

# THÈSE DE L'UNIVERSITÉ DE LYON

Délivrée par

UNIVERSITÉ CLAUDE BERNARD LYON 1

DIPLÔME 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 <sup>me</sup>	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 Isabelle et Didier, mes deux parents,  
qui ont tout donné pour que ceci me soit un jour possible.  
Merci*



# REMERCIEMENTS

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

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

Montréal, le 5 juillet 2016.

# TABLE DES MATIÈRES

TABLE DES MATIÈRES	vi
LISTE DES FIGURES	ix
NOTATIONS	1
<b>I Introduction générale</b>	<b>3</b>
<b>1 PRÉSENTATION DE LA THÉMATIQUE</b>	<b>7</b>
<b>1.1 ENREGISTREMENT DE L'ACTIVITÉ NEURONALE . . . . .</b>	<b>7</b>
<b>1.1.1 Enregistrement non-invasif . . . . .</b>	<b>7</b>
<b>1.1.2 Enregistrement invasif . . . . .</b>	<b>7</b>
<b>1.2 ÉTAT DE L'ART DES INTERFACES CERVEAU-MACHINE . . . . .</b>	<b>7</b>
<b>1.2.1 ICM non-invasives . . . . .</b>	<b>8</b>
<b>1.2.2 ICM invasives . . . . .</b>	<b>8</b>
<b>1.3 APPRENTISSAGE MACHINE : APPLICATIONS AUX NEUROSCIENCES</b>	<b>8</b>
<b>1.4 ENCODAGE ET DÉCODAGE MOTEUR : BASES PHYSIOLOGIQUES .</b>	<b>8</b>
<b>1.5 INTENTION ET EXÉCUTION . . . . .</b>	<b>9</b>
<b>1.6 DELAYED TASK : PROTOCOLE EXPÉRIMENTAL . . . . .</b>	<b>9</b>
<b>2 OBJECTIFS DE LA THÈSE</b>	<b>11</b>
<b>2.1 DÉCODAGE CÉRÉBRALE À PARTIR D'ACTIVITÉ INTRACRÂNIENNE</b>	<b>11</b>
<b>2.2 EXPLORATION ET AMÉLIORATION DES FEATURES . . . . .</b>	<b>11</b>
<b>2.3 COMPARATIF DES CLASSIFIEURS . . . . .</b>	<b>11</b>
<b>2.4 EXPLORATION DES RÉGIONS NON-MOTRICES . . . . .</b>	<b>12</b>
<b>3 MÉTHODOLOGIE</b>	<b>13</b>
<b>3.1 EXTRACTION DES FEATURES . . . . .</b>	<b>13</b>
<b>3.1.1 Pré-requis . . . . .</b>	<b>13</b>
<b>3.1.2 Puissance spectrale . . . . .</b>	<b>15</b>
<b>3.1.3 Phase . . . . .</b>	<b>17</b>
<b>3.1.4 Phase-amplitude coupling . . . . .</b>	<b>17</b>
<b>3.2 APPRENTISSAGE SUPERVISÉ . . . . .</b>	<b>22</b>
<b>3.2.1 Labellisation et apprentissage . . . . .</b>	<b>23</b>
<b>3.2.2 <i>Training, testing</i> et validation-croisée . . . . .</b>	<b>23</b>
<b>3.2.3 Classificateurs . . . . .</b>	<b>25</b>
<b>3.2.4 Évaluation de la performance de décodage . . . . .</b>	<b>27</b>
<b>3.2.5 Seuil de chance et évaluation statistique de la performance de décodage . . . . .</b>	<b>27</b>
<b>3.2.6 Du single au multi-features . . . . .</b>	<b>28</b>

3.3	CONFIGURATION POUR DÉBUTER . . . . .	31
4	DONNÉES EXPÉRIMENTALES	33
4.1	DONNÉES "CENTER-OUT" . . . . .	33
4.2	AUTRES DONNÉES . . . . .	33
5	OUVERTURE	35
<b>II</b>	<b>Étude 1 : niveau de chance et évaluation statistique des résultats de classification par apprentissage supervisé</b>	<b>37</b>
5.1	PRÉSENTATION DE L'ÉTUDE . . . . .	41
5.1.1	Contexte . . . . .	41
5.1.2	Problématique . . . . .	41
5.1.3	Résultats majeurs . . . . .	41
5.2	ARTICLE . . . . .	41
5.3	COMPLÉMENTS D'ÉTUDE . . . . .	53
<b>III</b>	<b>Étude 2 : encodage de l'intention et de l'exécution motrice</b>	<b>55</b>
5.4	RÉSUMÉ DE L'ÉTUDE . . . . .	59
5.5	ARTICLE . . . . .	59
CONCLUSION . . . . .	59	
<b>IV</b>	<b>Étude 3 : décodage des directions de mouvement pendant et avant l'exécution de mouvement de membres supérieurs</b>	<b>61</b>
5.6	RÉSUMÉ DE L'ÉTUDE . . . . .	65
5.7	ARTICLE . . . . .	65
CONCLUSION . . . . .	65	
<b>V</b>	<b>Étude 4 : optimisation des paramètres de la bande gamma</b>	<b>67</b>
5.8	RÉSUMÉ DE L'ÉTUDE . . . . .	71
5.9	ARTICLE . . . . .	71
CONCLUSION . . . . .	71	
<b>VI</b>	<b>Étude 5 : décodage des émotions</b>	<b>73</b>
5.10	RÉSUMÉ DE L'ÉTUDE . . . . .	77
5.11	ARTICLE . . . . .	77
CONCLUSION . . . . .	77	
CONCLUSION GÉNÉRALE		79
<b>A</b>	<b>ANNEXES</b>	<b>80</b>
A.1	COMPARATIF DE MÉTHODES PAC (TORT ET AL., 2010) . . . . .	81
A.2	PIPELINE STANDARD DE CLASSIFICATION . . . . .	82
A.3	COMPARATIF DE CLASSIFIERS (PEDREGOSA ET AL., 2011) . . . . .	84



# LISTE DES FIGURES

1.1	Techniques d'enregistrement de l'activité cérébrale . . . . .	7
1.2	Pipeline général d'un Interface Cerveau-Machine . . . . .	8
1.3	Contrôle d'un bras robotisé . . . . .	8
1.4	Comparatif . . . . .	9
2.1	Mécanismes du couplage phase-amplitude . . . . .	11
2.2	Localisation des aires sensorimotrices . . . . .	12
3.1	"One-tailed" et "two-tailed" test . . . . .	15
3.2	Exemple de représentation temps-fréquence de puissance normalisées z-score (Ossandon et al., 2011) . . . . .	17
3.3	Densité de probabilité d'une distribution d'amplitudes en fonction de tranches de phases . . . . .	18
3.4	(A) Exemple de cartes temps-fréquence phase locked sur le $\beta$ , (B) Exemple de comodulogramme . . . . .	21
3.5	Labellisation de données . . . . .	23
3.6	Exemple d'une cross validation 3-folds . . . . .	25
3.7	Principe du Linear Discriminant Analysis (Lotte et al., 2007)	25
3.8	Principe du Support Vector Machine (Lotte et al., 2007) . .	26
3.9	Principe du k-Nearest Neighbor (Weinberger et al., 2005) .	26
3.10	Entraînement puis test d'un classifieur linéaire . . . . .	26
3.11	Calcul de l'acuité de décodage . . . . .	27
3.12	Exemple d'une <i>Forward feature selection</i> appliquée sur six features . . . . .	30
3.13	Exemple d'une <i>Backward feature elimination</i> appliquée sur six features . . . . .	31
A.1	Comparatif de méthodes PAC (Tort et al., 2010) . . . . .	81
A.2	Pipeline standard de classification . . . . .	82
A.3	Comparatif de classifieurs (Pedregosa et al., 2011) . . . . .	84



# NOTATIONS

## Général

ICM Interface Cerveau-Machine  
BCI Brain Computer Interface

## Enregistrements

EEG Électroencéphalographie  
MEG Magnétoencéphalographie  
SUA Single Unit Activity  
MUA Multi Unit Activity  
SEEG Stéréoélectroencéphalographie  
ECoG Électrocorticographie

## Features

PAC Phase Amplitude Coupling

## Classifeurs

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



**Première partie**

**Introduction générale**



# B<sub>L</sub>ABLABLABKIBLABLOU INTRO ...

L'objectif de cette thèse a été de ...

Totalité des méthodes explorées durant ma thèse sont présentes dans une toolbox python appelée brainpipe, libre d'accès et de droit.



# PRÉSENTATION DE LA THÉMATIQUE

## 1.1 ENREGISTREMENT DE L'ACTIVITÉ NEURONALE

- Présentation de chacun des types de données
  - Résolution et RSB
  - Avantages // inconvénients
- (Waldert et al., 2009)

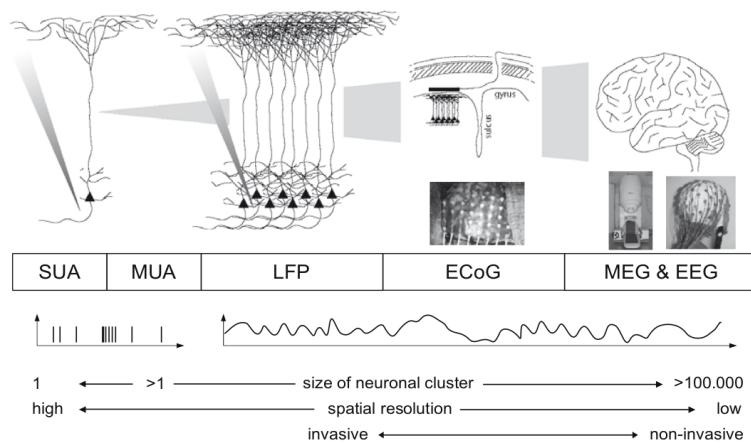


FIGURE 1.1 – Techniques d'enregistrement de l'activité cérébrale

### 1.1.1 Enregistrement non-invasif

Électroencéphalographie

Magnétoencéphalographie

### 1.1.2 Enregistrement invasif

Single Unit Activity

Multi Unit Activity

Stéréoélectroencéphalographie

Électrocorticographie

## 1.2 ÉTAT DE L'ART DES INTERFACES CERVEAU-MACHINE

(Bekaert et al., 2009)

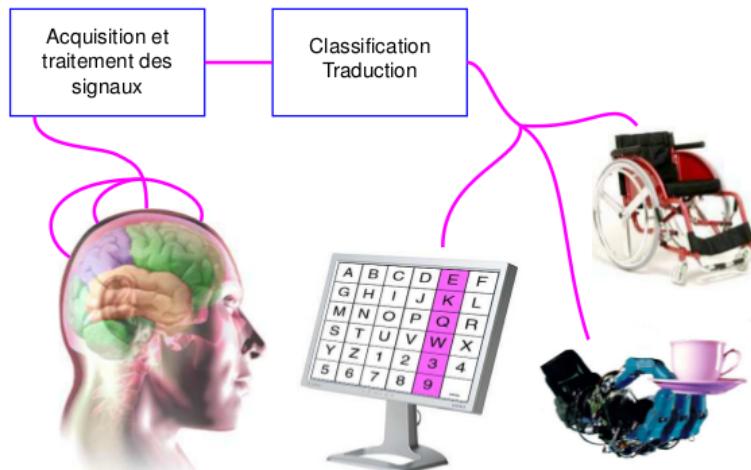


FIGURE 1.2 – Pipeline général d'un Interface Cerveau-Machine

### 1.2.1 ICM non-invasives

P300 speller

### 1.2.2 ICM invasives

(Hochberg et al., 2012)

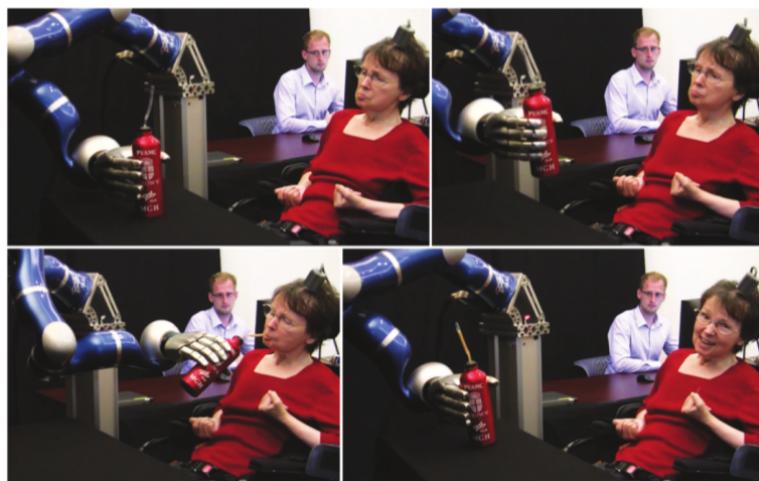


FIGURE 1.3 – Contrôle d'un bras robotisé

## 1.3 APPRENTISSAGE MACHINE : APPLICATIONS AUX NEUROSCIENCES

okok

## 1.4 ENCODAGE ET DÉCODAGE MOTEUR : BASES PHYSIOLOGIQUES

La figure 1.4 représente une trajectoire d'un processus markovien de saut, avec les notations associées. (Hanakawa et al., 2008)

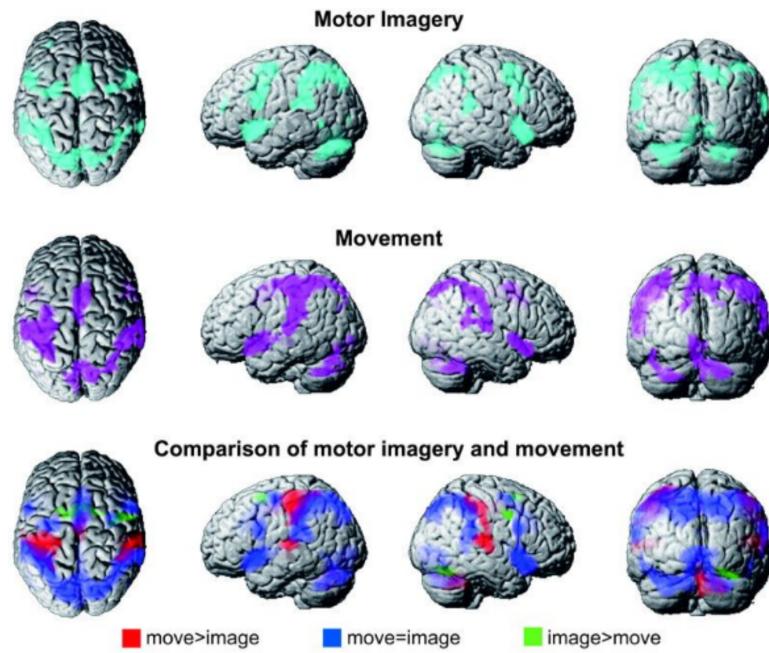


FIGURE 1.4 – Comparatif

## 1.5 INTENTION ET EXÉCUTION

okok

## 1.6 DELAYED TASK : PROTOCOLE EXPÉRIMENTAL

okok



# OBJECTIFS DE LA THÈSE

2

## 2.1 DÉCODAGE CÉRÉBRALE À PARTIR D'ACTIVITÉ INTRACRÂ-NIENNE

- Exemple d'un schéma d'implantation + IRM
- Bipolarisation : débruitage et augmentation de la spécificité (article Karim)
- Extraction de features (ici on pourrait mentionner que le deep learning pourrait marcher sur les données brutes)

## 2.2 EXPLORATION ET AMÉLIORATION DES FEATURES

### Rôle physiologique du phase-amplitude coupling

(Hyafil et al., 2015)

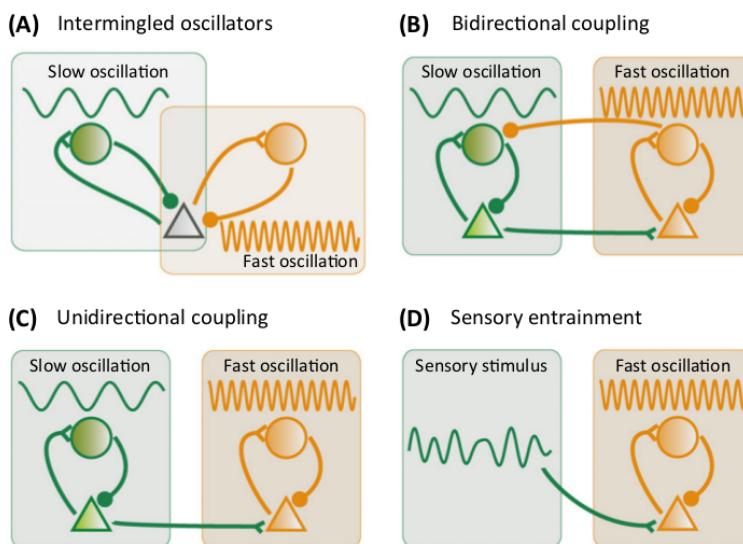


FIGURE 2.1 – Mécanismes du couplage phase-amplitude

## 2.3 COMPARATIF DES CLASSIFIERS

Expliquer que, chaque classifier possède une méthodologie propre permettant de répondre à des types de données différentes (en fonction des hypothèses de fonctionnement de chacun des classifiers)

## 2.4 EXPLORATION DES RÉGIONS NON-MOTRICES

(Van Langenhove et al., 2008)

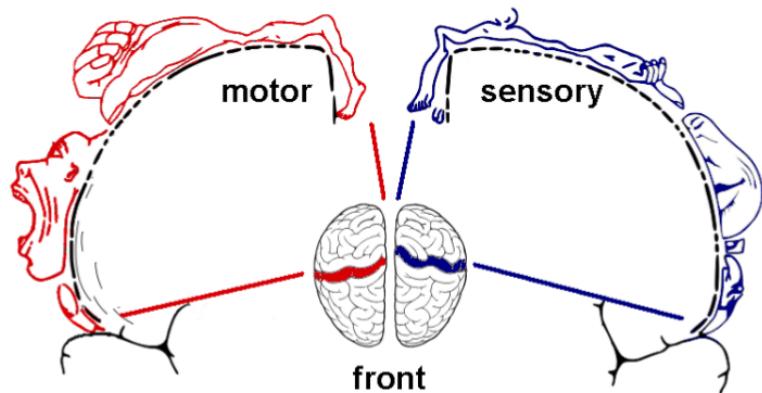


FIGURE 2.2 – Localisation des aires sensorimotrices

# MÉTHODOLOGIE

3

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

## 3.1 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.

### 3.1.1 Pré-requis

#### 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) :

$$FiltOrder = N_{cycle} \times f_s / f_{oi} \quad (3.1)$$

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.

### 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  $x_H(t)$ .

### 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 :

$$f(a, b) = \int_{-\infty}^{\infty} f(x) \bar{\psi}_{a,b} dx \quad (3.2)$$

Où  $\psi$  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 :

$$w(t, f_0) = A e^{-t^2/2\sigma_t^2} e^{2i\pi f_0 t} \quad (3.3)$$

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.

### 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  $p$  pré-définie, 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")
- $A \neq B$  si  $A$  est soit inférieur aux  $(N \times p)/2$  premiers échantillons soit supérieur aux  $(N - N \times p)/2$

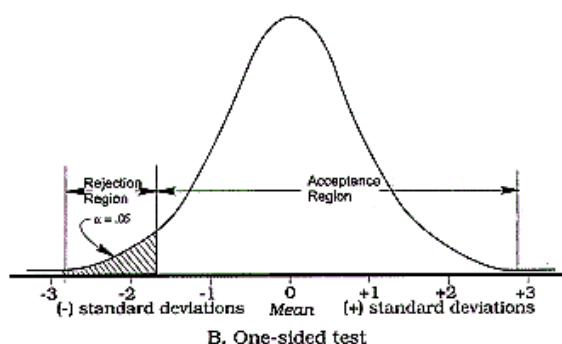
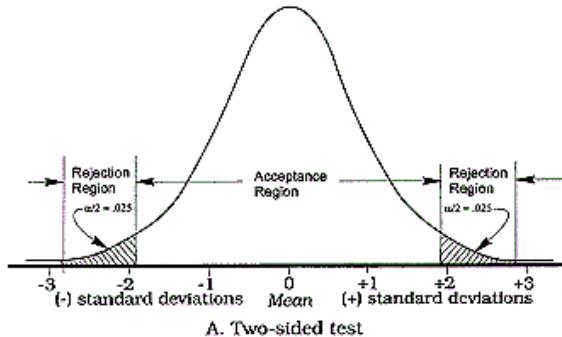


FIGURE 3.1 – "One-tailed" et "two-tailed" test

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  $1/p$ .

### 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  $N - 1$

$$\dim_{ESPACE} = \dim_{HYPERPLAN} + 1 \quad (3.4)$$

#### 3.1.2 Puissance spectrale

##### 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.

### 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,  $\beta_3$  était clairement la plus adaptée pour la visualisation. En revanche, dans le cadre de la classification, nous obtenions systématiquement de meilleurs résultats sans normalisation.

### 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)

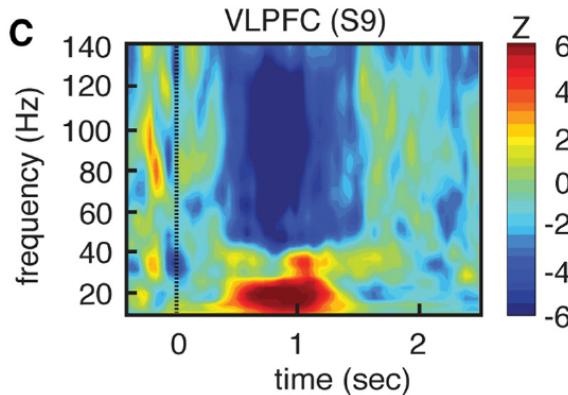


FIGURE 3.2 – Exemple de représentation temps-fréquence de puissance normalisées z-score (Ossandon et al., 2011)

### 3.1.3 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)

### 3.1.4 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.

### 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 :

$$MVL = \left| \sum_{j=1}^N a(j) \times e^{j\phi(j)} \right| \quad (3.5)$$

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  $20^\circ$ ). 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 3.3 (Tort et al., 2010) présente un exemple de DPA en fonction de tranches de phase.

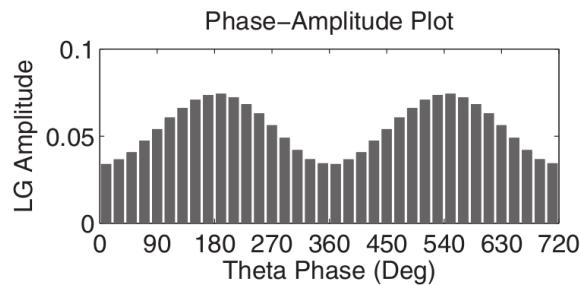


FIGURE 3.3 – Densité de probabilité d'une distribution d'amplitudes en fonction de tranches de phases

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 :

$$D_{KL}(P, Q) = \sum_{j=1}^N P(j) \times \log \frac{P(j)}{Q(j)} \quad (3.6)$$

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 :

$$hr = (f_{max} - f_{min}) / f_{max} \quad (3.7)$$

où  $f_{max}$  et  $f_{min}$  sont respectivement le maximum et le minimum de la 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 :

$$x_{lim} = N \times [erf^{-1}(1 - p)]^2 \quad (3.8)$$

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 supérieur 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.

### 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., 2012)

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.

### 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.1). Enfin, Canolty and Knight (2010) a fait une review qui comprend un descriptif très instructif.

### 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 moyennner 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

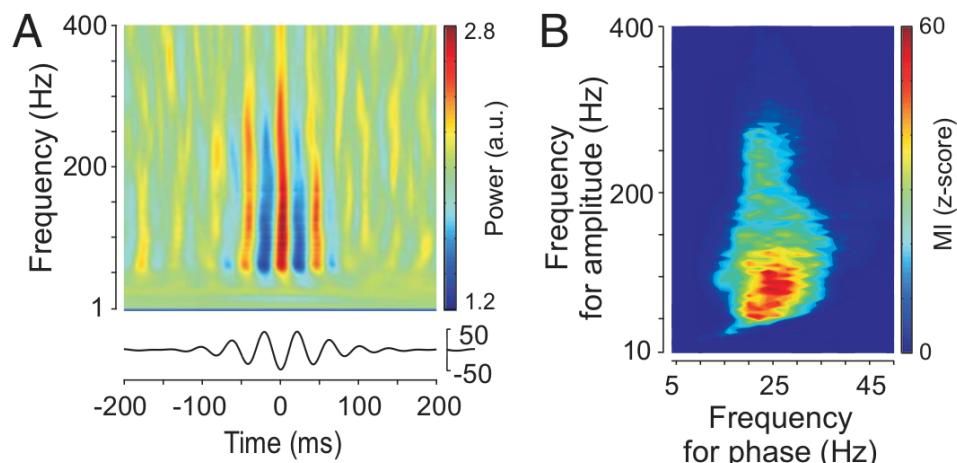


FIGURE 3.4 – (A) Exemple de cartes temps-fréquence phase locked sur le  $\beta$ , (B) Exemple de comodulogramme

La figure 3.4 (de Hemtinne 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 recalculer et d'autre part cette méthode est également limitée par l'instant où l'on choisit de recalculer. 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.

### Phase-amplitude coupling : résolution temporel ?

Comment peut-on savoir si un ensemble de musiciens jouent ensemble, en rythme ? L'approche traditionnelle consiste à dire que, en fonction de la prestation du groupe, on sera en mesure de dire si ils étaient en rythme ou

non. Donc on focalise notre attention sur chaque instant du morceau et on analyse chaque note, chaque décalage. Cela signifie aussi que toute notre attention a été mobilisée par l'analyse du rythme et finalement, on passe à côté de la musique. Notre attention au détail nous a écarté du morceau global. On pourrait dire que l'on a écrasé la dimension temporelle du morceau. Une autre approche consiste à assister à toutes les répétitions du fameux groupe. Ce faisant, on est capable de dire si d'une manière générale les musiciens ont tendance à jouer ensemble. Ainsi, le jour d'une représentation, toute notre attention peut rester uniquement sur le concert. On garde donc la dimension temporelle.

C'est par ce changement de positionnement face au problème de résolution temporelle que Voytek et al. (2013) introduit le Event Related Phase-Amplitude Coupling . L'approche traditionnelle du PAC nécessitant 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 calcule 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.

## 3.2 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. 3.2.1). 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, on 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 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

### 3.2.1 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  $x$ .

$$y_{direction} = \begin{pmatrix} up \\ down \\ down \\ \vdots \\ up \end{pmatrix}, y_{bool} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, x = \begin{pmatrix} x_{trial_1} \\ x_{trial_2} \\ x_{trial_3} \\ \vdots \\ x_{trial_N} \end{pmatrix} \quad (3.9)$$

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

FIGURE 3.5 – Labellisation de données

### 3.2.2 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.

### ***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 parître 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 représenterai qu'une partie des données. Une méthode plus exhaustive et plus rigoureuse consiste à utiliser une validation-croisée (ou cross-validation).

### **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 de CV mais de manière générale, toutes dérivent du même principe qui est la cross-validation k-Fold (Efron and Tibshirani, 1994, ?). 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 test 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 générale 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, ?).

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 puis-

qu'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.

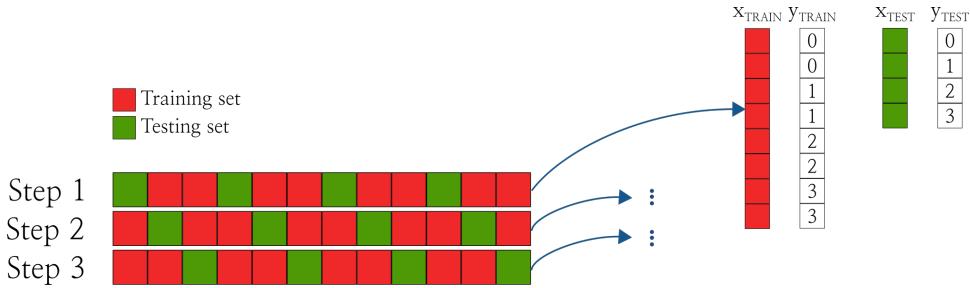


FIGURE 3.6 – Exemple d'une cross validation 3-folds

### 3.2.3 Classifieurs

#### 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*)

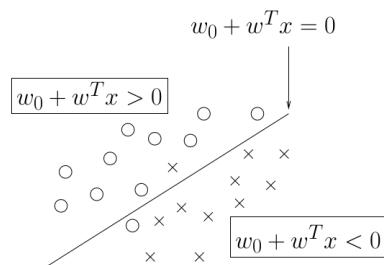


FIGURE 3.7 – Principe du Linear Discriminant Analysis (Lotte et al., 2007)

#### 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)* (?). Les problèmes multi-classes peuvent également être traités en *One-vs-All*

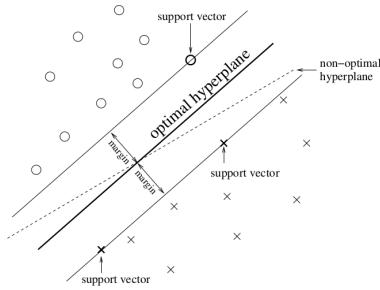


FIGURE 3.8 – Principe du Support Vector Machine (Lotte et al., 2007)

### 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)

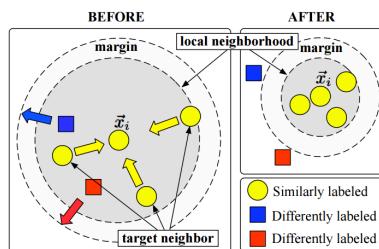


FIGURE 3.9 – Principe du k-Nearest Neighbor (Weinberger et al., 2005)

### 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.3 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)

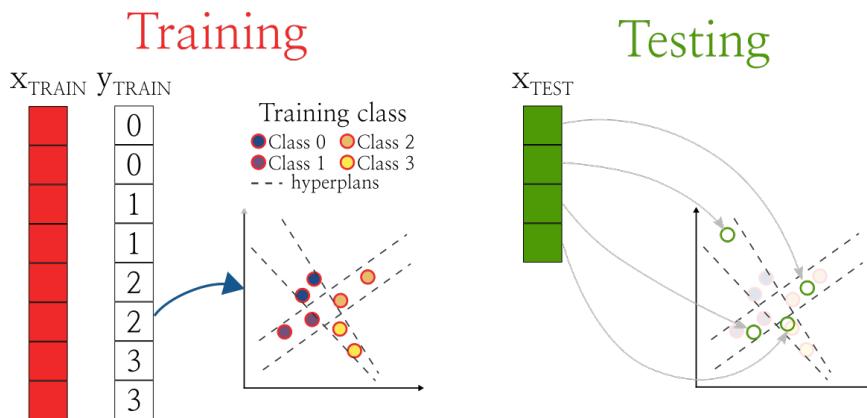


FIGURE 3.10 – Entraînement puis test d'un classifieur linéaire

### 3.2.4 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 prédictifs par le classifieur. En faisant la somme des labels correctement prédicts 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 3.11 – 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 .

### 3.2.5 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 distri-

bution binomiale cumulative, on peut utiliser la loi suivante pour en déduire la probabilité de prédire au moins  $z$  fois la classe  $c$  :

$$P(z) = \sum_{i=z}^n \binom{n}{i} \times \left(\frac{1}{c}\right)^i \times \left(\frac{c-1}{c}\right)^{n-1} \quad (3.10)$$

## 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'inter-dépendance. 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 est volontairement synthétique puisqu'elle a fait l'objet d'une publication scientifique (cf. II).

Un pipeline standard de classification est proposé en annexe (cf. A.2).

### 3.2.6 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 surapprentissage 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*.

### ***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)).

### 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 classifier-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 : 3.2.5). Enfin, on sélectionne les features dont la valeur  $p$  est inférieure 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 attribut. 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 continue 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  $i - 1$ .

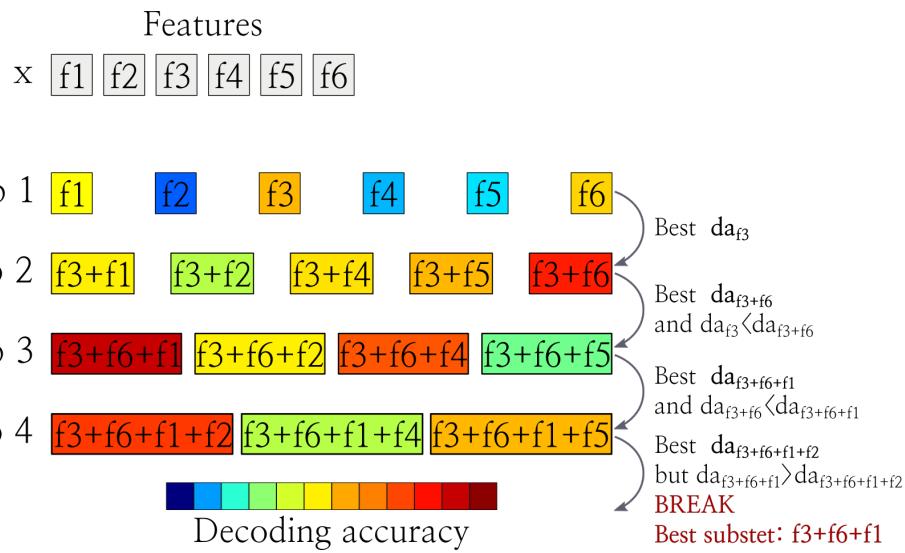


FIGURE 3.12 – Exemple d'une *Forward feature selection* appliquée sur six features

- 
- (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*.

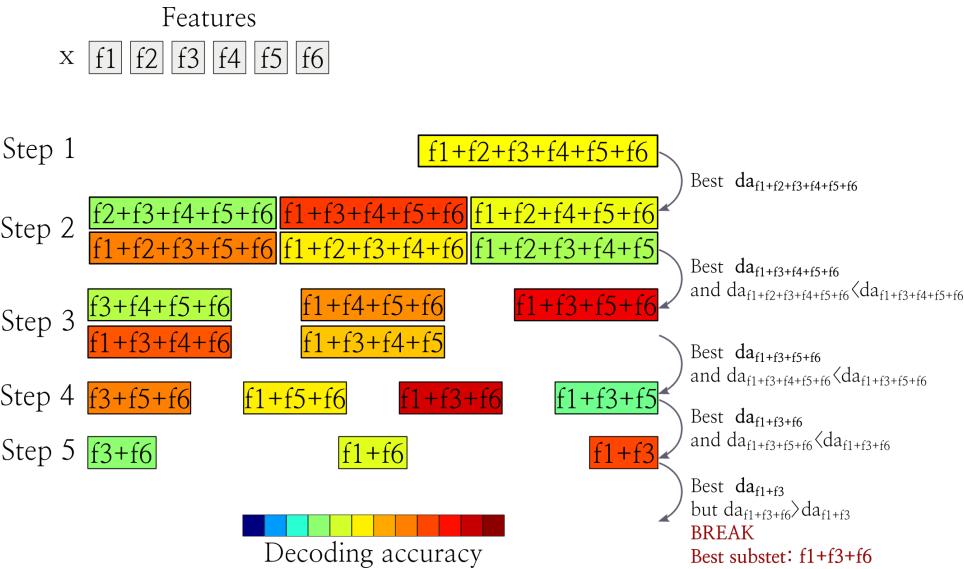


FIGURE 3.13 – Exemple d'une Backward feature elimination appliquée sur six features

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).

### 3.3 CONFIGURATION POUR DÉBUTER

Dans la jungle des méthodes, il peut parfois être difficile de s'y retrouver. Cette section a pour objectif de fournir une liste de méthodes conseillées pour débuter. Rien ne dit que ce sont les meilleures méthodes mais elles ont le mérite d'avoir fait leurs preuves, que ce soit dans cette thèse et surtout, dans la littérature. Gardons à l'esprit que les meilleures méthodes dépendent des données mais certaines, sont plus polyvalentes.

1. Pour le couplage phase-amplitude : Kullback-Leibler divergence avec swapping des essais de phase et d'amplitude (Tort et al., 2010)
2. Pour la classification :
  - (a) Cross-validation : 10-folds
  - (b) Classifieur : Support Vector Machine avec noyau linéaire (Vladimir and Vapnik, 1995, Lotte et al., 2007)

- (c) Évaluation statistique : permutations (Ojala and Garriga, 2010, Combrisson and Jerbi, 2015)
- (d) Multi-features : *Backward feature elimination* (Guyon and Elisseeff, 2003)

# DONNÉES EXPÉRIMENTALES

4

## 4.1 DONNÉES "CENTER-OUT"

Ajouter le tableau de Marcela

## 4.2 AUTRES DONNÉES



## OUVERTURE

5

Nos contributions portent sur : ...

Le *premier chapitre* expose la problématique de la thèse.  
Le *deuxième chapitre* présente en détail ...

etc.

Cette thèse a fait l'objet de divers travaux écrits : ...



## **Deuxième partie**

# **Étude 1 : niveau de chance et évaluation statistique des résultats de classification par apprentissage supervisé**



## SOMMAIRE

5.1	PRÉSENTATION DE L'ÉTUDE . . . . .	41
5.1.1	Contexte . . . . .	41
5.1.2	Problématique . . . . .	41
5.1.3	Résultats majeurs . . . . .	41
5.2	ARTICLE . . . . .	41
5.3	COMPLÉMENTS D'ÉTUDE . . . . .	53
5.4	RÉSUMÉ DE L'ÉTUDE . . . . .	59
5.5	ARTICLE . . . . .	59
	CONCLUSION . . . . .	59
5.6	RÉSUMÉ DE L'ÉTUDE . . . . .	65
5.7	ARTICLE . . . . .	65
	CONCLUSION . . . . .	65
5.8	RÉSUMÉ DE L'ÉTUDE . . . . .	71
5.9	ARTICLE . . . . .	71
	CONCLUSION . . . . .	71
5.10	RÉSUMÉ DE L'ÉTUDE . . . . .	77
5.11	ARTICLE . . . . .	77
	CONCLUSION . . . . .	77

Sensibilisation à l'importance du nombre d'essais par exemple



**5.1 PRÉSENTATION DE L'ÉTUDE****5.1.1 Contexte****5.1.2 Problématique****5.1.3 Résultats majeurs**

pourquoi cette étude ? Quelles questions ? - Seuil de chance théorique vs pratique ? - Impact sur des méthodes (cross-validation , classifieur ) - Validation sur des données réelles (Intra MEG) - dédié aux étudiants - Fournit une toolbox pour reproduire les résultats

**5.2 ARTICLE**



Contents lists available at ScienceDirect

# Journal of Neuroscience Methods

journal homepage: [www.elsevier.com/locate/jneumeth](http://www.elsevier.com/locate/jneumeth)



## 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 <sup>a,b</sup>, Karim Jerbi <sup>a,c,\*</sup>

<sup>a</sup> DYCOG Lab, Lyon Neuroscience Research Center, INSERM U1028, UMR 5292, University Lyon I, Lyon, France  
<sup>b</sup> Center of Research and Innovation in Sport, Mental Processes and Motor Performance, University of Lyon I, Lyon, France  
<sup>c</sup> 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 xxx

### 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  
EEG  
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. \(2012\)](#); [Korcyn](#)

[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](mailto: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; Sitaram 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  $\alpha$  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						
		$p < 0.05$		$p < 0.01$		$p < 10^{-3}$		$p < 10^{-4}$		$p < 0.05$		$p < 0.01$		$p < 10^{-3}$		
		$p < 0.05$	$p < 0.01$	$p < 10^{-3}$	$p < 10^{-4}$	$p < 0.05$	$p < 0.01$	$p < 10^{-3}$	$p < 10^{-4}$	$p < 0.05$	$p < 0.01$	$p < 10^{-3}$	$p < 10^{-4}$	$p < 0.05$	$p < 0.01$	
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

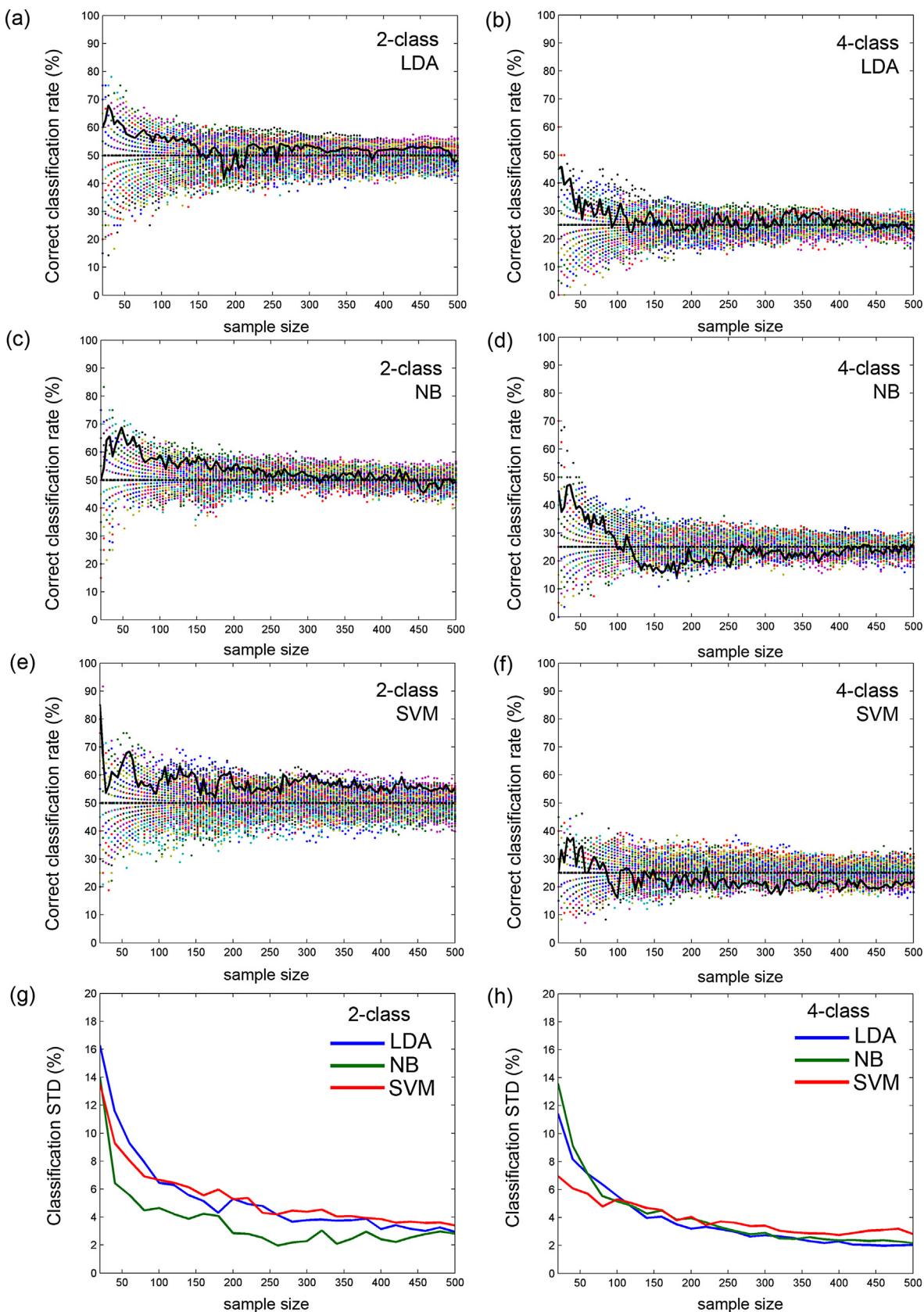
Fig. 1 shows the decoding accuracies obtained by conducting 10-fold cross validation on 100 randomly generated data sets. The decoding is depicted as a function of increasing sample size (from 24 to 500) and for the case of 2-class (left column) and 4-class (right column) classification. Although the 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.

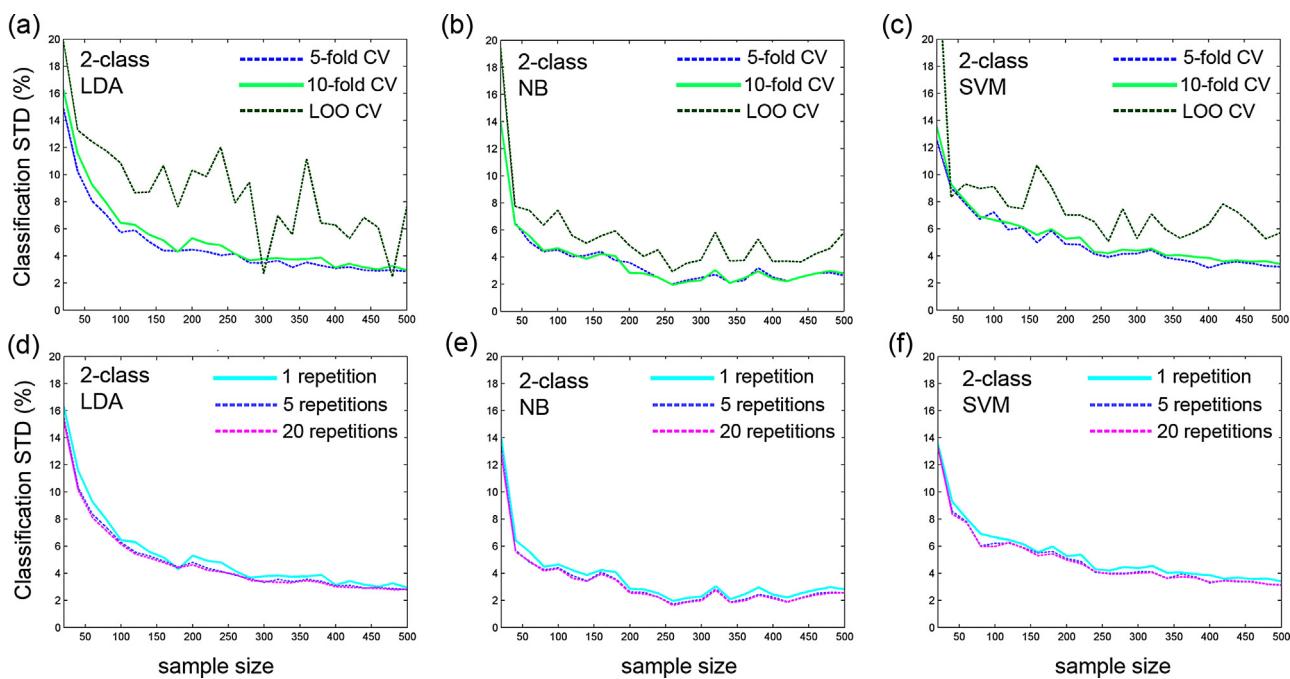
The small sample issue is persistent and qualitatively similar across all classifiers used. The first three rows of Fig. 1 show the results obtained with LDA, NB and SVM (with an RBF kernel). Panels (g) and (h) of Fig. 1 show that cross-validation results in all three classifiers have comparable deviation 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 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

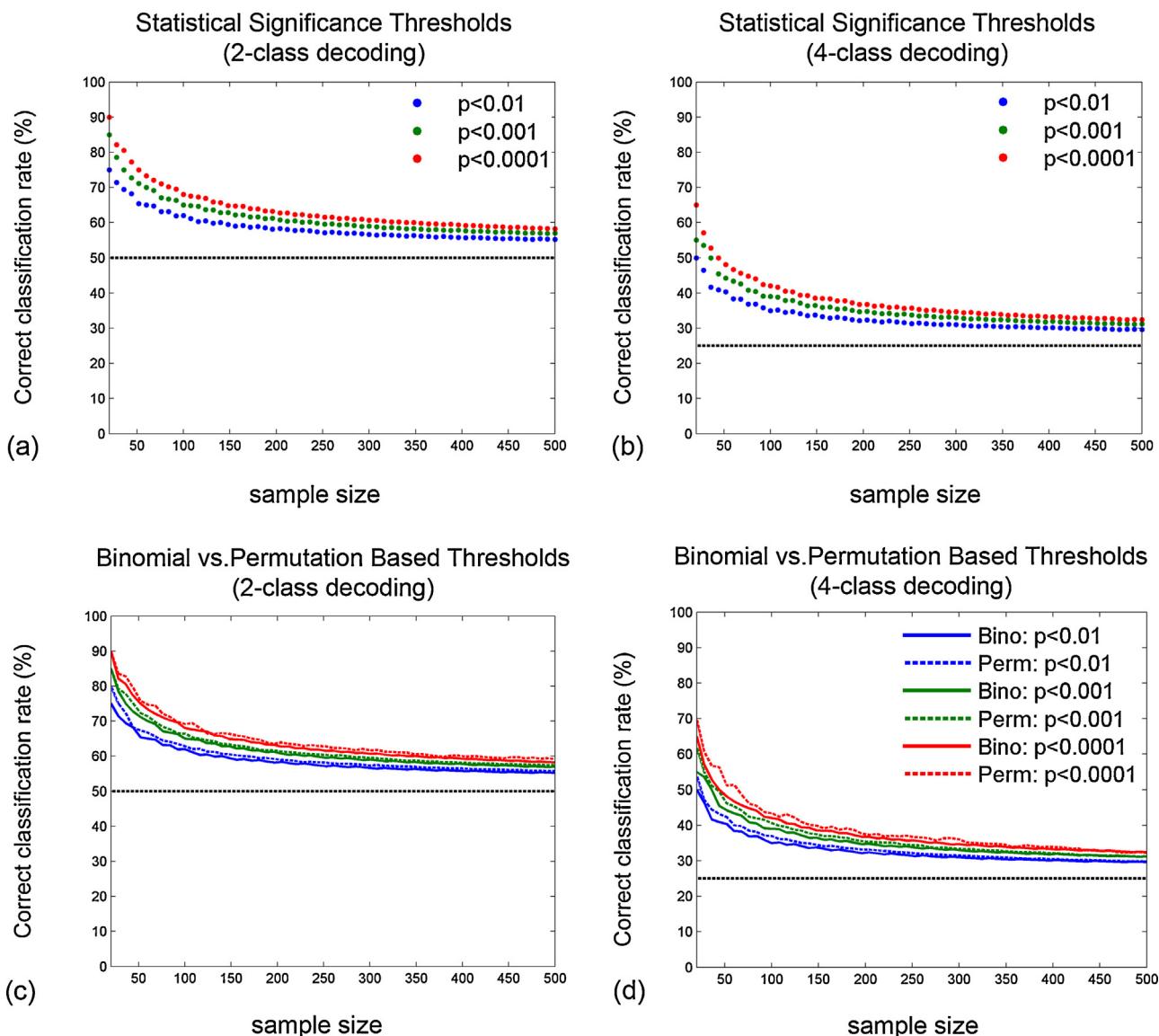
be more restrictive than the binomial formula. While this is a little more apparent for small values of  $n$ , the difference between the two methods rapidly vanishes as  $n$  increases.

### 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

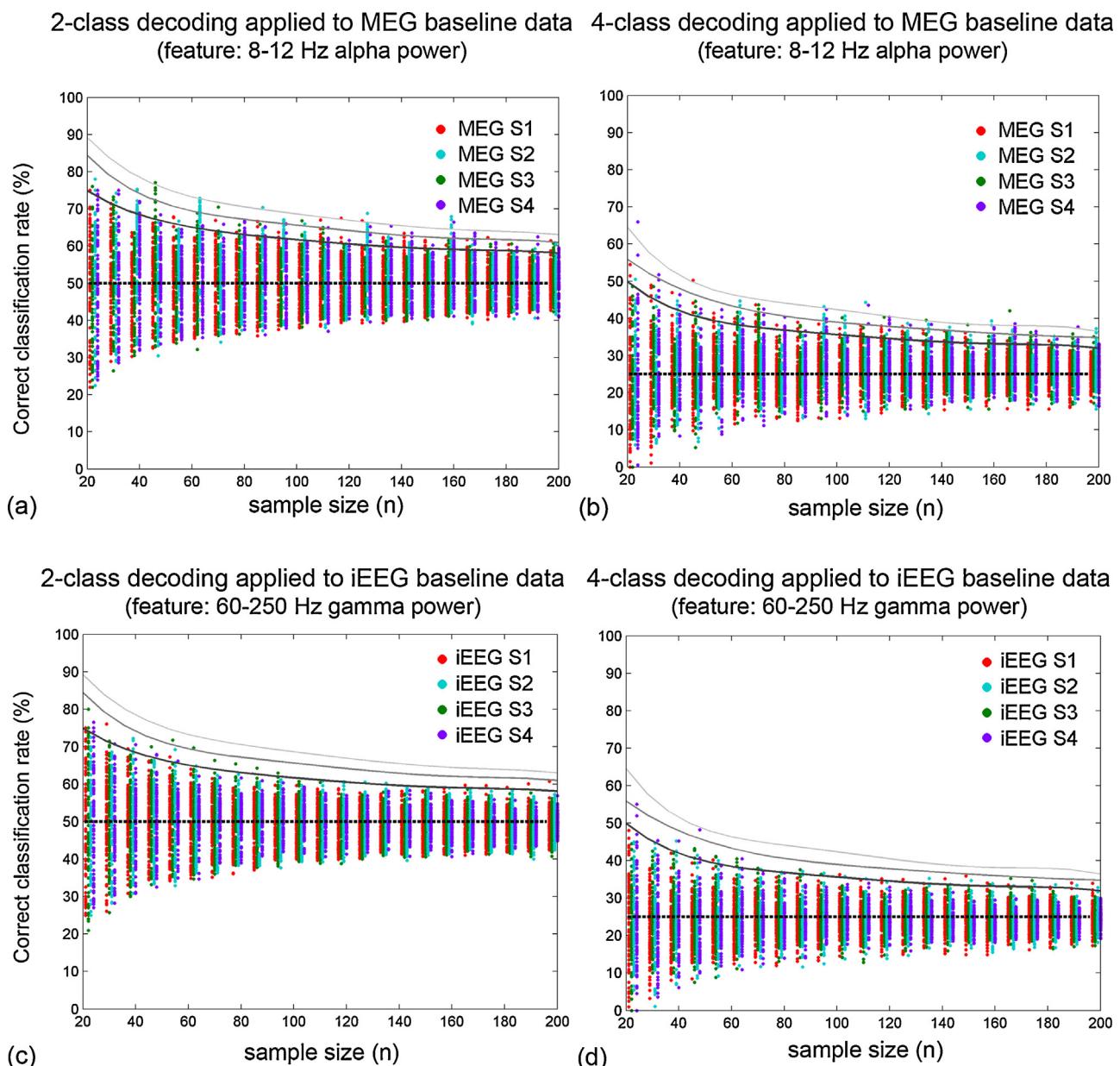


**Fig. 3.** Estimation of the statistical significance thresholds for 2- and 4-class classification as a function of sample size (assuming prediction errors are binomially distributed). Panels (a) and (b) show the evolution of the minimal statistically significant decoding rate as a function of sample 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; Pisto

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;8(June (2)):5549–52 [cited 2014 Jul 4]. Available from: [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=5626782](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, Mourino 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.
- Berrada 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, Martinier J, Garnier 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.
- Demandt 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/fnhum.2012.00065/full>.
- Derix J, Ilijina O, Schulze-Bonhage A, Aertsen A, Ball T. "Doctor" or "darling"? Decoding the communication partner from ECg 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)):1797–86.
- 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](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, Suguri 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-ICFBI 2007. It Meet 6th Int Symp On. IEEE; 2007a. p. 330–3. [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=4387767](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4387767).
- Jerbi K, Lachaux JP, 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, Lecaillard 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 JP, 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 JP, 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 Etetvelt 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, Quigadamo 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.

### **5.3 COMPLÉMENTS D'ÉTUDE**

compléments sur les différents types de permutation (Ojala and Garriga, 2010)



## **Troisième partie**

### **Étude 2 : encodage de l'intention et de l'exécution motrice**



## SOMMAIRE

5.1	PRÉSENTATION DE L'ÉTUDE . . . . .	41
5.1.1	Contexte . . . . .	41
5.1.2	Problématique . . . . .	41
5.1.3	Résultats majeurs . . . . .	41
5.2	ARTICLE . . . . .	41
5.3	COMPLÉMENTS D'ÉTUDE . . . . .	53
5.4	RÉSUMÉ DE L'ÉTUDE . . . . .	59
5.5	ARTICLE . . . . .	59
CONCLUSION . . . . .		59
5.6	RÉSUMÉ DE L'ÉTUDE . . . . .	65
5.7	ARTICLE . . . . .	65
CONCLUSION . . . . .		65
5.8	RÉSUMÉ DE L'ÉTUDE . . . . .	71
5.9	ARTICLE . . . . .	71
CONCLUSION . . . . .		71
5.10	RÉSUMÉ DE L'ÉTUDE . . . . .	77
5.11	ARTICLE . . . . .	77
CONCLUSION . . . . .		77

Ce chapitre introductif gnagnagna.  
Pas obligatoire!



**5.4 RÉSUMÉ DE L'ÉTUDE****5.5 ARTICLE****CONCLUSION DU CHAPITRE**

Ceci est la conclusion. Personnellement, je n'aime pas que la conclusion soit numéroté, mais je veux qu'elle apparaisse dans la table des matière, d'où la commande addcontentsline.



## **Quatrième partie**

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



## SOMMAIRE

5.1	PRÉSENTATION DE L'ÉTUDE . . . . .	41
5.1.1	Contexte . . . . .	41
5.1.2	Problématique . . . . .	41
5.1.3	Résultats majeurs . . . . .	41
5.2	ARTICLE . . . . .	41
5.3	COMPLÉMENTS D'ÉTUDE . . . . .	53
5.4	RÉSUMÉ DE L'ÉTUDE . . . . .	59
5.5	ARTICLE . . . . .	59
	CONCLUSION . . . . .	59
5.6	RÉSUMÉ DE L'ÉTUDE . . . . .	65
5.7	ARTICLE . . . . .	65
	CONCLUSION . . . . .	65
5.8	RÉSUMÉ DE L'ÉTUDE . . . . .	71
5.9	ARTICLE . . . . .	71
	CONCLUSION . . . . .	71
5.10	RÉSUMÉ DE L'ÉTUDE . . . . .	77
5.11	ARTICLE . . . . .	77
	CONCLUSION . . . . .	77

Ce chapitre introductif gnagnagna.  
Pas obligatoire!



**5.6 RÉSUMÉ DE L'ÉTUDE**

**5.7 ARTICLE**

**CONCLUSION DU CHAPITRE**

Ceci est la conclusion. Personnellement, je n'aime pas que la conclusion soit numéroté, mais je veux qu'elle apparaisse dans la table des matière, d'où la commande addcontentsline.



## **Cinquième partie**

### **Étude 4 : optimisation des paramètres de la bande gamma**



## SOMMAIRE

5.1	PRÉSENTATION DE L'ÉTUDE . . . . .	41
5.1.1	Contexte . . . . .	41
5.1.2	Problématique . . . . .	41
5.1.3	Résultats majeurs . . . . .	41
5.2	ARTICLE . . . . .	41
5.3	COMPLÉMENTS D'ÉTUDE . . . . .	53
5.4	RÉSUMÉ DE L'ÉTUDE . . . . .	59
5.5	ARTICLE . . . . .	59
CONCLUSION . . . . .		59
5.6	RÉSUMÉ DE L'ÉTUDE . . . . .	65
5.7	ARTICLE . . . . .	65
CONCLUSION . . . . .		65
5.8	RÉSUMÉ DE L'ÉTUDE . . . . .	71
5.9	ARTICLE . . . . .	71
CONCLUSION . . . . .		71
5.10	RÉSUMÉ DE L'ÉTUDE . . . . .	77
5.11	ARTICLE . . . . .	77
CONCLUSION . . . . .		77

Ce chapitre introductif gnagnagna.  
Pas obligatoire!



**5.8 RÉSUMÉ DE L'ÉTUDE**

**5.9 ARTICLE**

**CONCLUSION DU CHAPITRE**

Ceci est la conclusion. Personnellement, je n'aime pas que la conclusion soit numéroté, mais je veux qu'elle apparaisse dans la table des matière, d'où la commande addcontentsline.



## **Sixième partie**

### **Étude 5 : décodage des émotions**



## SOMMAIRE

5.1	PRÉSENTATION DE L'ÉTUDE . . . . .	41
5.1.1	Contexte . . . . .	41
5.1.2	Problématique . . . . .	41
5.1.3	Résultats majeurs . . . . .	41
5.2	ARTICLE . . . . .	41
5.3	COMPLÉMENTS D'ÉTUDE . . . . .	53
5.4	RÉSUMÉ DE L'ÉTUDE . . . . .	59
5.5	ARTICLE . . . . .	59
CONCLUSION . . . . .		59
5.6	RÉSUMÉ DE L'ÉTUDE . . . . .	65
5.7	ARTICLE . . . . .	65
CONCLUSION . . . . .		65
5.8	RÉSUMÉ DE L'ÉTUDE . . . . .	71
5.9	ARTICLE . . . . .	71
CONCLUSION . . . . .		71
5.10	RÉSUMÉ DE L'ÉTUDE . . . . .	77
5.11	ARTICLE . . . . .	77
CONCLUSION . . . . .		77

Ce chapitre introductif gnagnagna.  
Pas obligatoire!



**5.10 RÉSUMÉ DE L'ÉTUDE**

**5.11 ARTICLE**

**CONCLUSION DU CHAPITRE**

Ceci est la conclusion. Personnellement, je n'aime pas que la conclusion soit numéroté, mais je veux qu'elle apparaisse dans la table des matière, d'où la commande addcontentsline.



# DISCUSSION GÉNÉRALE

Enfin : la conclusion générale!!!

Au cours de ce mémoire, nous avons développé un modèle ...

1. **Modélisation**

2. **Inférence statistique**

## PERSPECTIVES

Dans la continuité directe de notre travail de thèse, nous pouvons ...

# A ANNEXES

## SOMMAIRE

A.1 COMPARATIF DE METHODES PAC (TORT ET AL., 2010) . . . . .	81
A.2 PIPELINE STANDARD DE CLASSIFICATION . . . . .	82
A.3 COMPARATIF DE CLASSIFIEURS (PEDREGOSA ET AL., 2011) . . . . .	84

## A.1 COMPARATIF DE METHODES 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.1 – Comparatif de méthodes PAC (Tort et al., 2010)

## A.2 PIPELINE STANDARD DE CLASSIFICATION

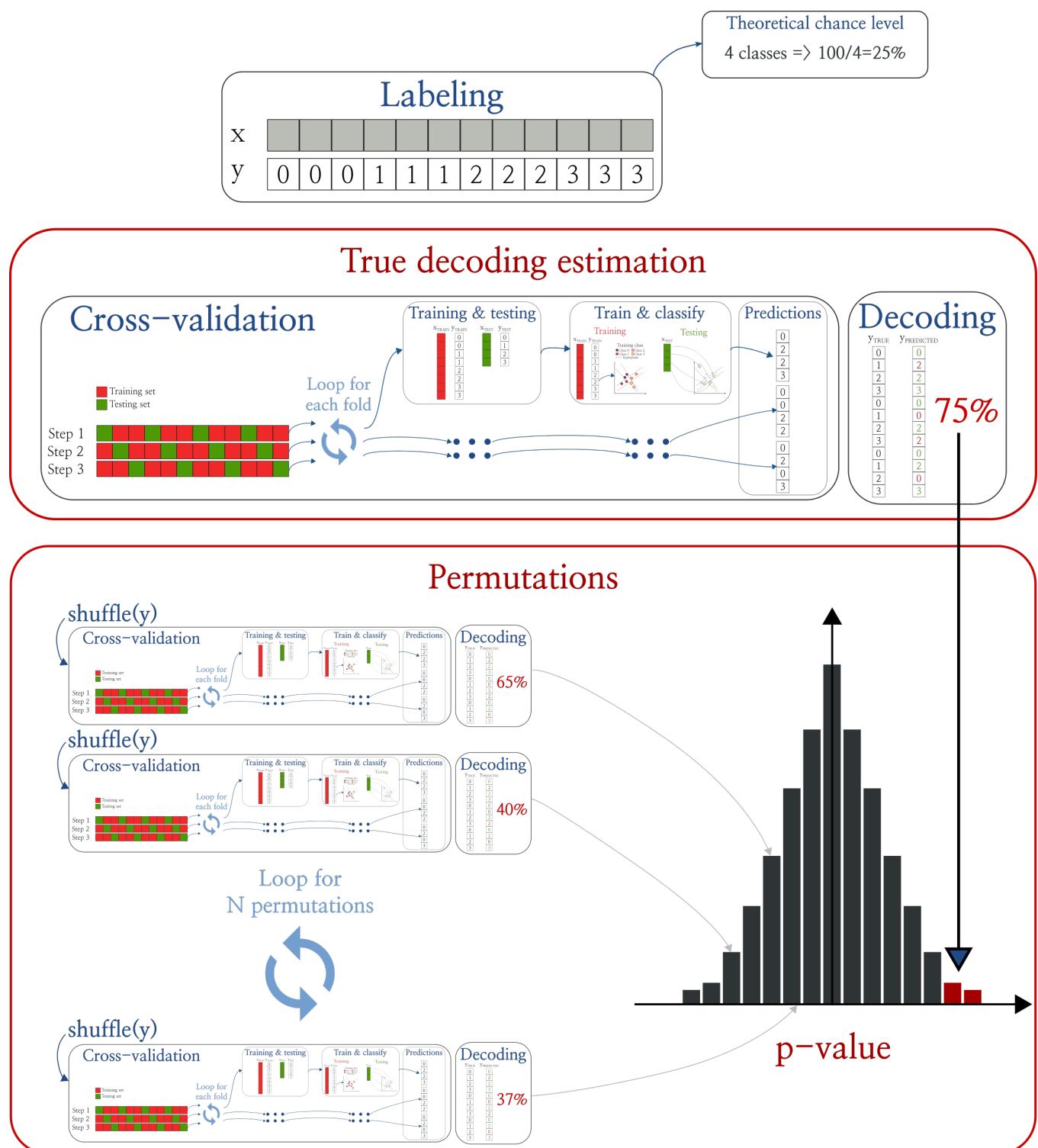


FIGURE A.2 – Pipeline standard de classification



### A.3 COMPARATIF DE CLASSIFIERS (PEDREGOSA ET AL., 2011)

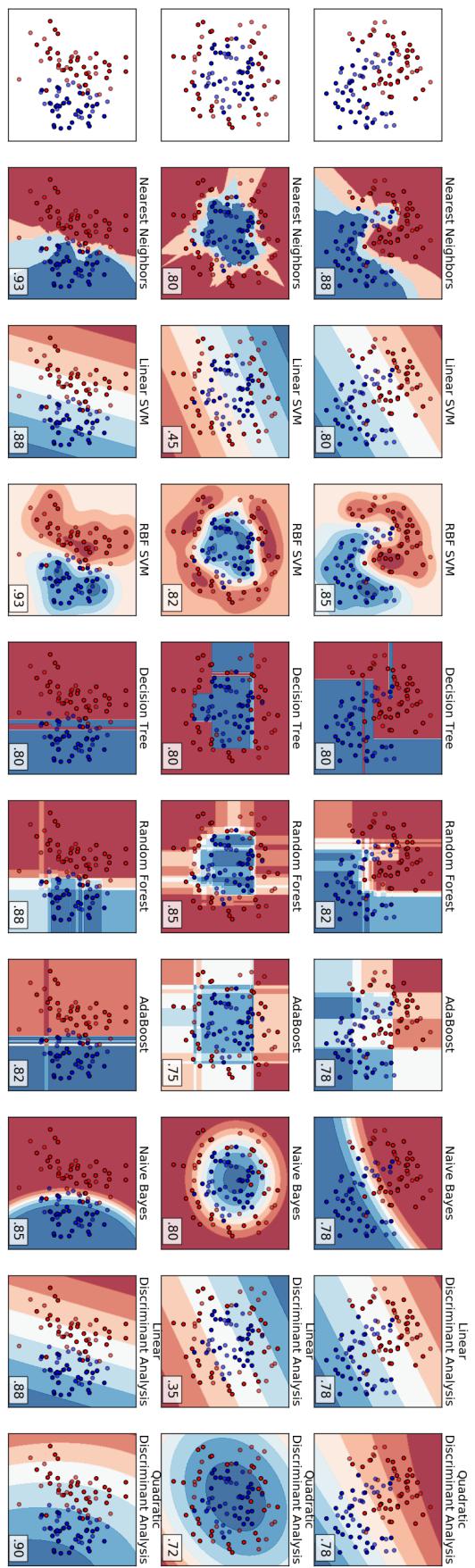


FIGURE A.3 – Comparatif de classifieurs (Pedregosa et al., 2011)

# BIBLIOGRAPHIE

- 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.
- Bekaert, M. H., Botte-Lecocq, C., Cabestaing, F., Rakotomamonjy, A., et al. (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.
- 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., Jerbi, K., Laurent, F., Baillet, S., Martinerie, J., Garnero, L., et al. (2007). Classification methods for ongoing EEG and MEG signals. *Biol Res*, 40(4) :415–437.
- 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.
- 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.
- 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.
- Cohen, M. X. (2008). Assessing transient cross-frequency coupling in EEG data. *Journal of Neuroscience Methods*, 168(2) :494–499.

- 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.
- 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*.
- 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.
- Ding, C. and Peng, H. (2005). Minimum Redundancy Feature Selection from Microarray Gene Expression Data.
- 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.
- 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.
- Fukunaga, K. (1990). Introduction to statistical pattern classification.
- Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3 :1157–1182.
- 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., Simmeral, 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.

- 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.
- 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.
- 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.
- 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.
- 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.
- Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F., Arnaldi, B., et al. (2007). A review of classification algorithms for EEG-based brain–computer interfaces. *Journal of neural engineering*, 4.
- 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.
- 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.

- 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.
- Rickert, J. (2005). Encoding of Movement Direction in Different Frequency Ranges of Motor Cortical Local Field Potentials. *Journal of Neuroscience*, 25(39) :8815–8824.
- Tallon-Baudry, C., Bertrand, O., Delpuech, C., and Pernier, J. (1997). Oscillatory  $\gamma$ -band (30–70 Hz) activity induced by a visual search task in humans. *The Journal of neuroscience*, 17(2) :722–734.
- 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.
- Van Langhenhove, A., Bekaert, M. H., Cabestaing, F., N'Guyen, J. P., et al. (2008). Interfaces cerveau-ordinateur et rééducation fonctionnelle : étude de cas chez un patient hémiplégique. *Sciences et Technologies pour le Handicap*, 2(1) :41–54.
- 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., D'Esposito, M., Crone, N., and Knight, R. T. (2013). A method for event-related phase/amplitude coupling. *NeuroImage*, 64 :416–424.
- 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.

- 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.
- 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, 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., Yamashita, O., Hirata, M., Kishima, H., Saitoh, Y., Goto, T., Yoshimine, T., and Kamitani, Y. (2012). 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.

**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