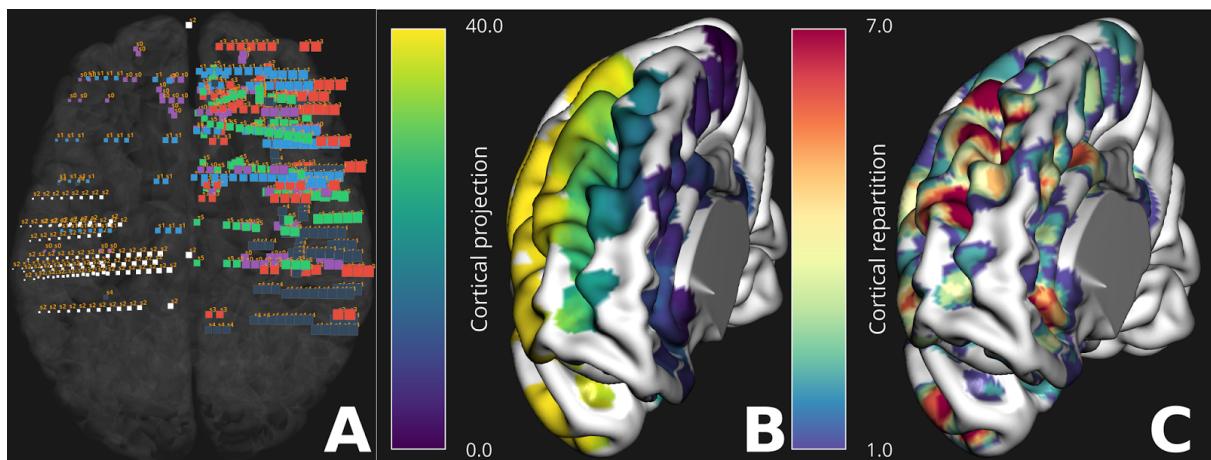


Figure 6 : Example of source objects and projections. (A) Sources obtained from the intracranial data of 6 subjects, each one having its own color. For the sake of this illustration, we set the activity of each source its x coordinate so that the diameter of each source increase along this axis. There's not neuroscientific meaning behind this. In addition, the name of each subject is added to each source, (B) Cortical projection of the source's activity only on the right hemisphere. Once again, according to the colorbar, the activity increase along the x axis, (C) Cortical repartition of this intracranial dataset. Accordingly to the colorbar, the blue means only one source contribute and in red, seven. this is particularly useful to see the distribution of the contributing sources on the cortical surface.

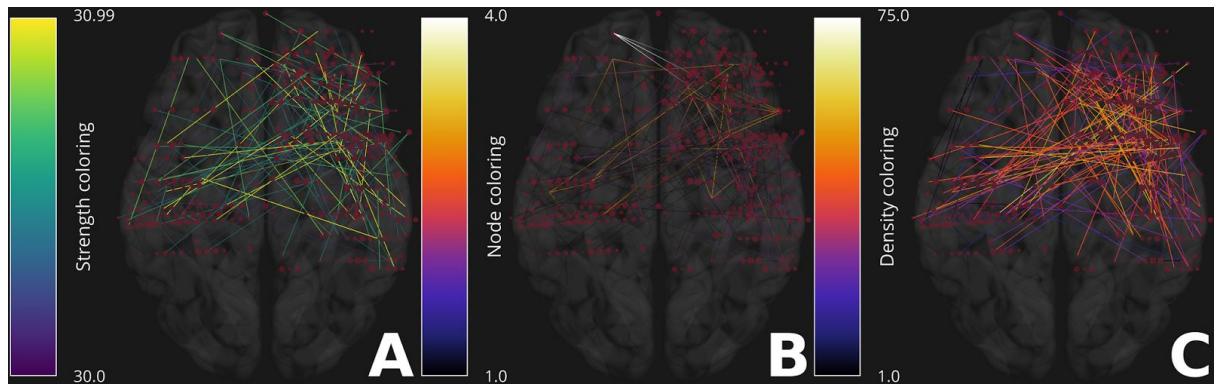


Figure 7 : Example of connectivity. Links are either colored according to the strength of connections (A) or by the number of connections per node (B) or finally according to the density i.e. the number of connections in a sphere of interest (C)

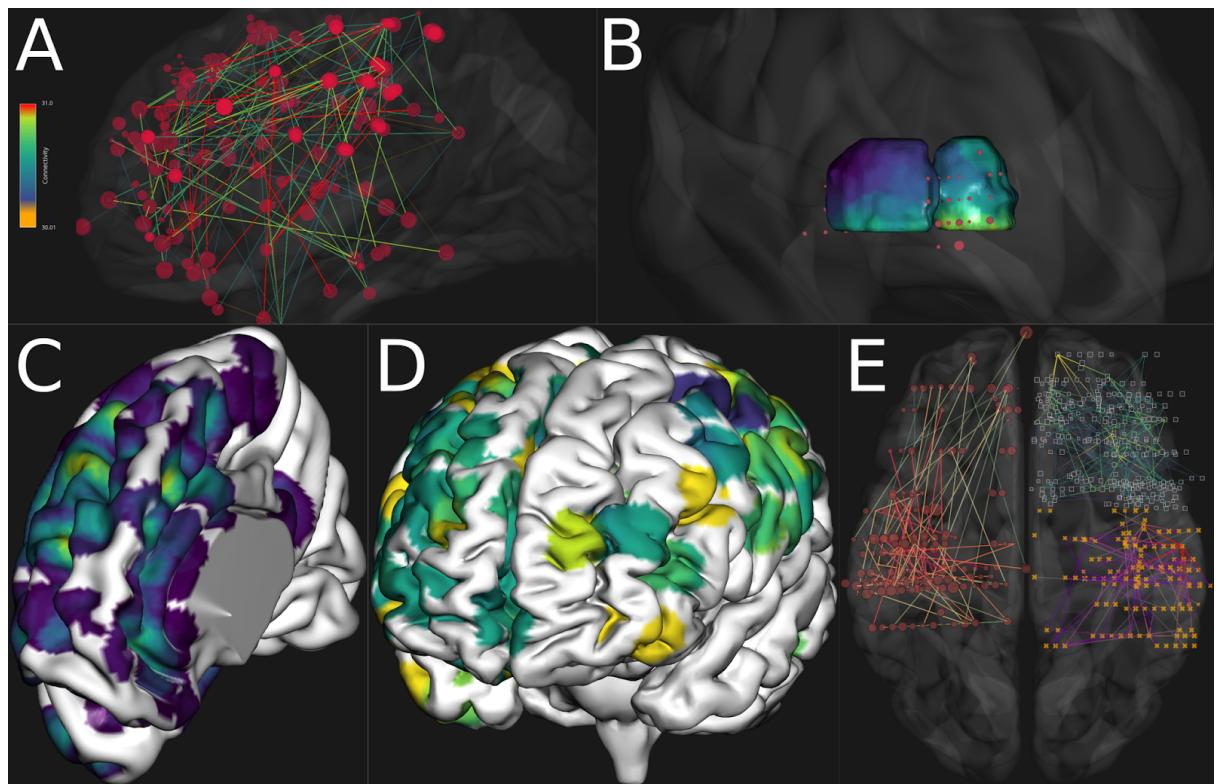


Figure 8 : Main functionalities of the *Brain* module, (A) Add deep sources inside the MNI brain and materialize connectivity, (B) Thalamus with the beta power of MEG source's reconstruction projected on it, (C) Cortical repartition i.e. number of contributing sources per vertex projected on the right brain hemisphere, (D) Cortical projection of high-gamma power recorded using intracranial data, (E) Complex scene made up of several objects including three types of sources (red disc, blue square and yellow cross) and connectivity link for each one of them.

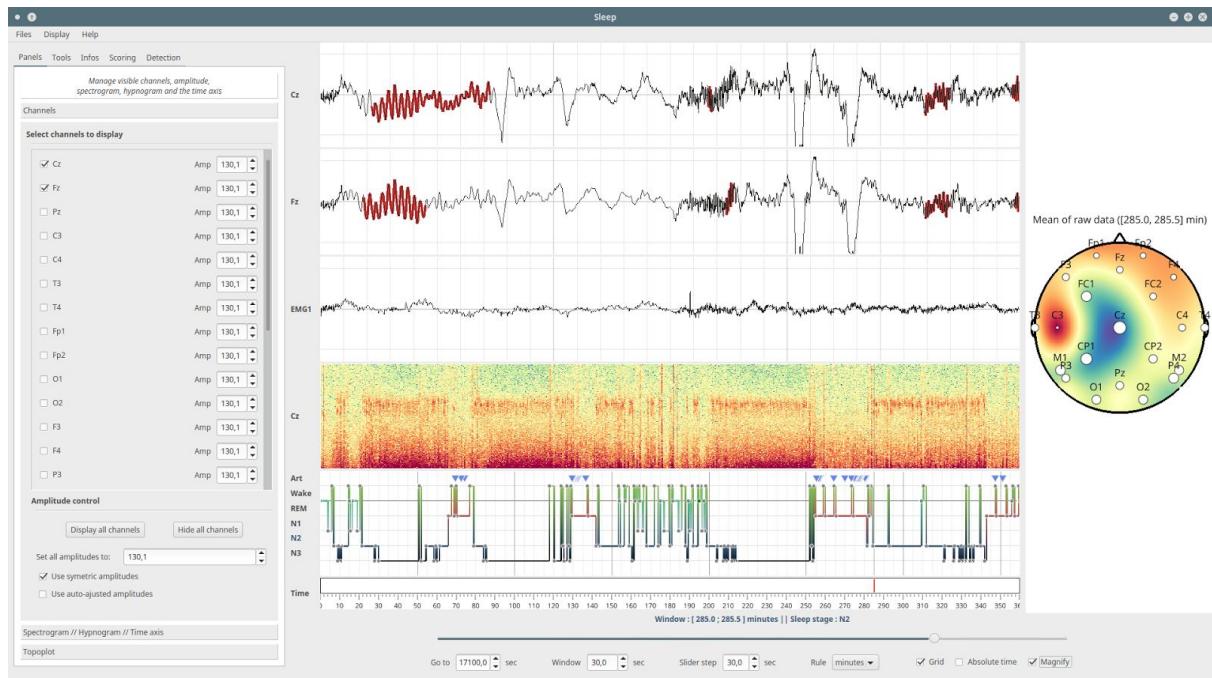


Figure 9 : Example of the graphical interface of the *Sleep* module. Leftmost is the setting panel that include all of plotting and signal processing controls and at the bottom of the figure, the navigation bar with time setting properties. Sleep data of three channels are plotted (Cz, Fz and EOG1) and in red, an example of GUI integration of microstructural event detection. *Sleep* also enable to add a spectrogram and provide further time-frequency settings. Finally, we also plot the hypnogram and the small markers on top of the hypnogram represent the time location of detected events. Finally, rightmost is an example of topographic representation.

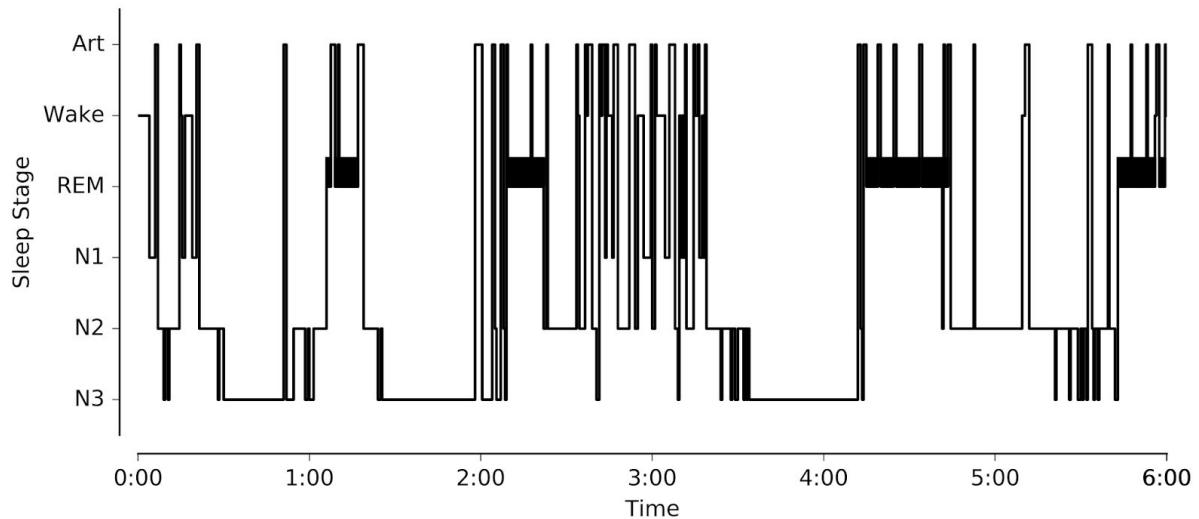


Figure 10 : Example of black and white exported hypnogram.

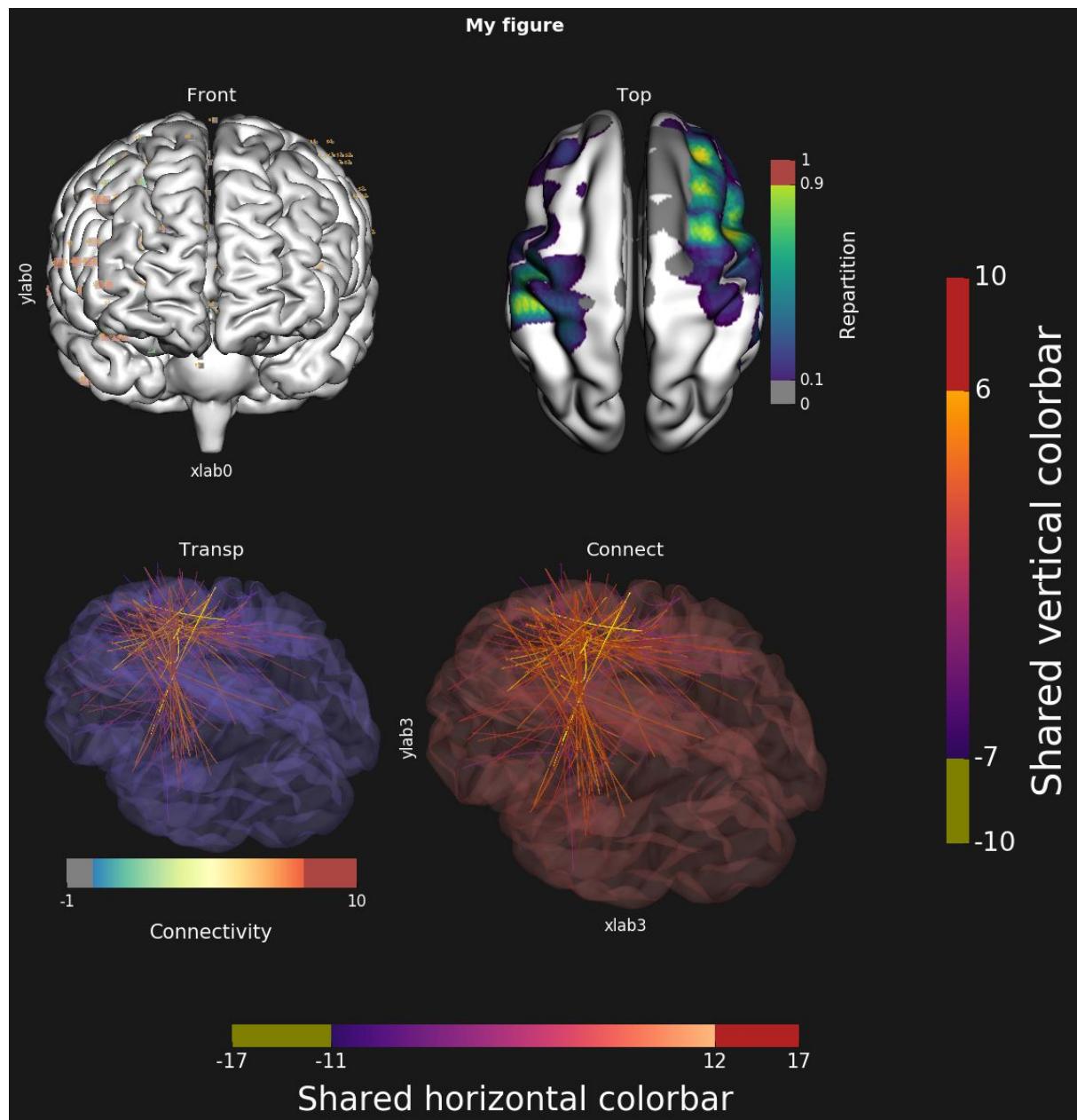


Figure 11: layout example produced with the *Figure* module.

Code snippets

```
import numpy as np
from visbrain import Ndviz

# Sampling frequency :
sf = 1024.
npts = 200
time = np.arange(-npts/2, npts/2)/1024.
# Create a 2d signal :
y = np.sinc(2*10*time).astype(np.float32)
y = y.reshape(len(y), 1) + y
# Add a little bit of noise :
y = y**2 + np.random.rand(*y.shape) / 10
Ndviz(y, sf=sf).show()
```

Code snippet 1 : generate 200 cardinal sine of 200 points each and visualize them inside the *Ndviz* module. This code generate the **figure x**.

```

# Import the Figure module :
from visbrain import Figure

# Set the list of files to load :
files = ['front.png', 'top.png', 'connect.png', 'connect.png']

# Define titles, xlabel and ylabel :
title = ['Front', 'Top', 'Transp', 'Connect']
xlabels = ['xlab0', None, None, 'xlab3']
ylabels = ['ylab0', None, None, 'ylab3']

# Define the background color of each axis : the two last pictures are
# transparent brains. By setting the background color to 'slateblue' and
# '#ab4642', brains turn respectively to blue and red :
ax_bgcolor = [None, None, 'slateblue', '#ab4642']

# Define the Figure object :
f = Figure(files, titles=title, figtitle='My figure', xlabel=xlabels,
            ylabel=ylabels, grid=(2, 2), ax_bgcolor=ax_bgcolor, y=1.,
            fig_bgcolor=(0.098, 0.098, 0.098), figsize=(12, 12),
            text_color='white', auto_crop=True)

# Add a colorbar only to the second axis :
f.colorbar_to_axis(1, (0, 1), 'viridis', title='Repartition', ticks='complete',
                   vmax=.9, over='#ab4642', fz_ticks=12,
                   vmin=.1, under='gray')

# Add a colorbar only to the third axis :
f.colorbar_to_axis(2, (-1, 10), 'Spectral_r', title='Connectivity', ticks='minmax',
                   vmax=8, over='#ab4642', fz_ticks=12,
                   vmin=0, under='gray', orientation='horizontal')

# Add a vertical shared colormap :
f.shared_colorbar((-10, 10), 'inferno', fz_title=30, vmin=-7, vmax=6,
                  under='olive', over='firebrick', position='right',
                  title='Shared vertical colormap', fz_ticks=20, pltmargin=.1,
                  figmargin=.1)

# Add a horizontal shared colormap :
f.shared_colorbar(cmap='magma', clim=(-17, 17), fz_title=25, vmin=-11, vmax=12,
                  under='olive', over='firebrick', position='bottom',
                  title='Shared horizontal colormap', fz_ticks=15, pltmargin=.1)

# Save the final figure in 600 dpi:
f.save('figlayout.png', dpi=600)

# Alternatively, you can show the figure :
f.show()

```

Code snippet 3 : generate 200 cardinal sine of 200 points each and visualize them inside the Ndviz module. This code was used to generate the **figure x**.

References

- Abraham, A., Pedregosa, F., Eickenberg, M., Gervais, P., Muller, A., Kossaifi, J., et al. (2014). Machine learning for neuroimaging with scikit-learn. *arXiv preprint arXiv:1412.3919*.
- Bezanson, J., Edelman, A., Karpinski, S., and Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review* 59, 65–98.
- Campagnola, L., Klein, A., Larson, E., Rossant, C., and Rougier, N. P. (2015). VisPy: Harnessing The GPU For Fast, High-Level Visualization. in *Proceedings of the 14th Python in Science Conference* Available at: <https://hal.inria.fr/hal-01208191/> [Accessed May 23, 2017].
- Delorme, A., and Makeig, S. (2004). EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of neuroscience methods* 134, 9–21.
- Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., et al. (2013). MEG and EEG data analysis with MNE-Python. *Frontiers in neuroscience* 7, 267.
- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing In Science & Engineering* 9, 90–95. doi:10.1109/MCSE.2007.55.
- Iber, C., Ancoli-Israel, S., Chesson, A., Quan, S. F., and others (2007). *The AASM manual for the scoring of sleep and associated events: rules, terminology and technical specifications*. American Academy of Sleep Medicine Westchester, IL.
- Jerbi, K., Ossandón, T., Hamamé, C. M., Senova, S., Dalal, S. S., Jung, J., et al. (2009). Task-related gamma-band dynamics from an intracerebral perspective: Review and implications for surface EEG and MEG. *Human Brain Mapping* 30, 1758–1771. doi:10.1002/hbm.20750.
- MathWorks, I. (2012). MATLAB and Statistics Toolbox Release.
- Oostenveld, R., Fries, P., Maris, E., and Schoffelen, J.-M. (2011). FieldTrip: open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. *Computational intelligence and neuroscience* 2011, 1.
- Tadel, F., Baillet, S., Mosher, J. C., Pantazis, D., and Leahy, R. M. (2011). Brainstorm: a user-friendly application for MEG/EEG analysis. *Computational intelligence and neuroscience* 2011, 8.

Talairach, J., and Tournoux, P. (1993). Referentially oriented cerebral MRI anatomy: an atlas of stereotaxic anatomical correlations for gray and white matter. Thieme.

Septième partie

Étude 6 : Sleep, visualisation de données polysomnographiques et scorage

INTRODUCTION

Mon introduction

Sleep: a Python open-source software for visualizing and scoring sleep data

Etienne Combrisson^{1,2*}, Raphael Vallat^{2,3*}, Jean-Baptiste Eichenlaub⁴, Christian O'Reilly⁵, Tarek Lajnef^{1,6}, Aymeric Guillot^{2,7}, Perrine Ruby^{2,3,§} and Karim Jerbi^{1,§}

¹ Département de Psychologie, Université de Montréal, Canada

² Lyon 1 University, Lyon, France

³ Lyon Neuroscience Research Center, Brain Dynamics and Cognition, INSERM U1028, UMR 5292, Lyon University, France

⁴ Department of Neurology, Massachusetts General Hospital, Harvard Medical School, Boston, MA, USA

⁵ Blue Brain Project, École Polytechnique Fédérale de Lausanne, Geneva, Switzerland

⁶ Center for Advanced Research in Sleep Medicine, Hôpital du Sacré-Coeur de Montréal, Montreal, Quebec, Canada

⁷ Inter-University Laboratory of Human Movement Biology, 27-29 Boulevard du 11 Novembre 1918, F-69622, Villeurbanne cedex, France

* These authors contributed equally to this work.

§ These senior authors contributed equally to this work

Corresponding authors:

Etienne Combrisson (e.combrisson@gmail.com)

Raphaël Vallat (raphael.vallat@inserm.fr)

Abstract

We introduce *Sleep*, a new Python open-source graphical user interface (GUI) dedicated to visualization, scoring and analyses of sleep data. Among its most prominent features are: 1) Dynamic display of polysomnographic data, spectrogram, hypnogram and topographic maps with several customizable parameters, 2) Implementation of several automatic detection of sleep features such as spindles, K-complexes, slow waves and rapid eye movements, 3) Implementation of practical signal processing tools such as re-referencing or filtering, and 4) Display of main descriptive statistics including publication-ready tables and figures. The software package supports loading and reading raw EEG data from a standard file formats such as European Data Format, in addition to a range of commercial data formats. Most importantly, *Sleep* is built on top of the VisPy library, which provides GPU-based fast and high-level visualization. As a result, it is capable of efficiently handling and displaying large sleep datasets. *Sleep* is freely available (<https://github.com/EtienneCmb/visbrain>) and comes with sample datasets and an extensive documentation. Novel functionalities will continue to be added and open-science community efforts are expected to enhance the capacities of this module.

Keywords: Electroencephalography (EEG), sleep scoring, sleep graphoelements, automatic detection, visualization, hypnogram, python, vispy, opengl

Declarations

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflict of interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Introduction

Polysomnography provides a comprehensive recording of the major physiological changes associated with sleep and is hence the gold standard for modern sleep analysis, both in research and clinical settings. At its simplest, it consists of monitoring at least 2 electroencephalogram (EEG), an electro-oculogram (EOG) and a submental electromyogram (EMG), providing sufficient information to identify sleep stages (sleep scoring) according to standard international established guidelines. A first set of rules were published by Rechtschaffen and Kales (R&K) in 1968 (Rechtschaffen and Kales, 1968) and proposed to divide sleep into 5 stages with distinct electrophysiological properties, named rapid-eye movement (REM) and non-REM (NREM) stages 1, 2, 3, 4. This nomenclature was updated in 2007 by the American Academy of Sleep Medicine (Iber, 2007) and sleep stage 3 and 4 have been merged into stage N3. In humans, a normal night of sleep consists of a repetition of four or five cycles in which sleep stages tend to follow each other in a particular order. Sleep staging is generally done visually by inspecting consecutive polysomnographic segments of 30 seconds. It results in a hypnogram which represents the succession of sleep stages across time. Apart from being time-consuming, visual sleep scoring is subject to both inter and intra-rater variability and is thus far from being optimal. By contrast, automatic sleep scoring has the advantage of being fast, reproducible and with generally good agreement with visual scoring (Berthomier et al., 2007; Lajnef et al., 2015a), yet its usage is far from being widespread and most sleep laboratories still rely on visual scoring, using either commercial softwares or in-house packages. In many cases, these software tools come with their own data and hypnogram file formats, and this heterogeneity can represent a substantial obstacle for sharing of sleep data across laboratories or clinics. Some of the very few existing open sources graphical user interface (GUI) for reading and scoring sleep include [Phypno](#)¹, written in Python, and the MATLAB-based toolboxes [sleepSMG](#)² or [SPiSOP](#)³.

With this in mind, we developed *Sleep*, an intuitive and efficient open-source graphical user interface dedicated to the visualization of polysomnographic recordings and scoring of sleep stages. *Sleep* supports a range of data file formats and provides several scoring aid including the detection of essential features of NREM and REM sleep such as spindles, K-complexes, slow waves and rapid eye movements. *Sleep* was written in Python, an easy-to-learn high-level programming language widely used in the scientific community. We developed *Sleep* on top of [VisPy](#)⁴ (Campagnola et al., 2015), a Python scientific library based on OpenGL which offloads graphics rendering to the GPU in order to provide fast and high-quality visualization, even under heavy loads as is the case with large dataset. *Sleep* therefore benefits from the high performances provided by VisPy alongside Python's inherent qualities such as its portability and ease of use.

¹ <https://github.com/gpiantoni/phypno>

² <http://sleepsmg.sourceforge.net/>

³ <http://spisop.org/>

⁴ <http://vispy.org/>

Materials and Methods

Efficient scientific visualization often consists of finding the best possible way to explore the data and to illustrate results in an intuitive and straightforward manner. The huge variety of neuroscientific data types and acquisition modalities naturally requires a wide range of specific visualization tools. Ideally, the various software packages needed for the various applications should be free and capable of handling several types of brain data recordings. In this context, we are currently developing a Python package we called [Visbrain](#)³ distributed under a BSD licence, which provides and centralizes a number of useful brain data visualization utilities. We found a lack of existing software that wrap together an efficient, portable and user-friendly interface for polysomnographic data visualization and edition. As a result, we combined our skills in sleep analysis and Python programming to provide an open-source module, included within Visbrain, and named *Sleep*.

The choice of Python and the project vision

The choice of the programming language naturally turned to Python as this high-level and open-source language benefits from many libraries, an extensive documentation and a dynamic community. From data analysis to the production of high-definition paper figures, Python offers all the tools needed by scientists, with the comfort of a clean and easy to read syntax. *Sleep* is a pure Python software built on top of NumPy, VisPy, [PyQt4](#)⁴ and uses a limited number of functions from SciPy and Matplotlib. Thanks to the Python portability, the software can be installed and used on any platform. One of the initial objectives of the project was to provide a user-friendly and intuitive interface capable of efficiently load and display large sleep dataset. To this end, we paid a particular attention to avoid deep data copy and display only what is necessary. Therefore, even very large recordings with a consequent number of channels can be handled by *Sleep* on any modern laptop with snappy GUI response. From a programming perspective, we did our best to provide a clean, commented and high-quality code, with a NumPy style documentation and using static analysis tool, as recommended by PEP 8. *Sleep* is hosted on GitHub and we encourage Python programmers and sleep scientists to collaborate in order to collectively improve this software by extending its functionalities and data compatibilities.

Hardware accelerated graphics

In addition to an ergonomics and portable interface, another goal was to use an efficient plotting library able to support and process large sleep data. Matplotlib was an option, but although this package has been developed to produce publication quality figures, it is not well suited for plotting and interacting in real-time with large datasets. VisPy is a scientific visualization library based on NumPy and OpenGL and was primarily designed to provide both high performances with real-time interactions and publication quality figures. VisPy make a bridge between the intuitive Python syntax and modern shader-based OpenGL pipeline allowing to offload graphical rendering cost to the graphics processor (GPU). This

³ <https://github.com/EtienneCmb/visbrain>

⁴ <https://riverbankcomputing.com/software/pyqt/intro>

package has been well designed and is built on four levels, from a Matplotlib oriented one to the lowest-level (closer to OpenGL) being more flexible and efficient at the cost of a slower learning curve. Because all *Sleep* graphical elements are primitive 2D objects (line, points and images) it was not a necessity to go down to the lowest level of VisPy (`vispy.gloo`). Indeed, all required objects were already implemented into the `Visual` library. Hence, any modern computer equipped with a graphics processor unit should see the benefits of the hardware accelerated graphics implemented in *Sleep*.

Portable GUI through Python

Among the major cross-platform GUI toolkits that interface with Python, [wxWidgets](#)⁵ (wxPython), [Tcl/Tk](#)⁶ (TkInter) and [Qt](#)⁷ (PyQt/PySide) are probably the most known and used. We choosed PyQt which is a python binding for the C++ Qt toolkit. In addition, Qt Designer was used to design the graphical user interface.

Taken together, VisPy provides high-performance rendering graphics that are well integrated in a portable, modular and responsive Qt GUI using Python PyQt package.

Automatic events detection

One of the main objectives of *Sleep* was to provide a complete and easy-to-use interface for analyzing and staging sleep data. To this purpose, we implemented several algorithms for the automatic detection of sleep features that are embedded within the software ("Detection" panels). This include detections of spindles, K-complexes, slow waves, rapid-eye movements, muscle twitches and signal peaks. With the exception of the latter, all those features represent the landmarks of specific sleep stages and can therefore be used to quickly and easily identify the repartition of sleep stages within a period of sleep (see **Figure 1**). The main characteristics of each of these features are listed below.

- Sleep spindles refer to burst of 12 to 14 Hz waves predominant over central scalp electrodes and lasting between 0.5 and 2 seconds. These bursts of oscillatory activity have been known as a defining characteristics of N2 sleep.
- K-complexes are defined as sharp negative waves followed by a positive component, prominent over frontal scalp electrodes and lasting more than 0.5 seconds. Along with spindles, they constitute one landmark of N2 sleep and should not be mistaken with slow waves that defined N3 sleep.
- Slow-waves (or delta waves) are high-amplitude ($>75\mu\text{V}$) and low-frequency ($<3\text{Hz}$) oscillations that are present during the deepest NREM sleep stage, i.e N3 sleep. According to the standard international guidelines, N3 sleep is defined by the presence of 20% or more slow waves in a given epoch..

⁵ <https://www.wxwidgets.org/>

⁶ <http://www.tcl.tk/>

⁷ <https://www.qt.io/>

- As its name suggests, rapid eye movements (REM) sleep is characterized by rapid eye movements easily observable on the EOG channels. They consist of conjugate, irregular and sharply peaked eye movements, similar to some extent to those exhibited during wakefulness.
- Another fundamental aspect of REM sleep is its muscle atonia, as revealed by a low EMG activity.. However, some transient muscle activity or muscle twitchings (MTs) can also be observed. These short irregular bursts of EMG activity are superimposed on the background of low EMG activity.
- Finally, *Sleep* implements a signal peak detection algorithm that is useful for example to calculate the heart rate, provided that an ECG channel is present.

Altogether, the set of detectors implemented in our software offers a valuable help for scoring sleep stages by offering efficient algorithms to automatically identify the main features of each sleep stages. Detections can also be used for a more in-depth analysis of the sleep microstructure (e.g. Vallat et al., 2017). Comparisons of performances between our detections and visual scoring are reported for K-complexes and spindles in the Results section.

Signal processing tools

In addition to the automatic detection presented above, *Sleep* provide a few more signal processing tools such as signal demeaning, detrending and a filtering (using either Butterworth or Bessel filters) with the following band option : lowpass, highpass, bandpass or bandstop. In addition, after being filtered, further informations can be extracted from the Morlet wavelet complex decomposition (Tallon-Baudry et al., 1996) such as time-resolved and band-specific amplitude, power or phase. Each one of this signal processing tools are reversible and can therefore be activated and deactivated without altering the original data and without any data copy in order to minimize memory requirements. Finally, loaded data can be re-referenced directly from the interface by either re-referencing to a single channel or common-average (frequently used for scalp EEG datasets) or by using bipolarization, which consists of subtracting neural activity from consecutives sites (classically used in intracranial EEG, see Jerbi et al., 2009).

Documentation and examples

We are conscious that not all neuroscientists are familiar with Python programming. Therefore, we provided an extensive and detailed step-by-step documentation, build with [Sphinx](#)⁸ and [hosted on github](#)⁹. This documentation include a description of the graphical components and the main functionalities of the software. A PDF version of the documentation can also be downloaded directly from the "Help" contextual menu of the software. For scientists or students that wish to familiarize themselves with *Sleep* but do not have a dataset, we also provide anonymous and free-to-use example datasets, including

⁸ <http://www.sphinx-doc.org/en/stable/>

⁹ <http://etiennecmb.github.io/visbrain/sleep.html>

the corresponding loading scripts. Finally, for a real-time documentation, we used the tooltips provided by PyQt to describe each element of the interface.

Results

Graphical User Interface

The *Sleep* GUI can be subdivided into six distinct components :

- Settings panel
- Time properties bar
- Hypnogram
- Electrophysiological time series
- Spectrogram
- Topographic map

As the user interface is built up in a modular way, each of these components can be hidden or displayed, depending on whether the user prefers a light or fully-featured interface. Using the contextual menu, users can save and subsequently load the current display properties in order to easily retrieve previous session.

Settings panel and navigation bar

All controls and properties are grouped in a settings panel. This panel is subdivided into five thematic tabs :

- Panels : manage the visibility and properties of each plotted canvas.
- Tools : bundle of signal processing tools.
- Infos : basic informations of the current recording (name, sampling rate) and sleep statistics computed using the hypnogram (sleep stage duration, latency, etc). Note that the statistics can be exported in *.csv or *.txt file and are automatically updated when the hypnogram is edited.
- Scoring : scoring table that can be used to inspect and edit where each stage start and finish. This panel represents one of the three methods available within the software to edit hypnogram (see hypnogram edition section) and may be useful for example to score long periods of continuous and homogenous sleep by just providing the starting and ending times.
- Detection : perform and manage the automatic detection of several sleep features.

In addition to this setting panel, *Sleep* provides a navigation bar that can be used to set several temporal properties, such as the length of the current time window, time step between each window, time units and the use, if provided, of the absolute time of the current recording. This navigation bar also includes a grid toggle button that either hide or display the grid and a magnify option to enlarge short events (see **Figure 3**).

Electrophysiological time series

Sleep offers a dynamic control of the displayed polysomnographic time series and most of the settings are in the “Panels” tab. Indeed, each channel can be added or removed from the list of the currently displayed canvas. By default, *Sleep* displays the time series by frames of 30 seconds, which is the standard duration for stage scoring (Iber et al. 2007), but this value can be changed directly from the navigation bar. Furthermore, the amplitude of each channel can either be set independently, using a same range across all channels, or automatically adjusted according to the minimum/maximum of the currently displayed signals.

Time-frequency representation

As shown in **Figure 2**, the visibility and amplitude of each channel can be controlled from the GUI. The same applies for the spectrogram, which corresponds to a time-frequency representation of the entire recording performed on one channel. Among the definable parameters of the spectrogram are the channel on which it is computed, bottom and top limiting frequencies, length and overlap of the fast fourier transform and colormap properties. Finally, a topographic map based on the MNE and SCoT implementations (Billinger et al., 2014; Gramfort, 2013) can also be embedded inside the GUI for a full data inspection. The topological plot is computed from the mean of time window currently displayed and can be used to visualize the raw data, the amplitude or power in specific frequency bands.

Shortcuts

Navigation and operations inside a software can be sometimes repetitive. For that reason, *Sleep* comes with a bunch of native shortcuts to facilitate the visualization and stage scoring, a non-exhaustive list of which is provided in **Table 1**. For a complete list we refer the reader to the “Shortcuts” paragraph of the documentation.

Keys	Description
n	Move to the next window
b	Move to the previous window
-	Decrease amplitude across all channels
+	Increase amplitude across all channels
s	Display / hide spectrogram
t	Display / hide topographic representation
h	Display / hide hypnogram
p	Display / hide time bar
z	Enable / disable zooming
a	Scoring: set current window to Art (-1)
w	Scoring: set current window to Wake (0)
r	Scoring: set current window to REM (4)
1	Scoring: set current window to N1 (1)
2	Scoring: set current window to N2 (2)
3	Scoring: set current window to N3 (3)

Table 1. Main native shortcuts of the software.

Supported electrophysiological and hypnogram data formats

Sleep natively supports several standard electrophysiological file formats, including European Data Format (EDF *.edf), Micromed (*.trc), Brain Vision (*.eeg), and Elan (*.eeg). In addition, it is possible to load directly NumPy array or Matlab file using the command-line parameters.

The hypnogram of the corresponding dataset can also be loaded and then edited directly from the GUI. Accepted hypnogram file formats are *.txt, *.csv or *.hyp. There is a great heterogeneity among sleep laboratories in the hypnogram format. This represents an obvious barrier for data sharing. To overcome this problem, *Sleep* allows the user to specify the hypnogram format in a separate text file. This file should contain the names and integer values assigned to each sleep stages in the hypnogram file, as well as the number of values per seconds. During loading, the hypnogram file will be converted to *Sleep* native hypnogram

format described in **Table 2**. An example description file can be found [on the documentation](#).

Parameters	Values	Description
Time	1	Hypnogram file contains one value per second
Wake	0	The value assigned to Wake in the hypnogram is 0
N1	1	The value assigned to N1 sleep in the hypnogram is 1
N2	2	The value assigned to N2 sleep in the hypnogram is 2
N3	3	The value assigned to N3 sleep in the hypnogram is 2
REM	4	The value assigned to REM in the hypnogram is 4
Artefact	-1	The value assigned to Artefact in the hypnogram is -1

Table 2. Parameters of *Sleep* native hypnogram format.

Hypnogram edition

The hypnogram can be edited either from scratch or from an existing hypnogram file. There are three methods to edit the hypnogram using *Sleep* GUI:

- Using intuitive keyboard shortcuts (see Table 1). When a new stage is entered, the next window is shown.
- Using a table where each stage can be specified by its starting and ending time point.
- Using a drag and drop operation directly on the hypnogram canvas.

At any moment, the user can export the hypnogram or save a black and white publication-ready figure using the contextual menu (**Figure 2**)

GUI integration and validation of automatic events detection

The automatic events detection can be performed on any selected or visible channel. When the detection is completed, detected events are depicted directly on the selected channel using a specific color-code for each feature. In addition, the starting point, duration and stage of occurrence of each detected events are reported in the “Location table”. Users can then easily navigate between the detected events by clicking on a row, which automatically set the time so that the event is centered on the screen. Furthermore, this table can be exported to a *.csv or *.txt file. Users can perform an unlimited number of detections in a row on a single channel and then switch from one to another using the “Location” panel. Last but not least, the location of each detected events is reported on the hypnogram using specific visual cues for each detection types. Integration of the detection inside the GUI is shown in **Figure 4**.

To test how these detections performed on real datasets, we measured performances of the spindles and K-complexes detection methods using visually-annotated EEG segments of N2 sleep collected from full-night polysomnographic recordings of 14 participants (Eichenlaub et al., 2012, 2014, Ruby et al., 2013a, 2013b). Spindles and K-complexes were visually scored by an expert (JBE) as part of a previous work that focused specifically on the detection of these sleep features using machine-learning algorithm (Lajnef et al. 2015a).

To perform the detection methods using *Sleep* algorithm, all N2-sleep EEG segments were concatenated into a single file of 210 minutes with a single channel (C3) and with a sampling rate of 100Hz (native downsampling frequency of *Sleep*). Then, to evaluate the performances of our detection, we used two standards metrics: the sensitivity (1), which measures the proportion of correctly identified detected events and the False Detection Rate (FDR) (2) which assess the proportion of incorrectly detected events.

$$(1) \text{ Sensitivity} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

$$(2) \text{ False Detection Rate} = \frac{\text{False Positive}}{\text{False Positive} + \text{True Positive}}$$

where True Positive refer to the events scored by the expert and correctly detected by our methods, False Negative refer to the events scored by the expert but not detected by our method and False Positive refer to the events detected by our methods but not scored by the expert.

Performances of our detection algorithm are reported in **Figure 7**. For both spindles and K-complexes, we used 25 different thresholds ranging from 0 to 5 with 0.2 steps. The optimal threshold was defined as the one that maximizes the difference between sensitivity and FDR (Lajnef et al. 2015a). Regarding spindles, the best performance of our algorithm was obtained at a threshold of 2.4 standard deviations, yielding a sensitivity of 77.2% and a FDR of 40.1%. Regarding K-complexes, a threshold of 1.0 resulted in the best performances with a sensitivity of 70.7% and a FDR of 27.2%. These figures are similar to those of previous detection methods (Lajnef et al. 2015a, Devuyst et al. 2011). Moreover, the time of execution of these two algorithms at a given threshold is less than a minute on any modern laptop computer.

Sleep class inputs and code example

From a programming point of view, the high-level interface with our software is provided by the *Sleep* class. This class can take into account a few input arguments. Hence, loading sleep data can be assessed in three ways adapted to a range of users, from non-programmers to advanced users. As shown in the **code snippet 1**, running *Sleep* without further input arguments will ask the user to specify the path to a supported sleep dataset (*.eeg, *.edf or *.trc). In addition, the user can either use an existing hypnogram or start a new one from scratch. Alternatively, instead of using the interface to select the files, they can be directly passed as input arguments (**code snippet 2**). In this example, we also demonstrate how to change the default order of the sleep stages in the hypnogram using a

simple command-line option. If this option is not specified, the default display of *Sleep* is as follows: *Art*, *REM*, *Wake*, *N1*, *N2*, *N3*. Finally, several others file formats such as EEGLab, Neuroscan, EGI, GDF and BDF can be loaded using [MNE python package](#). We report in **code snippet 3** a method to pass data to *Sleep* after loading them using MNE python.

Discussion

We are currently developing an open-source Python software called Visbrain dedicated to the visualization of neuroscientific data. As a part of this project, we developed a module called *Sleep* designed to visualize and score sleep data.

Sleep comes with a graphical user interface in which we embedded high-quality plots with graphical rendering offloaded to the GPU. As a result, plotting and user interactions can be processed in real-time. The software is capable of loading several widely-used sleep data files format, such as European Data Format and BrainVision, and to stream efficiently all of the polysomnographic channels, even on an average modern laptop. On top of that, *Sleep* also provides the possibility to display time-frequency (spectrogram) and topographic representations of the data, with several adjustable parameters for each. Regarding sleep staging and hypnogram editing, *Sleep* offers intuitive manual scoring functionalities, signal processing tools and automatic detection of sleep features in order to facilitate this fastidious process. Once completed, users can export the scored hypnogram data and/or a publication-ready high-quality figure of the hypnogram in one click. Regarding the automatic detections, *Sleep* includes 6 computationally-efficient and robust algorithms for detecting some of the most prominent features of each sleep stages, including spindles, K-complexes, slow waves, rapid eye movements and muscle twitches. Spindles and K-complexes detection algorithms were validated on a visually scored dataset including 210 minutes of N2 sleep from 14 participants and resulted in performances similar to recent publications. Last but not least, these detections are implemented inside the GUI in an ergonomic and intuitive manner. We think that these detections may represent a valuable help not only in the process of staging sleep, but also for researchers that are interested in the microstructure of sleep.

Future directions

Regarding the programming environment, Visbrain, and therefore *Sleep*, is built on top of PyQt4 and not the latest version PyQt5. As a consequence, *Sleep* is currently not able to run on the newest Python version (3.6) and users that wish to install the software must first create a dedicated Python 3.5 environment with PyQt4. Second, we are also considering to extend the list of the default supported files and we encourage programmers or sleep scientists interest by this project to collaborate on it. Regarding sleep analysis we are

working on an automatic scoring function based on machine-learning algorithms, according to our previous experience in this field (Combrisson et al., 2017; Combrisson and Jerbi, 2015; Lajnef et al., 2015b). Finally, as different users have different needs, we are constantly improving the interface and functionalities of the software thanks to the feedbacks we receive.

Conclusion

We offer a portable and cross-platform software, installable and usable on most configuration. While there is still place for improvement, *Sleep* already provides a complete and intuitive interface designed by and for sleep scientists. We hope this software will contribute to a more open science and strengthen the idea that open-source software can be of professional quality.

Figures

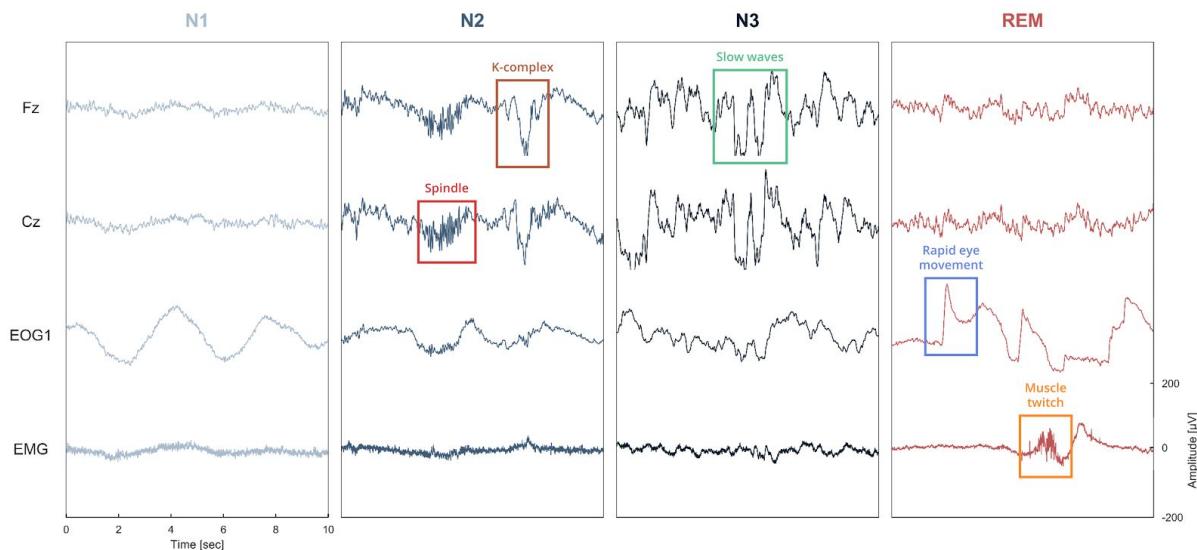


Figure 1: Illustration of the different sleep features observed in a polysomnographic recording of one individual. Note that these events are just displayed for illustrative purposes. To see examples of automatic detection actually performed by our software, see Figure 4. Spindles and K-complexes are the landmarks of N2 sleep. Slow waves are present during N3 sleep (sometimes referred to as slow wave sleep). Rapid eye movements, observed in the EOG channel, and muscle twitches, observed on the EMG channel, are two essential features of rapid eye movement (REM) sleep.

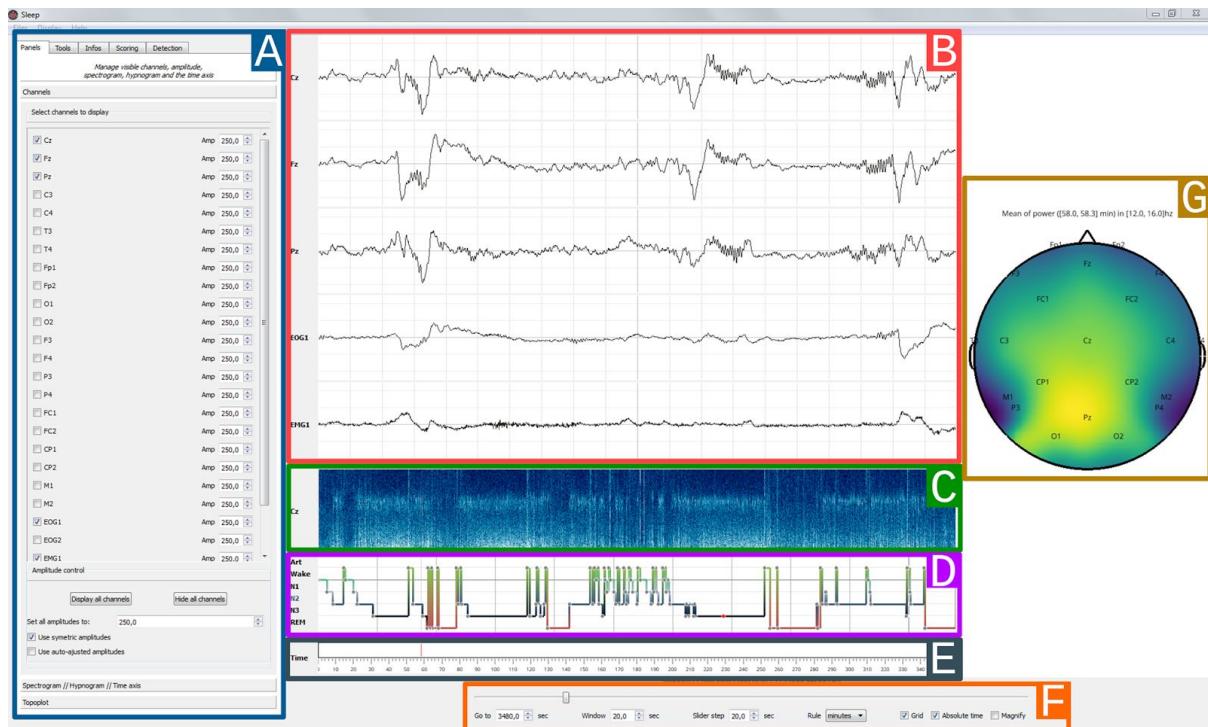


Figure 2 : Sleep main interface. Each element of the graphical user interface can either be displayed or hidden, (A) Settings panel containing all Sleep controls and parameters. The current displayed tab can be used to toggle channel visibility and to adjust individual amplitudes, (B) 30 seconds time window of electrophysiological data. Only 5 channels are currently displayed (Cz, Fz, Pz, EMG1), (C) The spectrogram displays the time-frequency representation of a specific channel for the entire recording, and can be useful to identify global changes in the spectral properties of the signal often associated with changes in sleep stages. Any channel can be picked and further time-frequency controls are available in the settings panel, (D) Hypnogram with one specific color per stage. The stage order can be changed from the default Artefact, Wake, REM, N1, N2, N3 , (E) Time axis with visual indicator, (F) Navigation bar with time settings : window length and step size, unit (seconds/minutes/hours), (G) Topographic data representation

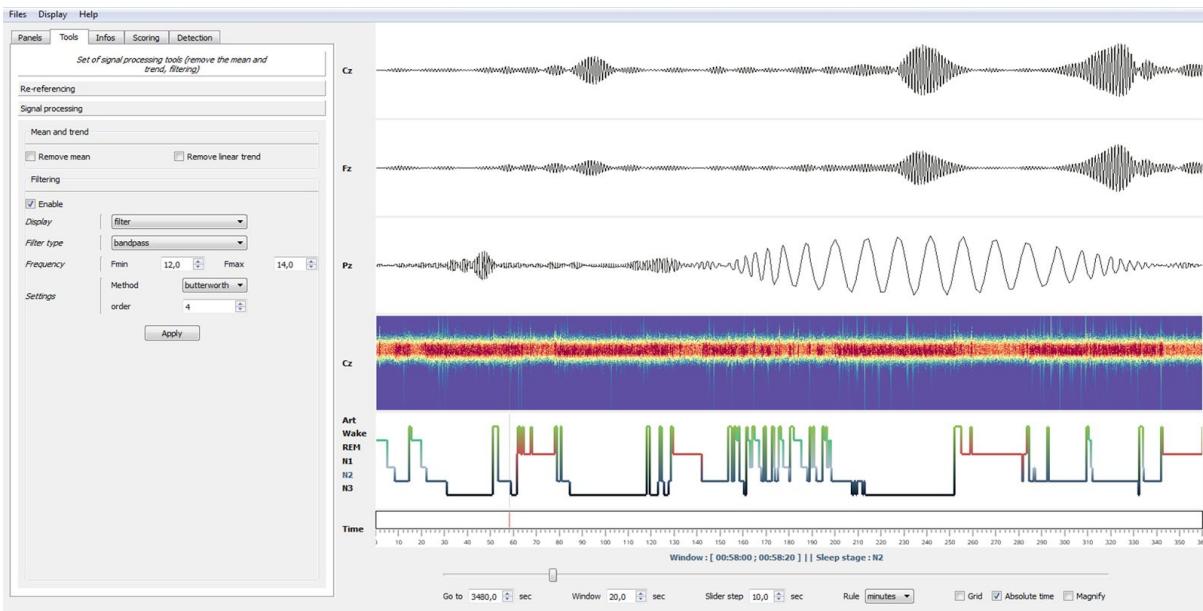


Figure 3 : Example of bandpass filtering. Using the Tools panel (left), the EEG signals have been bandpass-filtered in the spindles frequency band (12-14 Hz, butterworth filter). Using the “Enable” checkbox of the panel, this filtering operation can be disabled at any moment to retrieve the original EEG signals. Finally, by left-clicking on a specific time point in a channel or selecting the Magnify tools (bottom), users can enlarge events. This was used in this example to enlarge a sleep spindle observed on channel Pz.

Automatic events detection

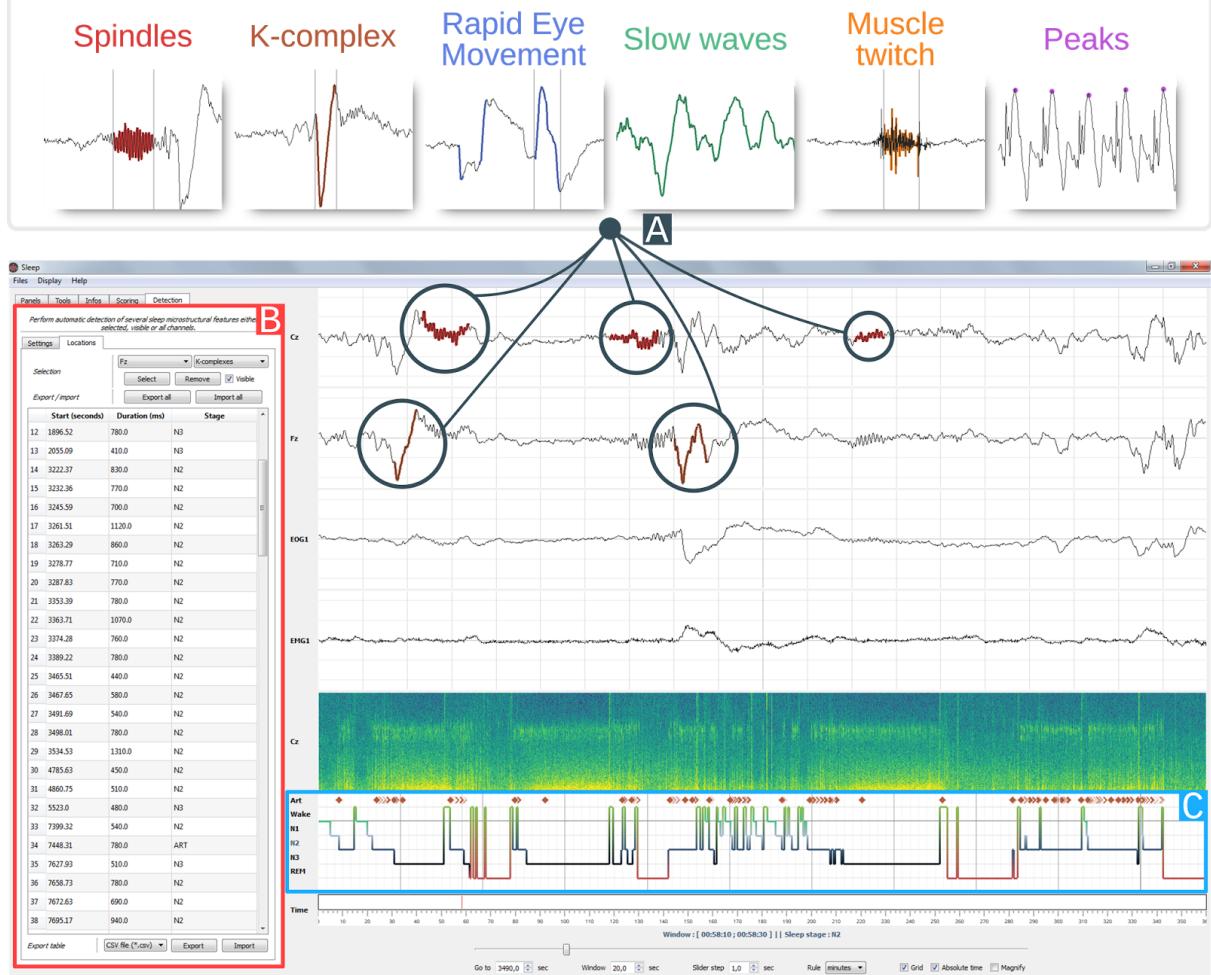


Figure 4 : GUI integration of the automatic event detection, (A) exemple of events detected on two channels. Sleep provide six detection types including spindles, k-complex, rapid eye movement, slow-waves, muscle twitch and peaks detection, and each one of them can be ran on any channel, (B) time location referencing all detected events and including the starting point, the duration and the sleep stage occurring during this event. A mouse click on a line center the corresponding event on the screen. This table can then be exported into *.csv or *.txt file, (C) hypnogram with reported events. Each detection type is identified using different symbols and colors.

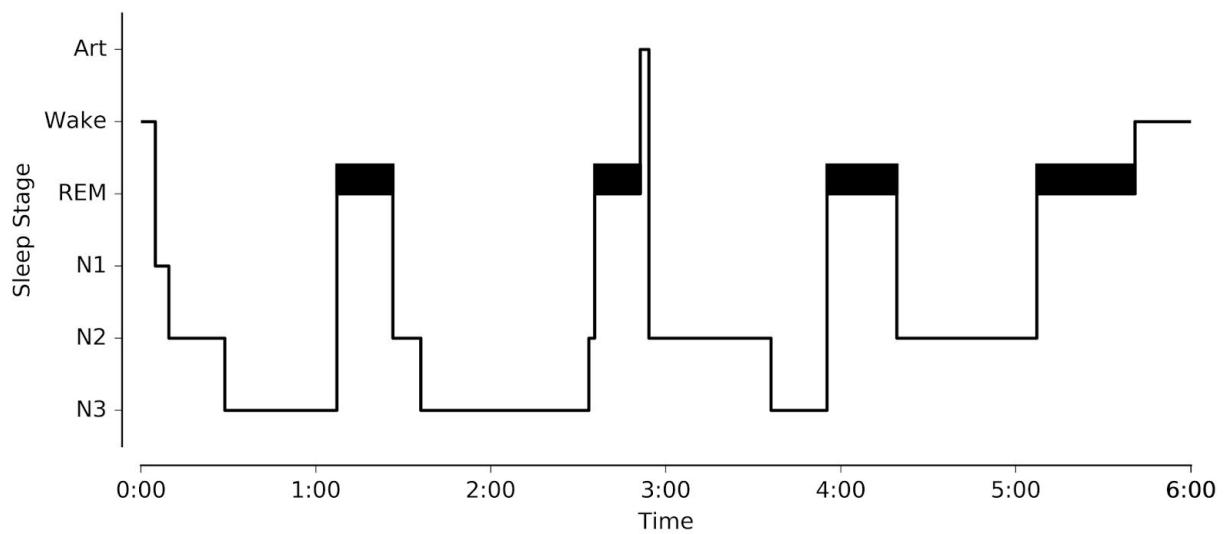


Figure 5 : Example of high-resolution publication-ready hypnogram figure exported using *Sleep GUI*.

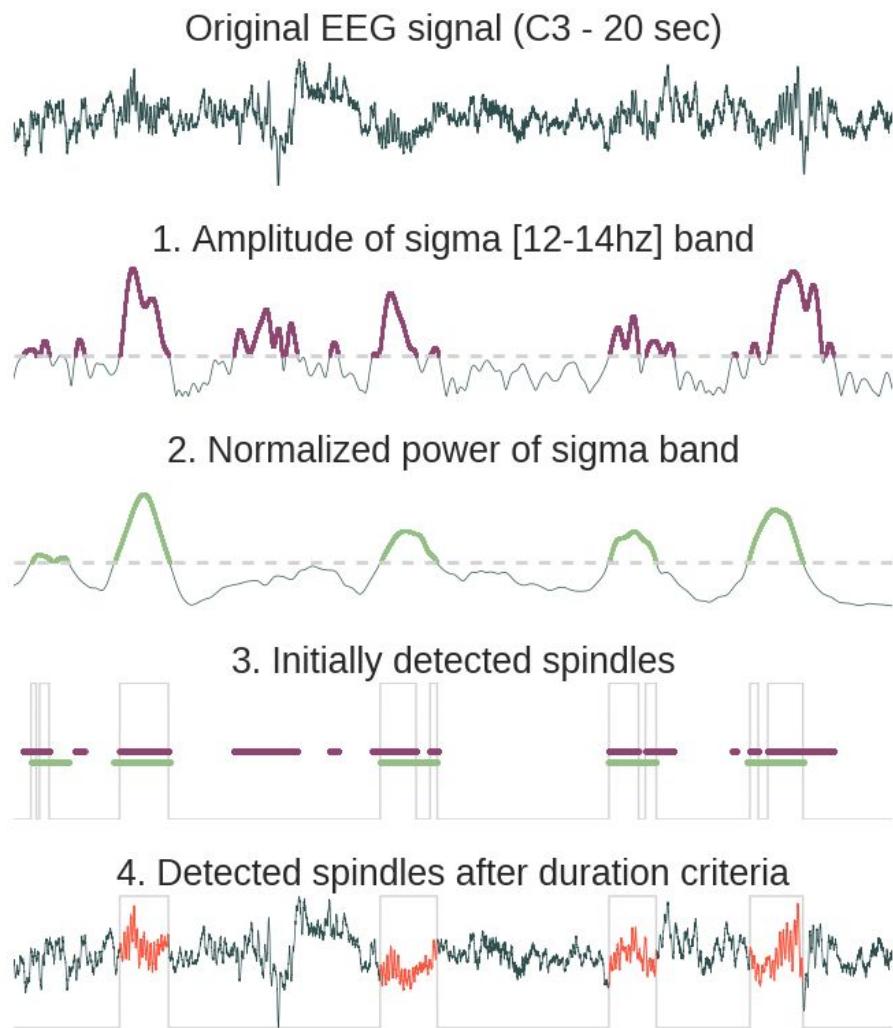


Figure 6. Method for the automatic sleep spindles detection. First, the original signal was convoluted with a Morlet wavelet centered in the spindles frequency band [12-14 Hz]. From the resulting complex decomposition, we kept only the amplitude and find time indices where the amplitude exceeded the threshold (purple in 1.). Then, we computed the normalized power in the sigma band and detected again time index where the power exceed a threshold (green in 2.). The time location of the initial detected spindles (gray line in 3.) is the result of the intersection of exceeding both the amplitude index (purple ligne) and the power index (green line). Finally, time gaps are filled only for proximate detected events (<500 ms) and a final duration criteria is applied in order to suppress events with a duration inferior to 500 ms or superior to 2000 ms (these thresholds are definable within *Sleep interface*, 4.).

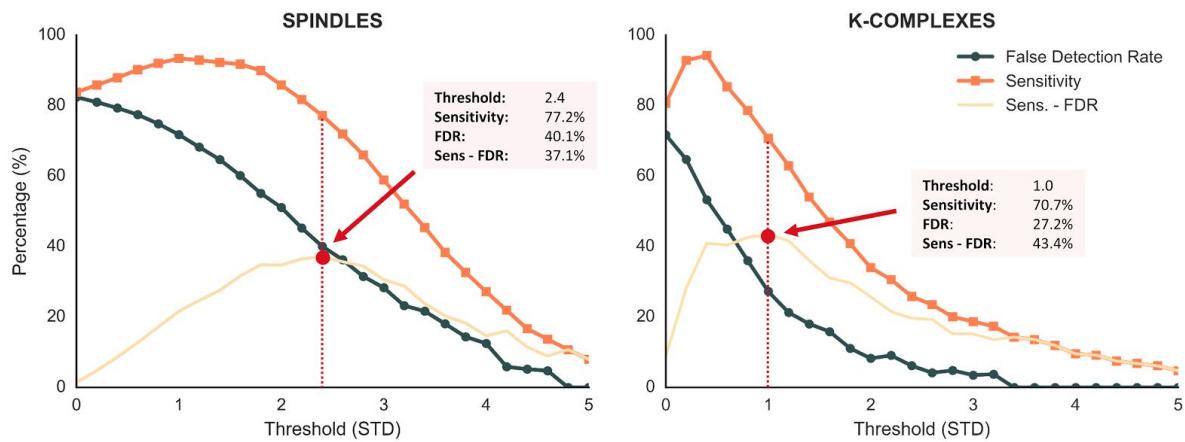


Figure 7. Performance metrics of our spindles and K-complexes detection methods evaluated at 25 different thresholds (range=0-5, step=0.2). Dark orange and blue lines depict the sensitivity and false detection rate (FDR), respectively. Light orange lines show the difference between sensitivity and FDR. Red dotted lines depict the threshold values that maximized this difference.

Code snippets

```
# Load the Sleep module from visbrain :  
from visbrain import Sleep  
  
# Open the default Sleep window :  
Sleep().show()
```

Code snippet 1: Simplest way to launch Sleep from a Python interpreter. This will open a window asking the user to select the EEG data and corresponding hypnogram.

```
# Import the Sleep module from visbrain :  
from visbrain import Sleep  
  
# Define where the data are located :  
dfile = '/home/perso/myfile.eeg'  
  
# Define where the hypnogram is located :  
hfile = '/home/perso/hypno.hyp'  
# hfile = None # Eventually, start from a fresh one  
  
# Inverse the default sleep stage order :  
norder = ['n3', 'n2', 'n1', 'rem', 'wake', 'art']  
  
# Finally, pass both file to the class :  
Sleep(file=dfile, hypno_file=hfile, href=norder).show()
```

Code snippet 2 : In this example, the paths to the EEG data and hypnogram are entered as inputs arguments of the main Sleep function, resulting in the software opening directly with the dataset and hypnogram loaded. We also show how to change the default display order of the hypnogram by changing the href argument of *Sleep* main function. The sleep stages will be displayed in the order defined in *norder* variable, with N3 on top and Art on bottom.

```

# Import the Sleep module and MNE:
from visbrain import Sleep
from mne import io

# - Biosemi Data Format (BDF)
raw = io.read_raw_edf('mybdffile.bdf', preload=True)

# - EGI format
# raw = io.read_raw_egi('myegifile.cgi', preload=True)

# - EEGLab
# raw = io.read_raw_eeglab('myeeglabfile.set', preload=True)

# Extract data, sampling frequency and channels names
data, sf, chan = raw._data, raw.info['sfreq'], raw.info['ch_names']

# Now, pass all the arguments to the Sleep module :
Sleep(data=data, sf=sf, channels=chan).show()

```

Code snippet 3 : This example shows a method to pass data to *Sleep* after loading them using MNE-Python package (see <http://martinos.org/mne/dev/manual/io.html> for a full list of the data formats supported by MNE).

References

- Berthomier, C., Drouot, X., Herman-Stoica, M., Berthomier, P., Prado, J., Bokar-Thire, D., et al. (2007). Automatic analysis of single-channel sleep EEG: validation in healthy individuals. *Sleep* 30, 1587–1595.
- Billinger, M., Brunner, C., and Müller-Putz, G. R. SCoT: a Python toolbox for EEG source connectivity. Available at: <https://pdfs.semanticscholar.org/b196/7f587fbea9ecf4cb6be3f757a8136fc60ca8.pdf>.
- Campagnola, L., Klein, A., Larson, E., Rossant, C., and Rougier, N. P. (2015). VisPy: Harnessing The GPU For Fast, High-Level Visualization. in *Proceedings of the 14th Python in Science Conference* Available at: <https://hal.inria.fr/hal-01208191/> [Accessed May 23, 2017].
- Combrisson, E., and Jerbi, K. (2015). Exceeding chance level by chance: The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy. *Journal of Neuroscience Methods* 250, 126–136. doi:10.1016/j.jneumeth.2015.01.010.
- Combrisson, E., Perrone-Bertolotti, M., Soto, J. L., Alamian, G., Kahane, P., Lachaux, J.-P., et al. (2017). From intentions to actions: Neural oscillations encode motor processes through phase, amplitude and phase-amplitude coupling. *NeuroImage* 147, 473–487. doi:10.1016/j.neuroimage.2016.11.042.
- Devuyst, S., Dutoit, T., Stenuit, P., and Kerkhofs, M. (2011). Automatic sleep spindles detection—overview and development of a standard proposal assessment method. in *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE* (IEEE), 1713–1716.
- Eichenlaub, J.-B., Bertrand, O., Morlet, D., and Ruby, P. (2014). Brain reactivity differentiates subjects with high and low dream recall frequencies during both sleep and wakefulness. *Cerebral Cortex* 24, 1206–1215.
- Eichenlaub, J.-B., Ruby, P., and Morlet, D. (2012). What is the specificity of the response to the own first-name when presented as a novel in a passive oddball paradigm? An ERP study. *Brain research* 1447, 65–78.
- Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., et al. (2013). MEG and EEG data analysis with MNE-Python. *Frontiers in neuroscience* 7, 267.

Iber, C., Ancoli-Israel, S., Chesson, A., Quan, S. F., and others (2007). *The AASM manual for the scoring of sleep and associated events: rules, terminology and technical specifications*. American Academy of Sleep Medicine Westchester, IL.

Jerbi, K., Ossandón, T., Hamamé, C. M., Senova, S., Dalal, S. S., Jung, J., et al. (2009). Task-related gamma-band dynamics from an intracerebral perspective: Review and implications for surface EEG and MEG. *Human Brain Mapping* 30, 1758–1771. doi:10.1002/hbm.20750.

Lajnef, T., Chaibi, S., Eichenlaub, J.-B., Ruby, P. M., Aguera, P.-E., Samet, M., et al. (2015a). Sleep spindle and K-complex detection using tunable Q-factor wavelet transform and morphological component analysis. *Frontiers in human neuroscience* 9.

Lajnef, T., Chaibi, S., Ruby, P., Aguera, P.-E., Eichenlaub, J.-B., Samet, M., et al. (2015b). Learning machines and sleeping brains: Automatic sleep stage classification using decision-tree multi-class support vector machines. *Journal of Neuroscience Methods* 250, 94–105. doi:10.1016/j.jneumeth.2015.01.022.

Rechtschaffen, A., and Kales, A. (1968). A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects.

Ruby, P., Blochet, C., Eichenlaub, J.-B., Bertrand, O., Morlet, D., and Bidet-Caulet, A. (2013a). Alpha reactivity to complex sounds differs during REM sleep and wakefulness. *PloS one* 8, e79989.

Ruby, P. M., Blochet, C., Eichenlaub, J.-B., Bertrand, O., Morlet, D., and Bidet-Caulet, A. (2013b). Alpha reactivity to first names differs in subjects with high and low dream recall frequency. *Frontiers in psychology* 4, 419.

Tallon-Baudry, C., Bertrand, O., Delpuech, C., and Pernier, J. (1996). Stimulus specificity of phase-locked and non-phase-locked 40 Hz visual responses in human. *The Journal of Neuroscience* 16, 4240–4249.

Vallat, R., Lajnef, T., Eichenlaub, J.-B., Berthomier, C., Jerbi, K., Morlet, D., et al. (2017). Increased Evoked Potentials to Arousing Auditory Stimuli during Sleep: Implication for the Understanding of Dream Recall. *Frontiers in Human Neuroscience* 11.

DISCUSSION GÉNÉRALE

Enfin : la conclusion générale!!!

Au cours de ce mémoire, nous avons développé un modèle ...

PERSPECTIVES

Dans la continuité directe de notre travail de thèse, nous pouvons ...

A ANNEXES

A.1 CARTES DES INTERFACES CERVEAU-MACHINE (GRAIMANN ET AL., 2009)

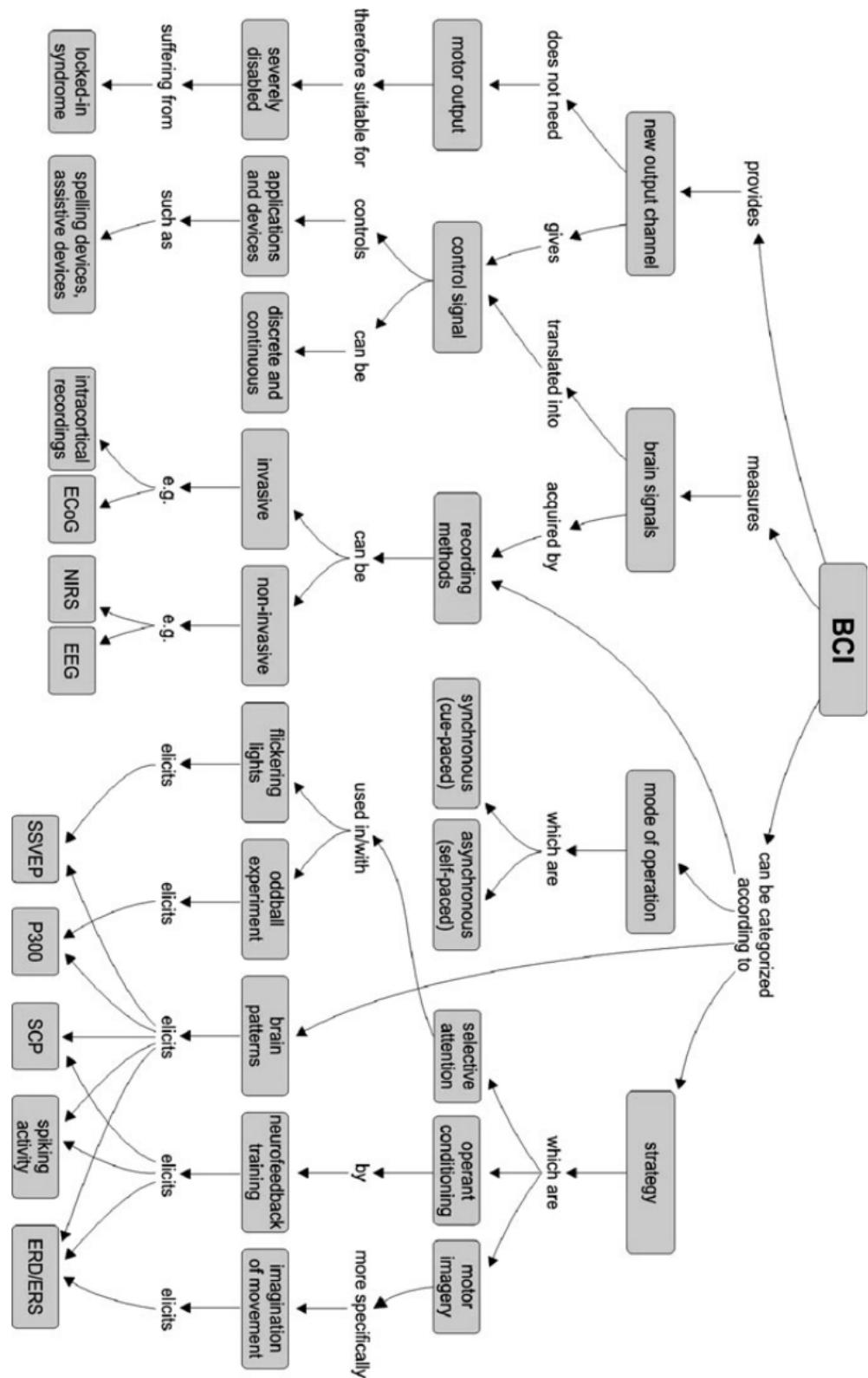


FIGURE A.1 – Cartes des Interfaces Cerveau-Machine (Graimann et al., 2009)

A.2 JEUX DE DONNÉES EN LIBRE ACCÈS (*BCI competition*)

BCI Competition II			
Set	N-Classes	Channel	Challenge
Ia/Ib	2	6-EEG	Decide whether the subject tried to produce cortical negativity or cortical positivity
IIa	4	64-EEG	Provide the intended target of the feedback test trials
IIb	36	64-EEG	Estimate to which letter of a 6-by-6 matrix with successively intensified rows resp. columns the subject was paying attention to
III	2	3-EEG	Provide a continuous output that could be used for a BCI- feedback
IV	2	28-EEG	Predict the laterality of upcoming finger movements (left vs. right hand) 130 ms before key-press
BCI Competition III			
I	2	64-ECoG	Cued motor imagery (left pinky, tongue) from one subject
II	36	64-EEG	Estimate to which letter of a 6-by-6 matrix with successively intensified rows resp. columns the subject was paying attention to
IIIa	4	60-EEG	Cued motor imagery with 4 classes (left hand, right hand, foot, tongue) from 3 subjects. Measure : kappa-coefficient
IIIb	2	2-EEG	Cued motor imagery with online feedback with 2 classes (left hand, right hand). Measure : mutual information
IVa/IVb/IVc	2	118-EEG	Cued motor imagery with 2 classes (right hand, foot) from 5 subjects
V	3	32-EEG	Cued mental imagery with 3 classes (left hand, right hand, word association) from 3 subjects
BCI Competition IV			
I	2	64-EEG	Motor imagery (2 classes of left hand, right hand, foot)
IIa	4	22-EEG	Cued motor imagery (left hand, right hand, feet, tongue)
IIb	2	3-EEG	Cued motor imagery (left hand, right hand)
III	4	10-MEG	Decoding directions of finger/hand/wrist movements
IV	5	64-ECoG	Discrimination of movements of individual fingers

FIGURE A.2 – Jeux de données en libre accès (*BCI competition*)

A.3 COMPARATIF DE MÉTHODES PAC ([TORT ET AL., 2010](#))

TABLE 1. *Summary of characteristics of the phase-amplitude coupling measures studied*

Phase-Amplitude Coupling Measure	Tolerance to Noise	Amplitude Independent	Sensitivity to Multimodality	Sensitivity to Modulation Width
Modulation index	Good	Yes	Good	Good
Heights ratio	Good	Yes	No discrimination	No
Mean vector length	Good	No	Restricted	Reasonable
Amplitude PSD	Low	No	Restricted	Good
Phase-locking value	Low	No*	Restricted	Low
Correlation measure	Low	No*	Restricted	Low
GLM measure	Low	No*	Restricted	Low
Coherence value	Low	No*	Restricted	Low

* Under the presence of noise.

FIGURE A.3 – Comparatif de méthodes PAC ([Tort et al., 2010](#))

A.4 PIPELINE STANDARD DE CLASSIFICATION

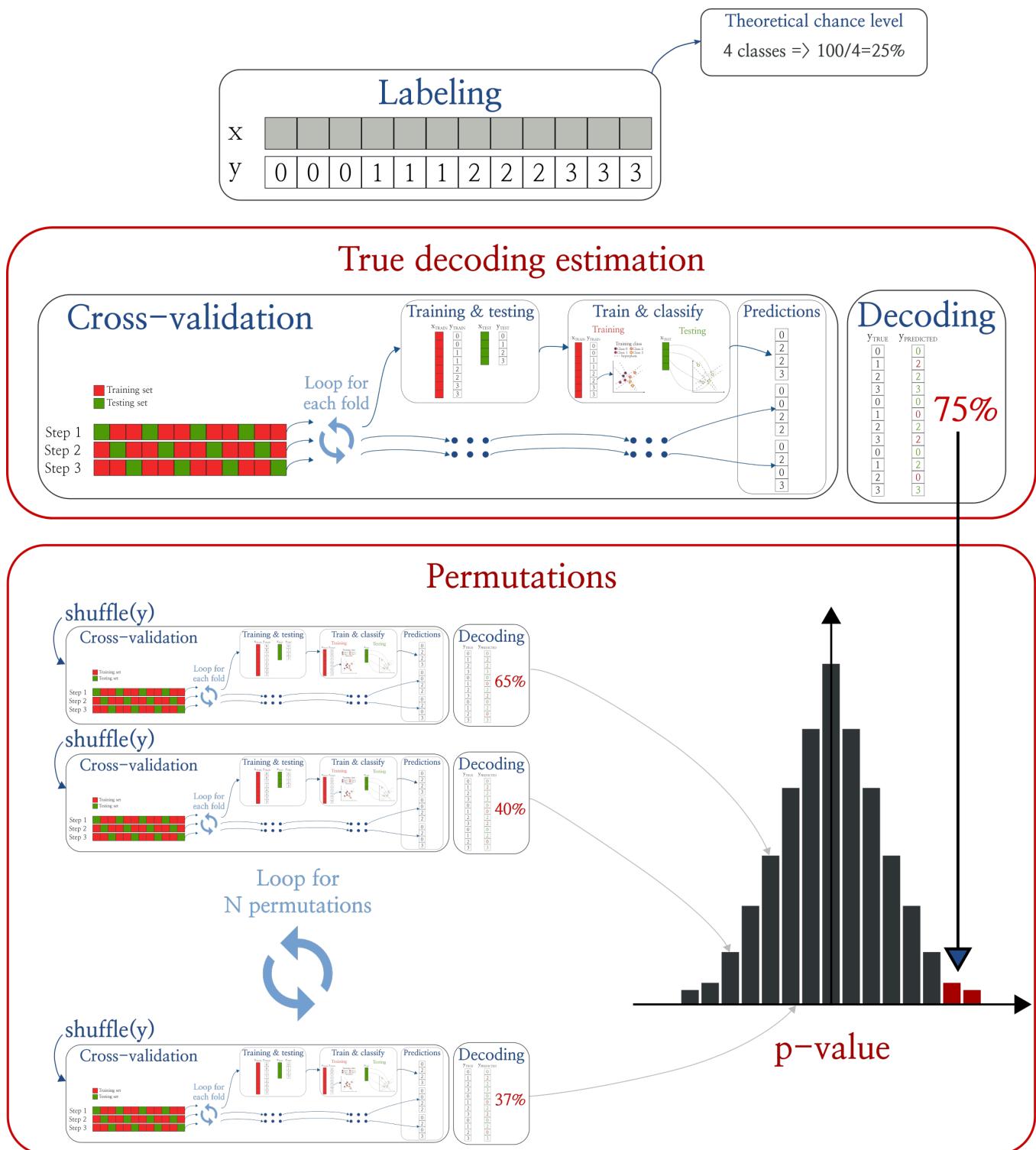


FIGURE A.4 – Pipeline standard de classification

A.5 COMPARATIF DE CLASSIFIERS (PEDREGOSA ET AL., 2011)

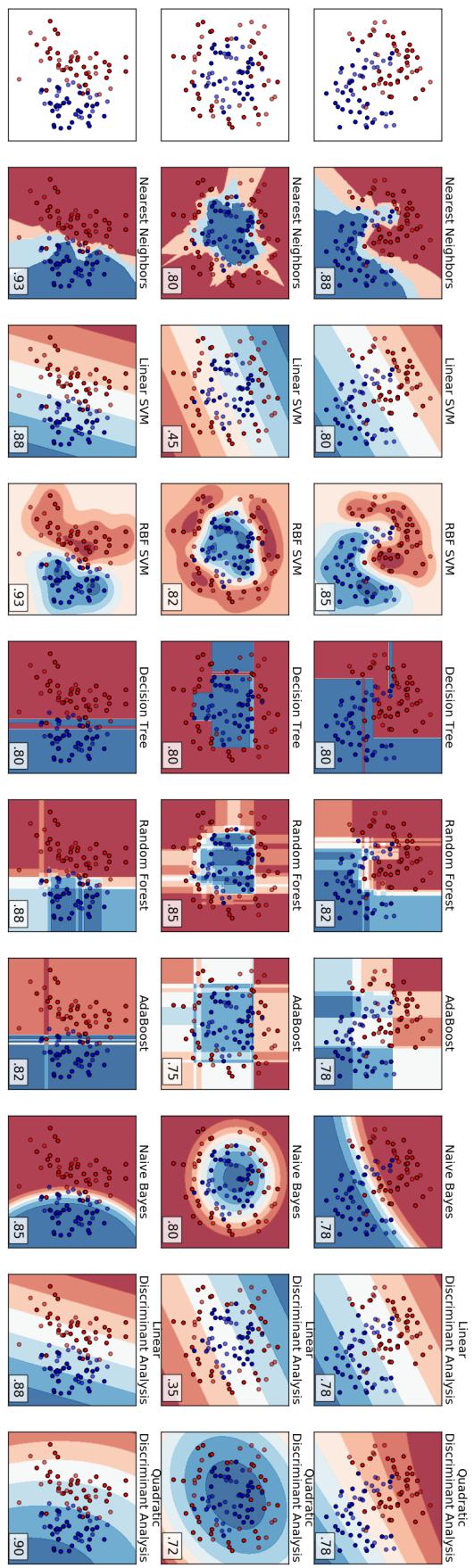
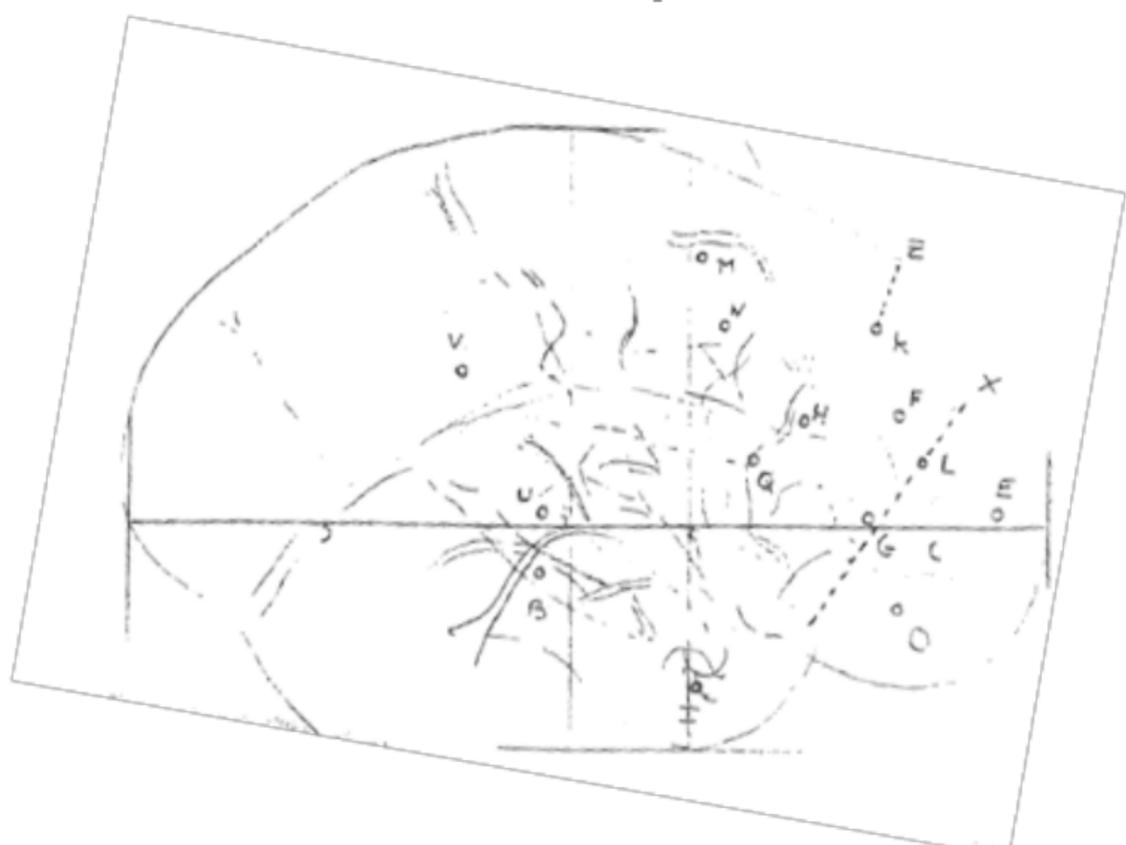
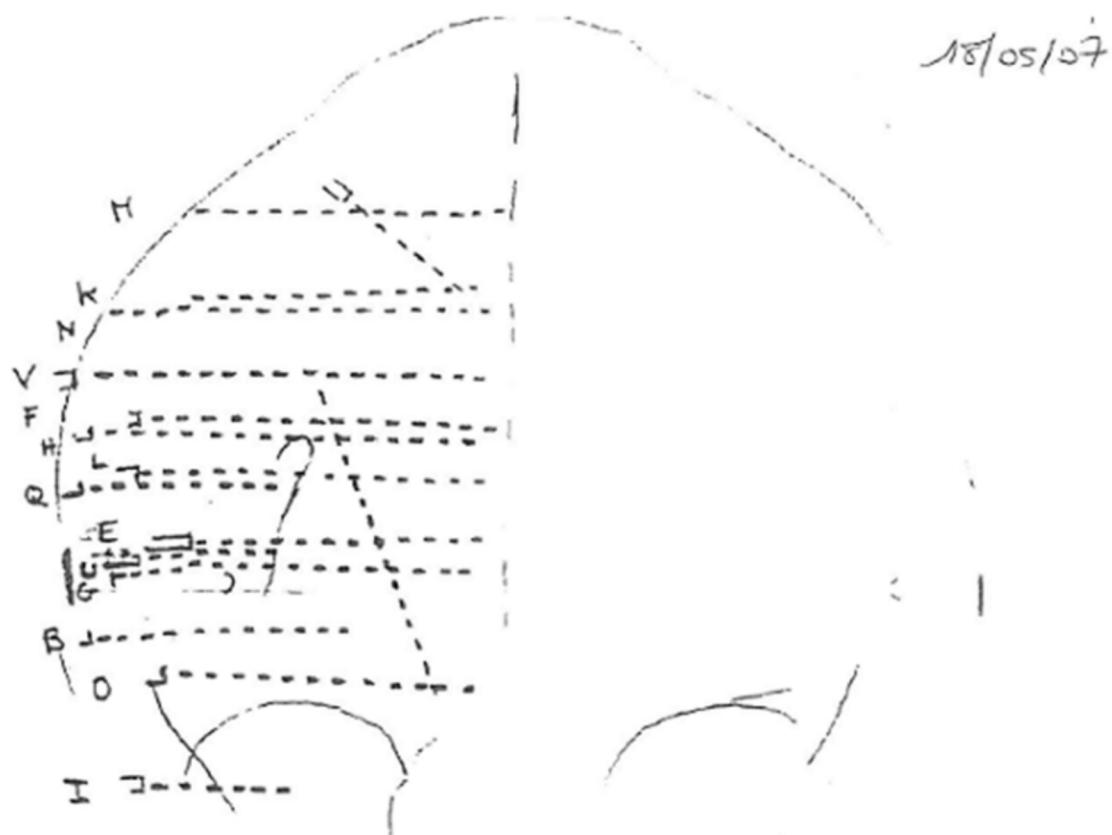


FIGURE A.5 – Comparatif de classifieurs (Pedregosa et al., 2011)

A.6 EXEMPLE DE SCHÉMA D'IMPLANTATION



BIBLIOGRAPHIE

- Acharya, S., Fifer, M. S., Benz, H. L., Crone, N. E., and Thakor, N. V. (2010). Electrocorticographic amplitude predicts finger positions during slow grasping motions of the hand. *Journal of neural engineering*, 7(4) :046002.
- Allison, B. Z., McFarland, D. J., Schalk, G., Zheng, S. D., Jackson, M. M., and Wolpaw, J. R. (2008). Towards an independent brain-computer interface using steady state visual evoked potentials. *Clinical neurophysiology*, 119(2) :399–408.
- Amzica, F. and Steriade, M. (1998). Electrophysiological correlates of sleep delta waves. *Electroencephalography and Clinical Neurophysiology*, 107(2) :69–83.
- Aru, J., Aru, J., Priesemann, V., Wibral, M., Lana, L., Pipa, G., Singer, W., and Vicente, R. (2015). Untangling cross-frequency coupling in neuroscience. *Current Opinion in Neurobiology*, 31 :51–61.
- Bahramisharif, A., van Gerven, M. A. J., Aarnoutse, E. J., Mercier, M. R., Schwartz, T. H., Foxe, J. J., Ramsey, N. F., and Jensen, O. (2013). Propagating Neocortical Gamma Bursts Are Coordinated by Traveling Alpha Waves. *Journal of Neuroscience*, 33(48) :18849–18854.
- Ball, T., Schreiber, A., Feige, B., Wagner, M., Lücking, C. H., and Kristeva-Feige, R. (1999). The role of higher-order motor areas in voluntary movement as revealed by high-resolution EEG and fMRI. *Neuroimage*, 10(6) :682–694.
- Ball, T., Schulze-Bonhage, A., Aertsen, A., and Mehring, C. (2009). Differential representation of arm movement direction in relation to cortical anatomy and function. *Journal of Neural Engineering*, 6(1) :016006.
- Bastin, J., Deman, P., David, O., Gueguen, M., Benis, D., Minotti, L., Hoffman, D., Combrisson, E., Kujala, J., Perrone-Bertolotti, M., Kahane, P., Lachaux, J.-P., and Jerbi, K. (2016). Direct Recordings from Human Anterior Insula Reveal its Leading Role within the Error-Monitoring Network. *Cerebral Cortex*, page bhv352.
- Bekaert, M. H., Botte-Lecocq, C., Cabestaing, F., Rakotomamonjy, A., and others (2009). Les interfaces Cerveau-Machine pour la palliation du handicap moteur sévère. *Sciences et Technologies pour le Handicap*, 3(1) :95–121.
- Berens, P. and others (2009). CircStat a MATLAB toolbox for circular statistics. *J Stat Softw*, 31(10) :1–21.
- Berger, H. (1929). Ueber das elektroenkephalogramm des menschen. *Archiv für Psychiatrie und Nervenkrankheiten*, 87(1) :527–570.
- Bertrand, O., Bohorquez, J., and Pernier, J. (1994). Time-frequency digital filtering based on an invertible wavelet transform : An application to evoked potentials. *IEEE Transactions on Biomedical Engineering*, 41(1) :77–88.

- Besserve, M. (2007). *Analyse de la dynamique neuronale pour les Interfaces Cerveau-Machines : un retour aux sources*. PhD thesis, Université Paris Sud - Paris XI.
- Besserve, M., Jerbi, K., Laurent, F., Baillet, S., Martinerie, J., Garnero, L., and others (2007). Classification methods for ongoing EEG and MEG signals. *Biol Res*, 40(4) :415–437.
- Beverina, F., Palmas, G., Silvoni, S., Piccione, F., and Giove, S. (2003). User adaptive BCIs : SSVEP and P300 based interfaces. *PsychNology Journal*, 1(4) :331–354.
- Birbaumer, N. (1997). Slow cortical potentials : Their origin, meaning, and clinical use. *Brain and behavior past, present, and future*, pages 25–39.
- Birbaumer, N. (1999). Slow cortical potentials : Plasticity, operant control, and behavioral effects. *The Neuroscientist*, 5(2) :74–78.
- Birbaumer, N. (2006). Breaking the silence : Brain computer interfaces (BCI) for communication and motor control. *Psychophysiology*, 43(6) :517–532.
- Birbaumer, N., Elbert, T., Canavan, A. G., and Rockstroh, B. (1990). Slow potentials of the cerebral cortex and behavior. *Physiological reviews*, 70(1) :1–41.
- Birbaumer, N., Ghanayim, N., Hinterberger, T., Iversen, I., Kotchoubey, B., Kübler, A., Perelmouter, J., Taub, E., and Flor, H. (1999). A spelling device for the paralysed. *Nature*, 398(6725) :297–298.
- Birbaumer, N., Hinterberger, T., Kubler, A., and Neumann, N. (2003). The thought-translation device (ttd) : Neurobehavioral mechanisms and clinical outcome. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2) :120–123.
- Birbaumer, N., Kubler, A., Ghanayim, N., Hinterberger, T., Perelmouter, J., Kaiser, J., Iversen, I., Kotchoubey, B., Neumann, N., and Flor, H. (2000). The thought translation device (TTD) for completely paralyzed patients. *IEEE Transactions on Rehabilitation Engineering*, 8(2) :190–193.
- Blankertz, B., Krauledat, M., Dornhege, G., Williamson, J., Murray-Smith, R., and Müller, K.-R. (2007). A note on brain actuated spelling with the Berlin brain-computer interface. In *International Conference on Universal Access in Human-Computer Interaction*, pages 759–768. Springer.
- Blankertz, B., Muller, K., Krusienski, D., Schalk, G., Wolpaw, J., Schlogl, A., Pfurtscheller, G., Millan, J., Schroder, M., and Birbaumer, N. (2006). The BCI Competition III : Validating Alternative Approaches to Actual BCI Problems. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2) :153–159.
- Blankertz, B., Muller, K. R., Curio, G., Vaughan, T. M., Schalk, G., Wolpaw, J. R., Schlogl, A., Neuper, C., Pfurtscheller, G., Hinterberger, T., and others (2004). The BCI competition 2003 : Progress and perspectives in detection and discrimination of EEG single trials. *Biomedical Engineering, IEEE Transactions on*, 51(6) :1044–1051.
- Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pages 144–152. ACM.
- Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7) :1145–1159.

- Brouwer, A.-M. and Van Erp, J. B. (2010). A tactile P300 brain-computer interface. *Frontiers in neuroscience*, 4 :19.
- Brunia, C. and Van Boxtel, G. (2000). Motor preparation.
- Brunner, P., Ritaccio, A. L., Emrich, J. F., Bischof, H., and Schalk, G. (2011). Rapid communication with a “P300” matrix speller using electrocorticographic signals (ECOG). *Frontiers in neuroscience*, 5 :5.
- Bruns, A. and Eckhorn, R. (2004). Task-related coupling from high-to low-frequency signals among visual cortical areas in human subdural recordings. *International Journal of Psychophysiology*, 51(2) :97–116.
- Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2) :121–167.
- Campagnola, L., Klein, A., Rossant, C., and Rougier, N. (2013). VISPY, A Modern and Interactive Scientific Visualisation. In *Euroscipy 2013*.
- Canolty, R. T., Edwards, E., Dalal, S. S., Soltani, M., Nagarajan, S. S., Kirsch, H. E., Berger, M. S., Barbaro, N. M., and Knight, R. T. (2006). High Gamma Power Is Phase-Locked to Theta Oscillations in Human Neocortex. *Science*, 313(5793) :1626–1628.
- Canolty, R. T. and Knight, R. T. (2010). The functional role of cross-frequency coupling.
- Cassim, F., Monaca, C., Szurhaj, W., Bourriez, J.-L., Defebvre, L., Derambure, P., and Guieu, J.-D. (2001). Does post-movement beta synchronization reflect an idling motor cortex? *Neuroreport*, 12(17) :3859–3863.
- Cecotti, H. (2010). A self-paced and calibration-less SSVEP-based brain–computer interface speller. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2) :127–133.
- Chai, H. and Domeniconi, C. (2004). An evaluation of gene selection methods for multi-class microarray data classification. In *Proceedings of the Second European Workshop on Data Mining and Text Mining in Bioinformatics*, pages 3–10.
- Chang, C.-C. and Lin, C.-J. (2011). LIBSVM a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3) :27.
- Chapin, J. K., Moxon, K. A., Markowitz, R. S., and Nicolelis, M. A. (1999). Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. *Nature neuroscience*, 2(7) :664–670.
- Chaudhary, U., Birbaumer, N., and Curado, M. (2015). Brain-Machine Interface (BMI) in paralysis. *Annals of Physical and Rehabilitation Medicine*, 58(1) :9–13.
- Cohen, M. X. (2008). Assessing transient cross-frequency coupling in EEG data. *Journal of Neuroscience Methods*, 168(2) :494–499.
- Cohen, M. X., Elger, C. E., and Fell, J. (2008). Oscillatory activity and phase–amplitude coupling in the human medial frontal cortex during decision making. *Journal of cognitive neuroscience*, 21(2) :390–402.

- Collinger, J. L., Wodlinger, B., Downey, J. E., Wang, W., Tyler-Kabara, E. C., Weber, D. J., McMorland, A. J., Velliste, M., Boninger, M. L., and Schwartz, A. B. (2013). High performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 381(9866) :557–564.
- Combrisson, E. and Jerbi, K. (2015). Exceeding chance level by chance : The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy. *Journal of Neuroscience Methods*, 250 :126–136.
- Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3) :273–297.
- Coyle, D., Garcia, J., Satti, A. R., and McGinnity, T. M. (2011). EEG-based continuous control of a game using a 3 channel motor imagery BCI : BCI game. In *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium On*, pages 1–7. IEEE.
- Coyle, S., Ward, T., Markham, C., and McDarby, G. (2004). On the suitability of near-infrared (NIR) systems for next-generation brain–computer interfaces. *Physiological Measurement*, 25(4) :815–822.
- Crone, N. (1998). Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis. I. Alpha and beta event- related desynchronization. *Brain*, 121(12) :2271–2299.
- Crone, N. E., Miglioretti, D. L., Gordon, B., and Lesser, R. P. (1998). Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis. II. Event-related synchronization in the gamma band. *Brain*, 121(12) :2301–2315.
- Curran, E. (2003). Learning to control brain activity : A review of the production and control of EEG components for driving brain–computer interface (BCI) systems. *Brain and Cognition*, 51(3) :326–336.
- Das, S. (2001). Filters, wrappers and a boosting-based hybrid for feature selection. In *ICML*, volume 1, pages 74–81. Citeseer.
- de Hemptinne, C., Ryapolova-Webb, E. S., Air, E. L., Garcia, P. A., Miller, K. J., Ojemann, J. G., Ostrem, J. L., Galifianakis, N. B., and Starr, P. A. (2013). Exaggerated phase–amplitude coupling in the primary motor cortex in Parkinson disease. *Proceedings of the National Academy of Sciences*.
- de Vries, S., Tepper, M., Otten, B., and Mulder, T. (2011). Recovery of Motor Imagery Ability in Stroke Patients. *Rehabilitation Research and Practice*, 2011 :1–9.
- Deiber, M.-P., Honda, M., Ibañez, V., Sadato, N., and Hallett, M. (1999). Mesial motor areas in self-initiated versus externally triggered movements examined with fMRI : Effect of movement type and rate. *Journal of neurophysiology*, 81(6) :3065–3077.
- Demandt, E., Mehring, C., Vogt, K., Schulze-Bonhage, A., Aertsen, A., and Ball, T. (2012). Reaching Movement Onset- and End-Related Characteristics of EEG Spectral Power Modulations. *Frontiers in Neuroscience*, 6.
- Di Rienzo, F., Collet, C., Hoyek, N., and Guillot, A. (2014). Impact of Neurologic Deficits on Motor Imagery : A Systematic Review of Clinical Evaluations. *Neuropsychology Review*, 24(2) :116–147.

- Di Rienzo, F., Debarnot, U., Daligault, S., Saruco, E., Delpuech, C., Doyon, J., Collet, C., and Guillot, A. (2016). Online and Offline Performance Gains Following Motor Imagery Practice : A Comprehensive Review of Behavioral and Neuroimaging Studies. *Frontiers in Human Neuroscience*, 10.
- Diez, P. F., Torres Müller, S. M., Mut, V. A., Laciár, E., Avila, E., Bastos-Filho, T. F., and Sarcinelli-Filho, M. (2013). Commanding a robotic wheelchair with a high-frequency steady-state visual evoked potential based brain-computer interface. *Medical Engineering & Physics*, 35(8) :1155–1164.
- Ding, C. and Peng, H. (2005). Minimum Redundancy Feature Selection from Microarray Gene Expression Data.
- Donchin, E., Spencer, K., and Wijesinghe, R. (2000). The mental prosthesis : Assessing the speed of a P300-based brain-computer interface. *IEEE Transactions on Rehabilitation Engineering*, 8(2) :174–179.
- Donoghue, J. P. (2002). Connecting cortex to machines : Recent advances in brain interfaces. *Nature Neuroscience*, 5(Supp) :1085–1088.
- Doud, A. J., Lucas, J. P., Pisansky, M. T., and He, B. (2011). Continuous three-dimensional control of a virtual helicopter using a motor imagery based brain-computer interface. *PloS one*, 6(10) :e26322.
- Driskell, J. E., Copper, C., and Moran, A. (1994). Does mental practice enhance performance? *Journal of applied psychology*, 79(4) :481.
- Duda, R. O., Hart, P. E., and Stork, D. G. (2001). Pattern classification. 2nd. Edition. New York.
- Dvorak, D. and Fenton, A. A. (2014). Toward a proper estimation of phase-amplitude coupling in neural oscillations. *Journal of Neuroscience Methods*, 225 :42–56.
- Efron, B. and Tibshirani, R. J. (1994). *An Introduction to the Bootstrap*. CRC press.
- Engel, A. K., Moll, C. K. E., Fried, I., and Ojemann, G. A. (2005). Invasive recordings from the human brain : Clinical insights and beyond. *Nature Reviews Neuroscience*, 6(1) :35–47.
- Farwell, L. and Donchin, E. (1988). Talking off the top of your head : Toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70(6) :510–523.
- Fazli, S., Mehnert, J., Steinbrink, J., Curio, G., Villringer, A., Müller, K.-R., and Blankertz, B. (2012). Enhanced performance by a hybrid NIRS-EEG brain computer interface. *Neuroimage*, 59(1) :519–529.
- Felton, E. A., Wilson, J. A., Williams, J. C., and Garell, P. C. (2007). Electrocorticographically controlled brain-computer interfaces using motor and sensory imagery in patients with temporary subdural electrode implants : Report of four cases. *Journal of neurosurgery*, 106(3) :495–500.
- Fetz, E. E. (1969). Operant conditioning of cortical unit activity. *Science (New York, N.Y.)*, 163(3870) :955–958.

- Fetz, E. E. and others (1999). Real-time control of a robotic arm by neuronal ensembles. *nature neuroscience*, 2 :583–584.
- Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2) :179–188.
- Fix, E. and Hodges Jr, J. L. (1951). Discriminatory analysis-nonparametric discrimination : Consistency properties. Technical report, DTIC Document.
- Fries, P. (2005). A mechanism for cognitive dynamics : Neuronal communication through neuronal coherence. *Trends in Cognitive Sciences*, 9(10) :474–480.
- Friston, K. J. (1997). Another neural code ? *Neuroimage*, 5(3) :213–220.
- Fukunaga, K. (1990). Introduction to statistical pattern classification.
- Furdea, A., Halder, S., Krusienski, D., Bross, D., Nijboer, F., Birbaumer, N., and Kübler, A. (2009). An auditory oddball (P300) spelling system for brain-computer interfaces. *Psychophysiology*, 46(3) :617–625.
- Galán, F., Nuttin, M., Lew, E., Ferrez, P., Vanacker, G., Philips, J., and Millán, J. d. R. (2008a). A brain-actuated wheelchair : Asynchronous and non-invasive Brain–computer interfaces for continuous control of robots. *Clinical Neurophysiology*, 119(9) :2159–2169.
- Galán, F., Nuttin, M., Lew, E., Ferrez, P. W., Vanacker, G., Philips, J., and Millán, J. d. R. (2008b). A brain-actuated wheelchair : Asynchronous and non-invasive brain–computer interfaces for continuous control of robots. *Clinical Neurophysiology*, 119(9) :2159–2169.
- Garnero, L., Baillet, S., and Renault, B. (1998). Magnétoencéphalographie / électroencéphalographie et imagerie cérébrale fonctionnelle. *Annales de l'Institut Pasteur / Actualités*, 9(3) :215–226.
- Georgopoulos, A., Schwartz, A., and Kettner, R. (1986). Neuronal population coding of movement direction. *Science*, 233(4771) :1416–1419.
- Georgopoulos, A. P. and Carpenter, A. F. (2015). Coding of movements in the motor cortex. *Current Opinion in Neurobiology*, 33 :34–39.
- Georgopoulos, A. P., Kalaska, J. F., Caminiti, R., and Massey, J. T. (1982). On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *The Journal of Neuroscience*, 2(11) :1527–1537.
- Graimann, B., Allison, B., and Pfurtscheller, G. (2009). Brain–Computer Interfaces : A Gentle Introduction. In Graimann, B., Pfurtscheller, G., and Allison, B., editors, *Brain-Computer Interfaces*, pages 1–27. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Gregoriou, G. G., Gotts, S. J., Zhou, H., and Desimone, R. (2009). High-frequency, long-range coupling between prefrontal and visual cortex during attention. *Science (New York, N.Y.)*, 324(5931) :1207–1210.
- Guillot, A., Collet, C., Nguyen, V. A., Malouin, F., Richards, C., and Doyon, J. (2008). Functional neuroanatomical networks associated with expertise in motor imagery. *NeuroImage*, 41(4) :1471–1483.

- Guillot, A., Collet, C., Nguyen, V. A., Malouin, F., Richards, C., and Doyon, J. (2009). Brain activity during visual versus kinesthetic imagery : An fMRI study. *Human Brain Mapping*, 30(7) :2157–2172.
- Gunduz, A., Brunner, P., Sharma, M., Leuthardt, E. C., Ritaccio, A. L., Pesaran, B., and Schalk, G. (2016). Differential roles of high gamma and local motor potentials for movement preparation and execution. *Brain-Computer Interfaces*, 3(2) :88–102.
- Gunduz, A., Sanchez, J. C., Carney, P. R., and Principe, J. C. (2009). Mapping broadband electrocorticographic recordings to two-dimensional hand trajectories in humans : Motor control features. *Neural Networks*, 22(9) :1257–1270.
- Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3 :1157–1182.
- Hamadicharef, B., Xu, M., and Aditya, S. (2010). Brain-Computer Interface (BCI) based musical composition. In *Cyberworlds (CW), 2010 International Conference On*, pages 282–286. IEEE.
- Hammer, J., Fischer, J., Ruescher, J., Schulze-Bonhage, A., Aertsen, A., and Ball, T. (2013). The role of ECoG magnitude and phase in decoding position, velocity, and acceleration during continuous motor behavior. *Frontiers in Neuroscience*, 7.
- Hammer, J., Pistohl, T., Fischer, J., Kršek, P., Tomášek, M., Marusič, P., Schulze-Bonhage, A., Aertsen, A., and Ball, T. (2016). Predominance of Movement Speed Over Direction in Neuronal Population Signals of Motor Cortex : Intracranial EEG Data and A Simple Explanatory Model. *Cerebral Cortex*, 26(6) :2863–2881.
- Hammon, P., Makeig, S., Poizner, H., Todorov, E., and De Sa, V. (2008). Predicting Reaching Targets from Human EEG. *IEEE Signal Processing Magazine*, 25(1) :69–77.
- Hanakawa, T., Dimyan, M. A., and Hallett, M. (2008). Motor Planning, Imagery, and Execution in the Distributed Motor Network : A Time-Course Study with Functional MRI. *Cerebral Cortex*, 18(12) :2775–2788.
- Hochberg, L. R., Bacher, D., Jarosiewicz, B., Masse, N. Y., Simen, J. D., Vogel, J., Haddadin, S., Liu, J., Cash, S. S., van der Smagt, P., and Donoghue, J. P. (2012). Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485(7398) :372–375.
- Hochberg, L. R., Serruya, M. D., Friehs, G. M., Mukand, J. A., Saleh, M., Caplan, A. H., Branner, A., Chen, D., Penn, R. D., and Donoghue, J. P. (2006). Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 442(7099) :164–171.
- Hoffmann, U., Vesin, J.-M., Ebrahimi, T., and Diserens, K. (2008). An efficient P300-based brain-computer interface for disabled subjects. *Journal of Neuroscience methods*, 167(1) :115–125.
- Huang, J. and Ling, C. X. (2005). Using AUC and accuracy in evaluating learning algorithms. *Knowledge and Data Engineering, IEEE Transactions on*, 17(3) :299–310.
- Hyafil, A., Giraud, A.-L., Fontolan, L., and Gutkin, B. (2015). Neural Cross-Frequency Coupling : Connecting Architectures, Mechanisms, and Functions. *Trends in Neurosciences*, 38(11) :725–740.

- Jackson, P. L., Lafleur, M. F., Malouin, F., Richards, C., and Doyon, J. (2001). Potential role of mental practice using motor imagery in neurologic rehabilitation. *Archives of Physical Medicine and Rehabilitation*, 82(8) :1133–1141.
- Jackson, P. L., Meltzoff, A. N., and Decety, J. (2006). Neural circuits involved in imitation and perspective-taking. *NeuroImage*, 31(1) :429–439.
- Jankelowitz and Colebatch (2002). Movement-related potentials associated with self-paced, cued and imagined arm movements. *Experimental Brain Research*, 147(1) :98–107.
- Jenkins, I. H., Jahanshahi, M., Jueptner, M., Passingham, R. E., and Brooks, D. J. (2000). Self-initiated versus externally triggered movements. *Brain*, 123(6) :1216–1228.
- Jerbi, K., Freyermuth, S., Minotti, L., Kahane, P., Berthoz, A., and Lachaux, J.-P. (2009a). Chapter 12 Watching Brain TV and Playing Brain Ball. In *International Review of Neurobiology*, volume 86, pages 159–168. Elsevier.
- Jerbi, K., Ossandón, T., Hamamé, C. M., Senova, S., Dalal, S. S., Jung, J., Minotti, L., Bertrand, O., Berthoz, A., Kahane, P., and Lachaux, J.-P. (2009b). Task-related gamma-band dynamics from an intracerebral perspective : Review and implications for surface EEG and MEG. *Human Brain Mapping*, 30(6) :1758–1771.
- Jervis, B. W., Nichols, M. J., Johnson, T. E., Allen, E., and Hudson, N. R. (1983). A fundamental investigation of the composition of auditory evoked potentials. *IEEE transactions on bio-medical engineering*, 30(1) :43–50.
- Jiang, D., Edwards, M. G., Mullins, P., and Callow, N. (2015). The neural substrates for the different modalities of movement imagery. *Brain and Cognition*, 97 :22–31.
- Jobsis, F. F. (1977). Noninvasive, infrared monitoring of cerebral and myocardial oxygen sufficiency and circulatory parameters. *Science*, 198(4323) :1264–1267.
- Kahane, P., Landré, E., Minotti, L., Francione, S., and Ryvlin, P. (2006). The Bancaud and Talairach view on the epileptogenic zone : A working hypothesis. *Epileptic disorders : international epilepsy journal with videotape*, 8 :S16–26.
- Kaiser, V., Bauernfeind, G., Kreilinger, A., Kaufmann, T., Kübler, A., Neuper, C., and Müller-Putz, G. R. (2014). Cortical effects of user training in a motor imagery based brain–computer interface measured by fNIRS and EEG. *Neuroimage*, 85 :432–444.
- Kayagil, T. A., Bai, O., Henriquez, C. S., Lin, P., Furlani, S. J., Vorbach, S., and Hallett, M. (2009). A binary method for simple and accurate two-dimensional cursor control from EEG with minimal subject training. *Journal of neuroengineering and rehabilitation*, 6(1) :1.
- Kim, S.-P., Simeral, J. D., Hochberg, L. R., Donoghue, J. P., Friehs, G. M., and Black, M. J. (2011). Point-and-Click Cursor Control With an Intracortical Neural Interface System by Humans With Tetraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(2) :193–203.
- King, J.-R. and Dehaene, S. (2014). Characterizing the dynamics of mental representations : The temporal generalization method. *Trends in Cognitive Sciences*, 18(4) :203–210.

- Kleber, B. and Birbaumer, N. (2005). Direct brain communication : Neuroelectric and metabolic approaches at Tübingen. *Cognitive Processing*, 6(1) :65–74.
- Klimesch, W. (1999). EEG alpha and theta oscillations reflect cognitive and memory performance : A review and analysis. *Brain research reviews*, 29(2) :169–195.
- Klimesch, W., Sauseng, P., and Hanslmayr, S. (2007). EEG alpha oscillations : The inhibition-timing hypothesis. *Brain Research Reviews*, 53(1) :63–88.
- Kohavi, R. and others (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, volume 14, pages 1137–1145.
- Kornhuber, H. H. and Deecke, L. (1965). Hirnpotentialänderungen bei Willkürbewegungen und passiven Bewegungen des Menschen : Bereitschaftspotential und reafferente Potentiale. *Pflüger's Archiv für die gesamte Physiologie des Menschen und der Tiere*, 284(1) :1–17.
- Kosslyn, S. M., Seger, C., Pani, J. R., and Hillger, L. A. (1990). When is imagery used in everyday life ? A diary study. *Journal of Mental Imagery*.
- Kotchoubey, B., Strehl, U., Holzapfel, S., Schneider, D., Blankenhorn, V., and Birbaumer, N. (1998). Control of cortical excitability in epilepsy. *Advances in neurology*, 81 :281–290.
- Krusienski, D. J. and Shih, J. J. (2011a). Control of a brain-computer interface using stereotactic depth electrodes in and adjacent to the hippocampus. *Journal of Neural Engineering*, 8(2) :025006.
- Krusienski, D. J. and Shih, J. J. (2011b). Control of a visual keyboard using an electrocorticographic brain-computer interface. *Neurorehabilitation and neural repair*, 25(4) :323–331.
- Kübler, A., Kotchoubey, B., Hinterberger, T., Ghanayim, N., Perelmouter, J., Schauer, M., Fritsch, C., Taub, E., and Birbaumer, N. (1999). The thought translation device : A neurophysiological approach to communication in total motor paralysis. *Experimental brain research*, 124(2) :223–232.
- Kübler, A., Kotchoubey, B., Kaiser, J., Wolpaw, J. R., and Birbaumer, N. (2001). Brain computer communication : Unlocking the locked in. *Psychological bulletin*, 127(3) :358.
- Lachaux, J., Rudrauf, D., and Kahane, P. (2003). Intracranial EEG and human brain mapping. *Journal of Physiology-Paris*, 97(4-6) :613–628.
- Lachaux, J.-P., Jerbi, K., Bertrand, O., Minotti, L., Hoffmann, D., Schoendorff, B., and Kahane, P. (2007). BrainTV a novel approach for online mapping of human brain functions. *Biological research*, 40(4) :401–413.
- Lachaux, J.-P., Rodriguez, E., Le Van Quyen, M., Lutz, A., Martinerie, J., and Varela, F. J. (2000). Studying single trials of phase synchronous activity in the brain. *International Journal of Bifurcation and Chaos*, 10(10) :2429–2439.
- Lachaux, J.-P., Rodriguez, E., Martinerie, J., Varela, F. J., and others (1999). Measuring phase synchrony in brain signals. *Human brain mapping*, 8(4) :194–208.

- Lajnef, T., Chaibi, S., Ruby, P., Aguera, P.-E., Eichenlaub, J.-B., Samet, M., Kachouri, A., and Jerbi, K. (2015). Learning machines and sleeping brains : Automatic sleep stage classification using decision-tree multi-class support vector machines. *Journal of Neuroscience Methods*, 250 :94–105.
- Lakatos, P. (2005). An Oscillatory Hierarchy Controlling Neuronal Excitability and Stimulus Processing in the Auditory Cortex. *Journal of Neurophysiology*, 94(3) :1904–1911.
- Lalor, E., Kelly, S. P., Finucane, C., Burke, R., Reilly, R. B., and McDarby, G. (2004). Brain computer interface based on the steady-state VEP for immersive gaming control. *Biomed. Tech*, 49(1) :63–64.
- Lalor, E. C., Kelly, S. P., Finucane, C., Burke, R., Smith, R., Reilly, R. B., and McDarby, G. (2005). Steady-State VEP-Based Brain-Computer Interface Control in an Immersive 3D Gaming Environment. *EURASIP Journal on Advances in Signal Processing*, 2005(19) :3156–3164.
- Latinne, P., Debeir, O., and Decaestecker, C. (2001). Limiting the number of trees in random forests. In *International Workshop on Multiple Classifier Systems*, pages 178–187. Springer.
- Lebedev, M. A. and Nicolelis, M. A. (2006). Brain-machine interfaces : Past, present and future. *Trends in Neurosciences*, 29(9) :536–546.
- Lee, J. and Jeong, J. (2013). Correlation of risk-taking propensity with cross-frequency phase-amplitude coupling in the resting EEG. *Clinical Neurophysiology*.
- Leeb, R., Friedman, D., Müller-Putz, G. R., Scherer, R., Slater, M., and Pfurtscheller, G. (2007). Self-Paced (Asynchronous) BCI Control of a Wheelchair in Virtual Environments : A Case Study with a Tetraplegic. *Computational Intelligence and Neuroscience*, 2007 :1–8.
- Lega, B., Burke, J., Jacobs, J., and Kahana, M. J. (2016). Slow-Theta-to-Gamma Phase-Amplitude Coupling in Human Hippocampus Supports the Formation of New Episodic Memories. *Cerebral Cortex*, 26(1) :268–278.
- Leuthardt, E. C., Schalk, G., Wolpaw, J. R., Ojemann, J. G., and Moran, D. W. (2004). A brain-computer interface using electrocorticographic signals in humans. *Journal of Neural Engineering*, 1(2) :63–71.
- Li, Z., O'Doherty, J. E., Hanson, T. L., Lebedev, M. A., Henriquez, C. S., and Nicolelis, M. A. L. (2009). Unscented Kalman Filter for Brain-Machine Interfaces. *PLoS ONE*, 4(7) :e6243.
- Liao, L.-D., Chen, C.-Y., Wang, I.-J., Chen, S.-F., Li, S.-Y., Chen, B.-W., Chang, J.-Y., and Lin, C.-T. (2012). Gaming control using a wearable and wireless EEG-based brain-computer interface device with novel dry foam-based sensors. *Journal of NeuroEngineering and Rehabilitation*, 9 :5.
- Lin, J.-S., Chen, K.-C., and Yang, W.-C. (2010). EEG and eye-blinking signals through a Brain-Computer Interface based control for electric wheelchairs with wireless scheme. In *New Trends in Information Science and Service Science (NISS), 2010 4th International Conference On*, pages 731–734. IEEE.

- Ling, C. X., Huang, J., and Zhang, H. (2003). AUC a statistically consistent and more discriminating measure than accuracy. In *IJCAI*, volume 3, pages 519–524.
- Liu, J., Ranka, S., and Kahveci, T. (2008). Classification and feature selection algorithms for multi-class CGH data. *Bioinformatics*, 24(13) :i86–i95.
- Liu, Y., Jiang, X., Cao, T., Wan, F., Mak, P. U., Mak, P.-I., and Vai, M. I. (2012). Implementation of SSVEP based BCI with Emotiv EPOC. In *2012 IEEE International Conference on Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS) Proceedings*, pages 34–37. IEEE.
- Lorey, B., Bischoff, M., Pilgramm, S., Stark, R., Munzert, J., and Zentgraf, K. (2009). The embodied nature of motor imagery : The influence of posture and perspective. *Experimental Brain Research*, 194(2) :233–243.
- Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F., Arnaldi, B., and others (2007). A review of classification algorithms for EEG-based brain–computer interfaces. *Journal of neural engineering*, 4.
- Malouin, F., Jackson, P. L., and Richards, C. L. (2013). Towards the integration of mental practice in rehabilitation programs. A critical review. *Frontiers in Human Neuroscience*, 7.
- Markand, O. N. (1990). Alpha rhythms. *Journal of Clinical Neurophysiology*, 7(2) :163–190.
- McFarland, D. J., Krusienski, D. J., Sarnacki, W. A., and Wolpaw, J. R. (2008). Emulation of computer mouse control with a noninvasive brain-computer interface. *Journal of neural engineering*, 5(2) :101.
- McFarland, D. J., Sarnacki, W. A., and Wolpaw, J. R. (2010). Electroencephalographic (EEG) control of three-dimensional movement. *Journal of Neural Engineering*, 7(3) :036007.
- McFarland, D. J. and Wolpaw, J. R. (2008). Brain-computer interface operation of robotic and prosthetic devices. *Computer*, 41(10) :52–56.
- Mehring, C., Nawrot, M. P., de Oliveira, S. C., Vaadia, E., Schulze-Bonhage, A., Aertsen, A., and Ball, T. (2004). Comparing information about arm movement direction in single channels of local and epicortical field potentials from monkey and human motor cortex. *Journal of Physiology-Paris*, 98(4-6) :498–506.
- Mellinger, J., Schalk, G., Braun, C., Preissl, H., Rosenstiel, W., Birbaumer, N., and Kübler, A. (2007). An MEG-based brain-computer interface (BCI). *Neuroimage*, 36(3) :581.
- Mihara, M., Hattori, N., Hatakenaka, M., Yagura, H., Kawano, T., Hino, T., and Miyai, I. (2013). Near-infrared spectroscopy-mediated neurofeedback enhances efficacy of motor imagery-based training in poststroke victims a pilot study. *Stroke*, 44(4) :1091–1098.
- Milekovic, T., Fischer, J., Pistohl, T., Ruescher, J., Schulze-Bonhage, A., Aertsen, A., Rickert, J., Ball, T., and Mehring, C. (2012). An online brain–machine interface using decoding of movement direction from the human electrocorticogram. *Journal of Neural Engineering*, 9(4) :046003.

- Miranda, E. R. (2006). Brain-computer music interface for composition and performance. *International Journal on Disability and Human Development*, 5(2) :119–126.
- Miranda, E. R., Sharman, K., Kilborn, K., and Duncan, A. (2003). On Harnessing the Electroencephalogram for the Musical Braincap. *Computer Music Journal*, 27(2) :80–102.
- Mugler, E. M., Ruf, C. A., Halder, S., Bensch, M., and Kubler, A. (2010). Design and implementation of a P300-based brain-computer interface for controlling an internet browser. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(6) :599–609.
- Muller-Putz, G. and Pfurtscheller, G. (2008). Control of an Electrical Prosthesis With an SSVEP-Based BCI. *IEEE Transactions on Biomedical Engineering*, 55(1) :361–364.
- Muller Putz, G., Scherer, R., Neuper, C., and Pfurtscheller, G. (2006). Steady state somatosensory evoked potentials suitable brain signals for brain computer interfaces? *IEEE transactions on neural systems and rehabilitation engineering*, 14(1) :30–37.
- Müller-Putz, G. R., Scherer, R., Pfurtscheller, G., and Rupp, R. (2005). EEG-based neuroprosthesis control : A step towards clinical practice. *Neuroscience letters*, 382(1) :169–174.
- Münßinger, J. I., Halder, S., Kleih, S. C., Furdea, A., Raco, V., Hösle, A., and Kübler, A. (2010). Brain Painting : First Evaluation of a New Brain–Computer Interface Application with ALS-Patients and Healthy Volunteers. *Frontiers in Neuroscience*, 4.
- Nagaoka, T., Sakatani, K., Awano, T., Yokose, N., Hoshino, T., Murata, Y., Katayama, Y., Ishikawa, A., and Eda, H. (2010). Development of a new rehabilitation system based on a brain-computer interface using near-infrared spectroscopy. In *Oxygen Transport to Tissue XXXI*, pages 497–503. Springer.
- Nakhnikian, A., Ito, S., Dwiel, L., Grasse, L., Rebec, G., Lauridsen, L., and Beggs, J. (2016). A novel cross-frequency coupling detection method using the generalized Morse wavelets. *Journal of Neuroscience Methods*, 269 :61–73.
- Naseer, N. and Hong, K.-S. (2015). fNIRS based brain-computer interfaces : A review. *Frontiers in Human Neuroscience*, 9.
- Neuper, C., Müller-Putz, G. R., Scherer, R., and Pfurtscheller, G. (2006). Motor imagery and EEG-based control of spelling devices and neuroprostheses. *Progress in brain research*, 159 :393–409.
- Neuper, C., Scherer, R., Reiner, M., and Pfurtscheller, G. (2005). Imagery of motor actions : Differential effects of kinesthetic and visual–motor mode of imagery in single-trial EEG. *Cognitive Brain Research*, 25(3) :668–677.
- Niedermeyer, E. (2004). The Normal EEG of the Waking Adult. *Electroencephalography : Basic principles, clinical applications, and related fields*, page 167.
- Niedermeyer, E. and da Silva, F. L. (2005). *Electroencephalography : Basic Principles, Clinical Applications, and Related Fields*. Lippincott Williams & Wilkins.

- Nijholt, A. (2008). BCI for games : A 'state of the art'survey. In *International Conference on Entertainment Computing*, pages 225–228. Springer.
- Ojala, M. and Garriga, G. C. (2010). Permutation tests for studying classifier performance. *The Journal of Machine Learning Research*, 11 :1833–1863.
- Ossandon, T., Jerbi, K., Vidal, J. R., Bayle, D. J., Henaff, M.-A., Jung, J., Minotti, L., Bertrand, O., Kahane, P., and Lachaux, J.-P. (2011). Transient Suppression of Broadband Gamma Power in the Default-Mode Network Is Correlated with Task Complexity and Subject Performance. *Journal of Neuroscience*, 31(41) :14521–14530.
- Oude Bos, D. and Reuderink, B. (2008). Brainbasher a bci game.
- Ozkurt, T. E. (2012). Statistically Reliable and Fast Direct Estimation of Phase-Amplitude Cross-Frequency Coupling. *Biomedical Engineering, IEEE Transactions on*, 59(7) :1943–1950.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn : Machine Learning in Python. *Journal of Machine Learning Research*, 12 :2825–2830.
- Penny, W., Duzel, E., Miller, K., and Ojemann, J. (2008). Testing for nested oscillation. *Journal of Neuroscience Methods*, 174(1) :50–61.
- Perelmouter, J., Kotchoubey, B., Kubler, A., Taub, E., and Birbaumer, N. (1999). Language support program for thought-translation-devices. *Automedica*, 18(1) :67–84.
- Pfurtscheller, G. and Berghold, A. (1989). Patterns of cortical activation during planning of voluntary movement. *Electroencephalography and clinical neurophysiology*, 72(3) :250–258.
- Pfurtscheller, G., Guger, C., Müller, G., Krausz, G., and Neuper, C. (2000). Brain oscillations control hand orthosis in a tetraplegic. *Neuroscience letters*, 292(3) :211–214.
- Pfurtscheller, G. and Lopes da Silva, F. H. (1999). Event-related EEG/MEG synchronization and desynchronization : Basic principles. *Clinical neurophysiology*, 110(11) :1842–1857.
- Pfurtscheller, G., Muller-Putz, G., Scherer, R., and Neuper, C. (2008). Rehabilitation with Brain-Computer Interface Systems. *Computer*, 41(10) :58–65.
- Philips, J., del R Millan, J., Vanacker, G., Lew, E., Galán, F., Ferrez, P. W., Van Brussel, H., and Nuttin, M. (2007). Adaptive shared control of a brain-actuated simulated wheelchair. In *Rehabilitation Robotics, 2007. ICORR 2007. IEEE 10th International Conference On*, pages 408–414.
- Pires, G., Castelo-Branco, M., and Nunes, U. (2008). Visual P300-based BCI to steer a wheelchair : A Bayesian approach. In *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 658–661. IEEE.
- Pistohl, T., Ball, T., Schulze-Bonhage, A., Aertsen, A., and Mehring, C. (2008). Prediction of arm movement trajectories from ECoG-recordings in humans. *Journal of Neuroscience Methods*, 167(1) :105–114.

- Pistohl, T., Schulze-Bonhage, A., Aertsen, A., Mehring, C., and Ball, T. (2012). Decoding natural grasp types from human ECoG. *NeuroImage*, 59(1) :248–260.
- Rebsamen, B. (2009). *A Brain Controlled Wheelchair to Navigate in Familiar Environments*. PhD thesis, national university of singapore.
- Rickert, J. (2005). Encoding of Movement Direction in Different Frequency Ranges of Motor Cortical Local Field Potentials. *Journal of Neuroscience*, 25(39) :8815–8824.
- Rockstroh, B. (1989). *Slow Cortical Potentials and Behaviour*. Urban & Schwarzenberg.
- Rockstroh, B., Elbert, T., Birbaumer, N., Wolf, P., Dückting-Röth, A., Reker, M., Daum, I., Lutzenberger, W., and Dichgans, J. (1993). Cortical self-regulation in patients with epilepsies. *Epilepsy research*, 14(1) :63–72.
- Ruby, P. and Decety, J. (2001). Effect of subjective perspective taking during simulation of action : A PET investigation of agency. *Nature neuroscience*, 4(5) :546–550.
- Sajda, P., Gerson, A., Muller, K.-R., Blankertz, B., and Parra, L. (2003). A data analysis competition to evaluate machine learning algorithms for use in brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2) :184–185.
- Salmelin, R. and Hari, R. (1994). Spatiotemporal characteristics of sensorimotor neuromagnetic rhythms related to thumb movement. *Neuroscience*, 60(2) :537–550.
- Sato, S., Balish, M., and Muratore, R. (1991). Principles of Magnetoencephalography :. *Journal of Clinical Neurophysiology*, 8(2) :144–156.
- Schacter, D. L. (1977). EEG theta waves and psychological phenomena : A review and analysis. *Biological psychology*, 5(1) :47–82.
- Schalk, G., Kubanek, J., Miller, K., Anderson, N., Leuthardt, E., Ojemann, J., Limbrick, D., Moran, D., Gerhardt, L., and Wolpaw, J. (2007). Decoding two-dimensional movement trajectories using electrocorticographic signals in humans. *Journal of neural engineering*, 4(3) :264.
- Schalk, G. and Leuthardt, E. C. (2011). Brain-Computer Interfaces Using Electrocorticographic Signals. *IEEE Reviews in Biomedical Engineering*, 4 :140–154.
- Schalk, G., McFarland, D., Hinterberger, T., Birbaumer, N., and Wolpaw, J. (2004). BCI2000 A General-Purpose Brain-Computer Interface (BCI) System. *IEEE Transactions on Biomedical Engineering*, 51(6) :1034–1043.
- Scherer, R., Zanos, S. P., Miller, K. J., Rao, R. P., and Ojemann, J. G. (2009). Classification of contralateral and ipsilateral finger movements for electrocorticographic brain-computer interfaces. *Neurosurgical Focus*, 27(1) :E12.
- Schreuder, M., Blankertz, B., and Tangermann, M. (2010). A new auditory multi-class brain-computer interface paradigm : Spatial hearing as an informative cue. *PloS one*, 5(4) :e9813.
- Schuster, C., Hilfiker, R., Amft, O., Scheidhauer, A., Andrews, B., Butler, J., Kischka, U., and Ettlin, T. (2011). Best practice for motor imagery : A systematic literature review on motor imagery training elements in five different disciplines. *BMC Medicine*, 9(1).

- Seiler, B. D., Monsma, E. V., and Newman-Norlund, R. D. (2015). Biological Evidence of Imagery Abilities : Intraindividual Differences. *Journal of Sport and Exercise Psychology*, 37(4) :421–435.
- Sellers, E. W. and Donchin, E. (2006). A P300-based brain-computer interface : Initial tests by ALS patients. *Clinical neurophysiology*, 117(3) :538–548.
- Sellers, E. W., Kubler, A., and Donchin, E. (2006). Brain-computer interface research at the University of South Florida Cognitive Psychophysiology Laboratory : The P300 speller. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2) :221–224.
- Sharma, N., Pomeroy, V. M., and Baron, J.-C. (2006). Motor Imagery : A Backdoor to the Motor System After Stroke ? *Stroke*, 37(7) :1941–1952.
- Shibasaki, H. and Hallett, M. (2006). What is the Bereitschaftspotential ? *Clinical Neurophysiology*, 117(11) :2341–2356.
- Shih, J. J. and Krusienski, D. J. (2012). Signals from intraventricular depth electrodes can control a brain–computer interface. *Journal of Neuroscience Methods*, 203(2) :311–314.
- Shih, J. J., Krusienski, D. J., and Wolpaw, J. R. (2012). Brain-Computer Interfaces in Medicine. *Mayo Clinic Proceedings*, 87(3) :268–279.
- Shirvalkar, P. R., Rapp, P. R., and Shapiro, M. L. (2010). Bidirectional changes to hippocampal theta–gamma comodulation predict memory for recent spatial episodes. *Proceedings of the National Academy of Sciences*, 107(15) :7054–7059.
- Siegel, M., Warden, M. R., and Miller, E. K. (2009). Phase-dependent neuronal coding of objects in short-term memory. *Proceedings of the National Academy of Sciences*, 106(50) :21341–21346.
- Silber, M. H., Ancoli-Israel, S., Bonnet, M. H., Chokroverty, S., Grigg-Damberger, M. M., Hirshkowitz, M., Kapen, S., Keenan, S. A., Kryger, M. H., Penzel, T., and others (2007). The visual scoring of sleep in adults. *J Clin Sleep Med*, 3(2) :121–131.
- Sitaram, R., Caria, A., Veit, R., Gaber, T., Rota, G., Kuebler, A., and Birbaumer, N. (2007a). fMRI Brain-Computer Interface : A Tool for Neuroscientific Research and Treatment. *Computational Intelligence and Neuroscience*, 2007 :1–10.
- Sitaram, R., Zhang, H., Guan, C., Thulasidas, M., Hoshi, Y., Ishikawa, A., Shimizu, K., and Birbaumer, N. (2007b). Temporal classification of multichannel near-infrared spectroscopy signals of motor imagery for developing a brain–computer interface. *NeuroImage*, 34(4) :1416–1427.
- Solodkin, A. (2004). Fine Modulation in Network Activation during Motor Execution and Motor Imagery. *Cerebral Cortex*, 14(11) :1246–1255.
- Soto, J. L. and Jerbi, K. (2012). Investigation of cross-frequency phase-amplitude coupling in visuomotor networks using magnetoencephalography. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 1550–1553. IEEE.

- Sunny, T., Aparna, T., Neethu, P., Venkateswaran, J., Vishnupriya, V., and Vyas, P. (2016). Robotic Arm with Brain – Computer Interfacing. *Procedia Technology*, 24 :1089–1096.
- Sutter, E. E. (1992). Special Issue on Computers for Handicapped PeopleThe brain response interface : Communication through visually-induced electrical brain responses. *Journal of Microcomputer Applications*, 15(1) :31–45.
- Talairach, J. and Tournoux, P. (1993). *Referentially Oriented Cerebral MRI Anatomy : An Atlas of Stereotaxic Anatomical Correlations for Gray and White Matter*. Thieme.
- Tallon-Baudry, C. and Bertrand, O. (1999). Oscillatory gamma activity in humans and its role in object representation. *Trends in cognitive sciences*, 3(4) :151–162.
- Tallon-Baudry, C., Bertrand, O., Delpuech, C., and Pernier, J. (1997). Oscillatory γ -band (30–70 Hz) activity induced by a visual search task in humans. *The Journal of neuroscience*, 17(2) :722–734.
- Tanaka, K., Matsunaga, K., and Wang, H. O. (2005). Electroencephalogram-based control of an electric wheelchair. *IEEE transactions on robotics*, 21(4) :762–766.
- Tangermann, M., Müller, K.-R., Aertsen, A., Birbaumer, N., Braun, C., Brunner, C., Leeb, R., Mehring, C., Miller, K. J., Müller-Putz, G. R., Nolte, G., Pfurtscheller, G., Preissl, H., Schalk, G., Schlögl, A., Vidaurre, C., Waldert, S., and Blankertz, B. (2012). Review of the BCI Competition IV. *Frontiers in Neuroscience*, 6.
- Taylor, D. M. (2002). Direct Cortical Control of 3D Neuroprosthetic Devices. *Science*, 296(5574) :1829–1832.
- Tort, A. B. L., Komorowski, R., Eichenbaum, H., and Kopell, N. (2010). Measuring Phase-Amplitude Coupling Between Neuronal Oscillations of Different Frequencies. *Journal of Neurophysiology*, 104(2) :1195–1210.
- Trejo, L. J., Rosipal, R., and Matthews, B. (2006). Brain-computer interfaces for 1-D and 2-D cursor control : Designs using volitional control of the EEG spectrum or steady-state visual evoked potentials. *IEEE transactions on neural systems and rehabilitation engineering*, 14(2) :225–229.
- Vadera, S., Marathe, A. R., Gonzalez-Martinez, J., and Taylor, D. M. (2013). Stereoelectroencephalography for continuous two-dimensional cursor control in a brain-machine interface. *Neurosurgical focus*, 34(6) :E3.
- van Elswijk, G., Maij, F., Schoffelen, J.-M., Overeem, S., Stegeman, D. F., and Fries, P. (2010). Corticospinal Beta-Band Synchronization Entails Rhythmic Gain Modulation. *The Journal of Neuroscience*, 30(12) :4481–4488.
- Vanacker, G., Millán, J. d. R., Lew, E., Ferrez, P. W., Moles, F. G., Philips, J., Van Brussel, H., and Nuttin, M. (2007). Context-Based Filtering for Assisted Brain-Actuated Wheelchair Driving. *Computational Intelligence and Neuroscience*, 2007 :1–12.
- Vaughan, T., Mcfarland, D., Schalk, G., Sarnacki, W., Krusinski, D., Sellers, E., and Wolpaw, J. (2006). The Wadsworth BCI Research and Development Program : At Home With BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2) :229–233.

- Velliste, M., Perel, S., Spalding, M. C., Whitford, A. S., and Schwartz, A. B. (2008). Cortical control of a prosthetic arm for self-feeding. *Nature*, 453(7198) :1098–1101.
- Vidal, J. J. (1973). Toward Direct Brain-Computer Communication. *Annual Review of Biophysics and Bioengineering*, 2(1) :157–180.
- Vidal, J. J. (1977). Real-time detection of brain events in EEG. *Proceedings of the IEEE*, 65(5) :633–641.
- Vidaurre, C., Krämer, N., Blankertz, B., and Schlögl, A. (2009). Time domain parameters as a feature for EEG-based brain-computer interfaces. *Neural Networks*, 22(9) :1313–1319.
- Vladimir, V. N. and Vapnik, V. (1995). The nature of statistical learning theory.
- Voytek, B. (2010). Shifts in gamma phase-amplitude coupling frequency from theta to alpha over posterior cortex during visual tasks. *Frontiers in Human Neuroscience*, 4.
- Voytek, B., D'Esposito, M., Crone, N., and Knight, R. T. (2013). A method for event-related phase/amplitude coupling. *NeuroImage*, 64 :416–424.
- Waldert, S., Braun, C., Preissl, H., Birbaumer, N., Aertsen, A., and Mehring, C. (2007). Decoding performance for hand movements : EEG vs. MEG. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 5346–5348. IEEE.
- Waldert, S., Pistohl, T., Braun, C., Ball, T., Aertsen, A., and Mehring, C. (2009). A review on directional information in neural signals for brain-machine interfaces. *Journal of Physiology-Paris*, 103(3-5) :244–254.
- Waldert, S., Preissl, H., Demandt, E., Braun, C., Birbaumer, N., Aertsen, A., and Mehring, C. (2008). Hand Movement Direction Decoded from MEG and EEG. *Journal of Neuroscience*, 28(4) :1000–1008.
- Watrous, A. J., Deuker, L., Fell, J., and Axmacher, N. (2015). Phase-amplitude coupling supports phase coding in human ECoG. *eLife*, 4 :e07886.
- Weinberger, K. Q., Blitzer, J., and Saul, L. K. (2005). Distance metric learning for large margin nearest neighbor classification. In *Advances in Neural Information Processing Systems*, pages 1473–1480.
- Weiskopf, N., Mathiak, K., Bock, S., Scharnowski, F., Veit, R., Grodd, W., Goebel, R., and Birbaumer, N. (2004). Principles of a Brain-Computer Interface (BCI) Based on Real-Time Functional Magnetic Resonance Imaging (fMRI). *IEEE Transactions on Biomedical Engineering*, 51(6) :966–970.
- Wieland, M. and Pittore, M. (2014). Performance Evaluation of Machine Learning Algorithms for Urban Pattern Recognition from Multi-spectral Satellite Images. *Remote Sensing*, 6(4) :2912–2939.
- Wilson, J. A., Felton, E. A., Garell, P. C., Schalk, G., and Williams, J. C. (2006). ECoG factors underlying multimodal control of a brain-computer interface. *IEEE transactions on neural systems and rehabilitation engineering*, 14(2) :246–250.

- Wolpaw, J., McFarland, D., Vaughan, T., and Schalk, G. (2003). The wadsworth center brain-computer interface (bci) research and development program. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2) :204–207.
- Wolpaw, J. R. (2007). Brain-computer interfaces as new brain output pathways : BCIs as new brain outputs. *The Journal of Physiology*, 579(3) :613–619.
- Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., Vaughan, T. M., and others (2002). Brain computer interfaces for communication and control. *Clinical neurophysiology*, 113(6) :767–791.
- Wolpaw, J. R. and McFarland, D. J. (2004a). Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences of the United States of America*, 101(51) :17849–17854.
- Wolpaw, J. R. and McFarland, D. J. (2004b). Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences of the United States of America*, 101(51) :17849–17854.
- Wolpaw, J. R., McFarland, D. J., Neat, G. W., and Forneris, C. A. (1991). An EEG-based brain-computer interface for cursor control. *Electroencephalography and clinical neurophysiology*, 78(3) :252–259.
- Worrell, G., Jerbi, K., Kobayashi, K., Lina, J., Zelmann, R., and Le Van Quyen, M. (2012). Recording and analysis techniques for high-frequency oscillations. *Progress in neurobiology*, 98(3) :265–278.
- Wu, W., Black, M., Mumford, D., Gao, Y., Bienenstock, E., and Donoghue, J. (2004). Modeling and Decoding Motor Cortical Activity Using a Switching Kalman Filter. *IEEE Transactions on Biomedical Engineering*, 51(6) :933–942.
- Wu, W., Black, M. J., Gao, Y., Bienenstock, E., Serruya, M., and Donoghue, J. P. (2002). Inferring hand motion from multi-cell recordings in motor cortex using a Kalman filter. In *SAB'02-Workshop on Motor Control in Humans and Robots : On the Interplay of Real Brains and Artificial Devices*, pages 66–73.
- Wu, W., Black, M. J., Gao, Y., Bienenstock, E., Serruya, M., Shaikhouni, A., and Donoghue, J. P. (2003). Neural decoding of cursor motion using a Kalman filter. *Advances in neural information processing systems*, pages 133–140.
- Wu, W., Gao, Y., Bienenstock, E., Donoghue, J. P., and Black, M. J. (2006). Bayesian population decoding of motor cortical activity using a Kalman filter. *Neural computation*, 18(1) :80–118.
- Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Yu, P. S., Zhou, Z.-H., Steinbach, M., Hand, D. J., and Steinberg, D. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14(1) :1–37.
- Yanagisawa, T., Hirata, M., Saitoh, Y., Kato, A., Shibuya, D., Kamitani, Y., and Yoshimine, T. (2009). Neural decoding using gyral and intrasulcal electrocorticograms. *NeuroImage*, 45(4) :1099–1106.

- Yanagisawa, T., Hirata, M., Saitoh, Y., Kishima, H., Matsushita, K., Goto, T., Fukuma, R., Yokoi, H., Kamitani, Y., and Yoshimine, T. (2012a). Electrocorticographic control of a prosthetic arm in paralyzed patients. *Annals of Neurology*, 71(3) :353–361.
- Yanagisawa, T., Yamashita, O., Hirata, M., Kishima, H., Saitoh, Y., Goto, T., Yoshimine, T., and Kamitani, Y. (2012b). Regulation of Motor Representation by Phase-Amplitude Coupling in the Sensorimotor Cortex. *The Journal of Neuroscience*, 32(44) :15467–15475.
- Yu, L. and Liu, H. (2004). Redundancy based feature selection for microarray data. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 737–742. ACM.
- Zickler, C., Halder, S., Kleih, S. C., Herbert, C., and Kübler, A. (2013). Brain painting : Usability testing according to the user-centered design in end users with severe motor paralysis. *Artificial intelligence in medicine*, 59(2) :99–110.
- Zimmermann, R., Marchal-Crespo, L., Edelmann, J., Lambercy, O., Fluet, M.-C., Riederer, R., Wolf, M., and Gassert, R. (2013). Detection of motor execution using a hybrid fNIRS-biosignal BCI : A feasibility study. *Journal of neuroengineering and rehabilitation*, 10(1) :1.