



Université du Québec  
à Rimouski

# **RECONNAISSANCE AUTOMATIQUE DES GESTES DE LA MAIN PAR ANALYSE DES SIGNAUX EMG DE SURFACE**

**MÉMOIRE PRÉSENTÉ**

dans le cadre du programme de maîtrise en ingénierie

en vue de l'obtention du grade de maître ès sciences appliquées (M.Sc.A.)

PAR

**© ABDELAZIZ HAMMOUDI**

**Décembre 2023**



**Composition du jury :**

**Tan-Sy Nguyen (Ph.D.), président du jury, Université du Québec à Rimouski**

**Mohammed Bahoura (Ph.D.), directeur de recherche, Université du Québec à Rimouski**

**François Nougrou (Ph.D.), examinateur externe, Université du Québec à Trois-Rivières**

Dépôt initial le [Décembre 2023]

Dépôt final le [Mars 2024]



UNIVERSITÉ DU QUÉBEC À RIMOUSKI  
Service de la bibliothèque

Avertissement

La diffusion de ce mémoire ou de cette thèse se fait dans le respect des droits de son auteur, qui a signé le formulaire « *Autorisation de reproduire et de diffuser un rapport, un mémoire ou une thèse* ». En signant ce formulaire, l'auteur concède à l'Université du Québec à Rimouski une licence non exclusive d'utilisation et de publication de la totalité ou d'une partie importante de son travail de recherche pour des fins pédagogiques et non commerciales. Plus précisément, l'auteur autorise l'Université du Québec à Rimouski à reproduire, diffuser, prêter, distribuer ou vendre des copies de son travail de recherche à des fins non commerciales sur quelque support que ce soit, y compris Internet. Cette licence et cette autorisation n'entraînent pas une renonciation de la part de l'auteur à ses droits moraux ni à ses droits de propriété intellectuelle. Sauf entente contraire, l'auteur conserve la liberté de diffuser et de commercialiser ou non ce travail dont il possède un exemplaire.



À mon père

À ma mère

À mes frères

À mes sœurs

À mes amis



## **REMERCIEMENTS**

Je tiens à exprimer ma sincère gratitude à mon directeur de recherche, Monsieur Mohammed Bahoura, professeur au Département de Mathématiques, d'Informatique et de Génie (DMIG) de l'Université du Québec à Rimouski (UQAR), pour son soutien constant et bienveillant, ses conseils, son encouragement, et pour avoir suivi ce travail avec un grand intérêt. Cette réalisation n'aurait pas été possible sans les précieuses perspectives qu'il m'a fournies tout au long de ce parcours.

Je souhaite également remercier le professeur Tan-Sy Nguyen de l'Université du Québec à Rimouski pour avoir accepté d'évaluer mon mémoire.

De plus, mes remerciements vont au professeur François Nougrou de l'Université du Québec à Trois-Rivières pour avoir accepté d'évaluer mon mémoire.

Finalement, je tiens à exprimer ma profonde gratitude envers mes parents, mes frères et sœurs, ainsi que mes amis, pour leur soutien inébranlable et leur encouragement continu tout au long de mon parcours universitaire.



## RÉSUMÉ

Ce mémoire présente le processus de conception et d'implémentation d'un système de classification des gestes de la main basé sur les signaux électromyographique (EMG) de surface à l'aide de modèles d'apprentissage machine. Le mémoire couvre différentes étapes, allant de l'acquisition des signaux EMG à la classification finale. Il commence par fournir une compréhension de l'électromyographie (EMG), qui est le potentiel électrique généré par les fibres musculaires pendant la contraction ou la relaxation. Différentes méthodes d'acquisition des signaux EMG sont explorées, y compris les systèmes d'acquisition couramment utilisés, ainsi que de bases de données EMG populaires. Pour extraire efficacement les informations pertinentes des signaux, une technique de découpage en fenêtres est appliquée, suivie de l'extraction des caractéristiques clés de chaque fenêtre. Ensuite, l'accent est mis sur l'apprentissage machine, qui a gagné en popularité dans différents domaines de classification. Quatre algorithmes spécifiques, à savoir les machines à vecteurs de support (SVM), le perceptron multicouche (MLP), les k plus proches voisins (k-NN) et l'analyse discriminante linéaire (LDA), sont discutés en détail. Ce projet de recherche applique ces algorithmes à la reconnaissance des gestes de la main basée sur les signaux EMG en utilisant notre base de données. De plus, ils sont évalués sur la base de données EMG, obtenus du référentiel UCI (*University of California at Irvine*) pour évaluer leurs capacités de classification.

Le système a été testé et évalué sous MATLAB dans les conditions réelles d'utilisation. Les résultats indiquent que la reconnaissance des gestes de la main basée sur les signaux EMG à l'aide d'algorithmes d'apprentissage machine est une approche prometteuse pour reconnaître avec précision les gestes de la main dans les conditions réelles. L'étude souligne le potentiel des algorithmes d'apprentissage machine pour améliorer l'exactitude de la reconnaissance des gestes par rapport à d'autres outils ou approches, ce qui permet le développement d'interfaces homme-machine (HMI) innovantes et efficaces.

Finalement, cette étude démontre le potentiel de la reconnaissance des gestes de la main basée sur l'EMG à l'aide d'algorithmes d'apprentissage machine et fournit des informations précieuses pour les futures recherches dans ce domaine. Il subsiste encore des défis à surmonter concernant l'acquisition des données, l'extraction des caractéristiques et l'optimisation des algorithmes, dans le but d'améliorer le fonctionnement dans des conditions réelles.

Mots clés : Classification des gestes de la main basée sur l'EMG, apprentissage machine, électromyographie, extraction de caractéristiques, test en condition réelles.



## ABSTRACT

This dissertation presents the process of designing and implementing a hand gesture classification system based on surface electromyographic (sEMG) signals using machine learning models. The dissertation covers various stages, starting from the acquisition of EMG signals to the final classification. It begins by providing an understanding of electromyography (EMG), which is the electrical potential generated by muscle fibers during contraction or relaxation. Different methods of capturing EMG signals are explored, including commonly used acquisition systems, along with popular EMG databases. To extract relevant information from the signals efficiently, a windowing technique is used, followed by the extraction of key features from each window. Subsequently, the focus shifts to machine learning, which has gained popularity in various classification fields. Four specific algorithms, namely support vector machines (SVM), multilayer perceptron (MLP),  $k$ -nearest neighbors (k-NN) and linear discriminant analysis (LDA), are discussed in detail. This research project applies these algorithms to EMG-based hand gesture recognition using our database. Additionally, these algorithms are evaluated on the EMG database obtained from the UCI (University of California at Irvine) repository to assess their classification capabilities.

The system was tested and evaluated with MATLAB under real conditions of use. The results indicate that EMG-based hand gesture recognition using machine learning algorithms is a promising approach for accurately recognizing hand gestures in real conditions. The study highlights the potential of machine learning algorithms to improve the accuracy of gesture recognition compared with other approaches, enabling the development of innovative and effective human-machine interfaces (HMIs).

Finally, this research highlights the potential of EMG-based hand gesture recognition using machine learning algorithms, and provides valuable information for future research in this field. However, challenges still lie ahead in terms of data acquisition, feature extraction and algorithm optimization, with the aim of improving performance under real conditions.

Keywords: EMG-based hand gesture classification, machine learning, electromyography, feature extraction, test in real conditions.



## TABLE DES MATIÈRES

REMERCIEMENTS.....	ix
RÉSUMÉ.....	xi
ABSTRACT.....	xiii
TABLE DES MATIÈRES.....	xv
LISTE DES TABLEAUX.....	xix
LISTE DES FIGURES.....	xxi
LISTE DES ABRÉVIATIONS.....	xxv
INTRODUCTION GÉNÉRALE.....	1
CHAPITRE 1 MUSCLES SQUELETTIQUES ET SIGNAUX EMG.....	3
1.1 INTERACTION HOMME-MACHINE.....	3
1.2 GENERALITES SUR LES MUSCLES.....	4
1.2.1 Définition et types de muscles.....	4
1.2.2 muscles squelettiques.....	5
1.2.3 cerveau et muscles.....	6
1.2.4 Mouvement des mains.....	7
1.2.5 Signal électromyographique (EMG).....	8
1.2.6 Unité motrice.....	9
1.2.7 Signal électromyographique (EMG) de la main.....	10
1.2.8 Acquisition du signal EMG.....	11
1.2.9 Bases de données populaires.....	12
1.3 PROBLEMATIQUE DE RECHERCHE.....	17
1.4 OBJECTIFS.....	18
1.5 HYPOTHESE.....	18
1.6 METHODOLOGIE.....	19
1.7 CONTRIBUTION.....	21

CHAPITRE 2 PRÉTRAITEMENT DU SIGNAL EMG ET EXTRACTION DES CARACTÉRISTIQUES.....	23
2.1 PRETRAITEMENT .....	23
2.1.1 Reduction du bruit.....	24
2.1.2 Segmentation des signaux.....	24
2.1.3 Fenêtrage.....	26
2.2 EXTRACTION DES CARACTERISTIQUES.....	27
2.2.1 Caractéristiques du domaine temporel.....	27
2.2.2 Caractéristiques du domaine fréquentiel.....	30
2.3 APPROCHE PROPOSÉ .....	32
CHAPITRE 3 ALGORITHMES D'APPRENTISSAGE MACHINE POUR LES TACHES DE CLASSIFICATION AUTOMATIQUE.....	35
3.1 MACHINE A VECTEURS DE SUPPORT (SVM) .....	36
3.1.1 Classification linéaire.....	36
3.1.2 Classificateur SVM non linéaire.....	40
3.1.3 Fonction noyau.....	41
3.1.4 Multi-classification par SVM .....	42
3.2 PERCEPTRON MULTICOUCHES ( <i>MLP</i> ) .....	45
3.2.1 Perceptron ou Neurone Artificiel .....	47
3.2.2 Fonction d'activation .....	47
3.2.3 L'algorithme de rétropropagation.....	51
3.3 METHODE DU K PLUS PROCHES VOISINS (K-NN).....	52
3.3.1 Calcul de la distance .....	55
3.4 ANALYSE DISCRIMINANTE LINEAIRE LDA .....	56
3.5 MATRICE DE CONFUSION .....	58
3.6 APPROCHE PROPOSEE POUR LA CLASSIFICATION .....	60
CHAPITRE 4 RESULTATS EXPERIMENTAUX.....	61
4.1 OUTILS ET MATERIEL UTILISEES .....	61
4.2 MYO GESTURE ARMBAND.....	62
4.3 BASE DE DONNEES.....	64

4.4	PRETRAITEMENT ET EXTRACTION DES CARACTERISTIQUES .....	66
4.4.1	Prétraitement et d'extraction des caractéristiques.....	66
4.4.2	Validation croisée.....	67
4.4.3	Choix de la méthode d'extraction des caractéristiques .....	69
4.4.4	Choix de la longueur de fenêtre et du taux de chevauchement .....	70
4.5	RESULTATS DE LA CLASSIFICATION.....	73
4.5.1	Machine a vecteur de support (SVM) .....	73
4.5.2	k-plus proches voisins (k-NN).....	75
4.5.3	Analyse discriminante linéaire (LDA) .....	76
4.5.4	Perceptron multicouche (MLP) .....	77
4.6	COMPARAISON DES RESULTATS .....	81
4.7	COMPARAISON DES RESULTATS AVEC D'AUTRE TRAVAUX UTILISANT LE MEME CAPTEUR MYO GESTURE ARMBAND.....	82
4.8	COMPARAISON DES RESULTATS DE CLASSIFICATION AVEC CEUX DES AUTRES CHERCHEURS EN UTILISANT LA BASE DE DONNES UCI.....	83
4.9	RECONNAISSANCE DES GESTES DANS LES CONDITIONS REELLES .....	86
	CONCLUSION GÉNÉRALE.....	87
	RÉFÉRENCES BIBLIOGRAPHIQUES.....	89



## LISTE DES TABLEAUX

Tableau 1. 1 Ensembles de données NINAPRO.....	13
Tableau 3. 1 Matrice de confusion en classification binaire.....	59
Tableau 3. 2 Matrice de confusion en multiclasse.....	59
Tableau 4. 1 Effet du choix du technique de caractérisation sur les performances de classification.....	70
Tableau 4. 2 Étude de l'effet de variation de la taille des fenêtres et du taux de chevauchement sur l'exactitude de la classification. ....	71
Tableau 4. 3 Exactitude de la classification à différents nombres de voisins k.....	75
Tableau 4. 4 Exactitude de la classification avec différents nombres de couches cachées et de perceptrons. ....	79
Tableau 4. 5 Comparaison des modèles proposés dans cette recherche. ....	82
Tableau 4. 6 Performance de classification obtenue par des chercheurs utilisant le brassard Myo Gesture Armband avec une fréquence d'échantillonnage de 200Hz. ....	83
Tableau 4. 7 Performances de classification obtenues par des études utilisant la base de données de l'UCI. ....	84
Tableau 4. 8 Variation de l'exactitude avec l'algorithme k-NN en fonction du nombre de sujets de la base de données UCI, en sélectionnant les premiers sujets de la base de données. ....	85



## LISTE DES FIGURES

Figure 1. 1 Types de muscles .....	5
Figure 1. 2 Structure d'un muscles squelettique .....	6
Figure 1. 3 Schéma simplifié de la division du système nerveux somatique en relation avec le cerveau et la moelle épinière humains .....	7
Figure 1. 4 Les muscles de l'avant-bras qui provoquent le mouvement .....	8
Figure 1. 5 Mesure des signaux électriques musculaires à l'aide de la technologie EMG .....	9
Figure 1. 6 Unité motrice .....	10
Figure 1. 7 Électromyographie (EMG) invasive .....	11
Figure 1. 8 Électromyographie (EMG) de surface . .....	12
Figure 1. 9 Détails des 52 mouvements de la main de NINAPRO DB1 . .....	14
Figure 1. 10 Illustrations iconiques des 27 gestes effectués contenus dans la base de données SCL-HDEMG. ....	15
Figure 1. 11 Processus de la reconnaissance des gestes de la main.....	20
Figure 2. 1 Représentation des étapes de traitement du signal : acquisition des données, pré-traitement des données et extraction des caractéristiques.....	23
Figure 2. 2 Segmentation du signal par: (a) sans chevauchement; (b) avec chevauchement .....	25
Figure 2. 3 Illustration du processus d'extraction des caractéristiques du signal EMG.....	32
Figure 3. 1 Exemple de classification linéaire. Les deux caractéristiques représentent les composantes du vecteur $x$ .....	37
Figure 3. 2 Les hyperplans de séparation et les marges.....	38
Figure 3. 3 Exemple de noyau pour deux classes. ....	42
Figure 3. 4 Exemple de multi classification SVM utilisant la stratégie un contre tous. ....	43
Figure 3. 5 Exemple de multi classification SVM utilisant la stratégie un contre un. ....	45

Figure 3. 6 Architecture générale d'un réseau de neurones avec une seule couche cachée. Il contient $N_e = 4$ entrées, $N_c = 5$ neurones cachés et $N_s = 2$ neurones de sortie. ....	46
Figure 3. 7 Représentation d'un perceptron.....	47
Figure 3. 8 Fonction sigmoïde.....	48
Figure 3. 9 Fonction tangente hyperbolique.....	49
Figure 3. 10 Fonction ReLU. ....	50
Figure 3. 11 Représentation schématique de l'application de la fonction d'activation Softmax dans un réseau de neurones pour la catégorisation multi-classes. ....	51
Figure 3. 12 Règle de décision k-NN pour les valeurs $k = 1, k = 3$ et $k = 7$ .....	54
Figure 3. 13 Exemple d'utilisation de l'analyse discriminante linéaire. ....	56
Figure 3. 14 Différence entre les modes de projection dans LDA: (a) Maximisation seulement de la distance entre les moyennes; (b) Maximisation la distance entre les moyennes et minimiser la covariance. ....	58
Figure 4. 1 Myo Gesture Armband developpé par Thematic Lab. ....	63
Figure 4. 2 Représentation des signaux EMG à 8 canaux collectés à partir du brassard Myo Gesture Armband. ....	63
Figure 4. 3 Placement du brassard Myo Gesture Armband avec vue en coupe des muscles de l'avant-bras . ....	64
Figure 4. 4 Illustrations des gestes effectués et les signaux EMG associés dans les 8 canaux. ....	65
Figure 4. 5 Représentation de la préparation de la base de données. ....	66
Figure 4. 6 Illustration du processus de fenêtrage et d'extraction des caractéristiques.....	67
Figure 4. 7 Représentation d'un exemple illustratif de la validation croisée pour $k = 5$ . ....	68
Figure 4. 8 Variation de l'exactitude en fonction de la taille de la fenêtre et de chevauchement en utilisant k-NN.....	72
Figure 4. 9 Variation de l'exactitude en fonction de la taille des fenêtres en utilisant classificateur SVM, k-NN et MLP. ....	72
Figure 4. 10 Matrice de confusion pour SVM de noyau polynomial d'ordre 3.....	74

Figure 4. 11 Matrice de confusion pou k-NN à 3 plus proches voisins.....	76
Figure 4. 12 Matrice de confusion pour LDA à hyperplan quadratique.....	77
Figure 4. 13 Représentation de processus Cross-Validation $k = 10$ avec le classificateur MLP.....	78
Figure 4. 14 Matrice de confusion pour MLP. ....	81
Figure 4. 15 Représentation du résultat dans les conditions réelles d'utilisation. ....	86



## LISTE DES ABRÉVIATIONS

<i>AAC</i>	: <i>Average Amplitude Change</i>	: Variation de l'amplitude moyenne
<i>AMPSO-SVM</i>	: <i>Adaptive mutation particle swarm optimization SVM</i>	: Mutation adaptative Optimisation par essais de particules SVM
<i>ANN</i>	: <i>Artificial neural network</i>	: Réseau de neurones artificiels
<i>DASDV</i>	: <i>Deference absolute standard: deviation value</i>	: Déférence écart type absolu valeur
<i>DT</i>	: <i>Decision tree</i>	: Arbre de décision
<i>GNB</i>	: <i>Gaussian naive bayes</i>	: Bayes naïves gaussiennes
<i>GS-SVM</i>	: <i>Grid search-SVM</i>	: Recherche de grille-SVM
<i>IAV</i>	: <i>Integrated Absolute Value</i>	: Valeur Absolue Intégrée
<i>K-NN</i>	: <i>K-nearest neighbors</i>	: k-plus proches voisins
<i>LDA</i>	: <i>Linear Discriminant Analysis</i>	: Analyse discriminante linéaire
<i>LR</i>	: <i>Linear regression</i>	: Régression linéaire
<i>MAV</i>	: <i>Mean absolute value</i>	: Valeur absolue moyenne
<i>MFL</i>	: <i>Maximum fractal length</i>	: Longueur fractale maximale
<i>MLP</i>	: <i>Multilayer perceptron</i>	: Perceptron multicouche
<i>MMAV</i>	: <i>Modified mean absolute value</i>	: Valeur absolue moyenne modifiée
<i>MMAV2</i>	: <i>Modified mean absolute value type 2</i>	: Valeur absolue moyenne modifiée type 2

MNF	: <i>Mean Frequency</i>	: Fréquence Moyenne
MNP	: <i>Mean Power</i>	: Puissance moyenne
PKF	: <i>Peak Frequency</i>	: Fréquence de pic
PSO-SVM	: <i>Particle swarm optimization SVM</i>	: Optimisation par essaim particulaire SVM
QDA	: <i>Quadratique Discriminant Analysis</i>	: Analyse discriminante quadratique
RF	: <i>Random forest</i>	: Forêt aléatoire
RMS	: <i>Root mean square</i>	: Moyenne quadratique
SD	: <i>Standard deviation</i>	: Écart-type
SM	: <i>Spectral Moment</i>	: Moment spectral
SSC	: <i>Slop sign change</i>	: Changement de signe de la pente
SSI	: <i>Simple Square Integral</i>	: Intégrale carrée simple
SVM	: <i>Support vector machine</i>	: Machine à vecteur de support
WAMP	: <i>Willison Amplitude</i>	: Amplitude de Willison
WL	: <i>Window length</i>	: Longueur de la fenêtre
ZC	: <i>Zero crossing</i>	: Passage à zéro

## INTRODUCTION GÉNÉRALE

Afin d'aider un grand nombre de personnes à mobilité réduite, des membres artificiels tels que des prothèses ont été spécialement conçus. Depuis des décennies et jusqu'à aujourd'hui, la recherche a fait en sorte que les prothèses puissent accomplir des tâches aussi parfaitement que les membres naturels. Elles sont devenues de plus en plus légères, plus robustes, plus esthétiques et ressemblant de plus en plus aux organes humains, permettant de retrouver une partie de la qualité de vie perdue à cause de différentes déficiences physiques. Une solution pour rendre ces prothèses intuitives et intelligentes est la reconnaissance de mouvements basée sur les informations contenues dans le signal électrique captés aux niveaux des muscles appelé électromyographie (EMG) (Marks & Michael, 2001; Ortiz-Catalan *et al.*, 2012).

Les signaux électromyographique (EMG) sont utilisés dans de nombreuses applications, notamment le diagnostic des maladies neuromusculaires, le contrôle des prothèses/orthèses, les interfaces homme-machine, les jeux en réalité virtuelle et le développement d'équipements d'excitation axés sur les muscles. Diverses études ont été réalisées dans lesquelles les signaux EMG ont été utilisés pour interagir avec des machines/ordinateurs (Gomez-Gil *et al.*, 2011; Xiong *et al.*, 2021). L'objectif de la recherche sur la reconnaissance des gestes est de développer un système capable de discerner des mouvements humains spécifiques et de les utiliser pour transmettre des informations ou contrôler un appareil (Ahsan *et al.*, 2011). En général, le signal EMG présente la réaction d'un muscle en temps réel ; l'amplitude du signal EMG représente la façon dont le muscle se comporte, ce qui permet aux chercheurs d'établir un diagnostic rapide sur le sujet. Le signal EMG recueilli lors d'une contraction musculaire régissant un geste de la main contient de nombreuses informations sur les muscles qui peuvent être utilisés pour décrire le comportement et l'état de ces muscles, ainsi que le mouvement de la main (Zheng *et al.*, 2022).

La procédure de reconnaissance de mouvement basée sur le signal EMG comprend plusieurs étapes. L'étape initiale consiste à acquérir les signaux EMG (Benatti *et al.*, 2015; Wang *et al.*, 2013). La deuxième étape concerne le prétraitement de ces signaux qui consiste à les découper en segments avec un chevauchement partiel suivi d'un fenêtrage (Tepe & Demir, 2021; Wahid *et al.*, 2020). La troisième étape consiste à extraire les caractéristiques de chacun des segments. Trois domaines sont couramment utilisés pour extraire les caractéristiques des signaux EMG: le domaine temporel, le domaine fréquentiel et le domaine temps-fréquence, les caractéristiques du domaine temporel sont simples et rapides à implémenter puisqu'ils sont directement extraits des données brutes d'une série temporelle, évitant ainsi une complexité de calcul élevée (Phinyomark *et al.*, 2013). La dernière partie est l'étape de classification des signaux à partir desquels les caractéristiques ont été extraites. De nombreux algorithmes ont été développés pour classifier les données EMG. Parmi ceux-ci, nous pouvons citer : la forêt aléatoire RF (*Random Forest*) (Wahid *et al.*, 2018), les k-plus proches voisins k-NN (*k-Nearest Neighbors*) (Shi *et al.*, 2018), l'analyse discriminante linéaire LDA (*Linear Discriminant Analysis*), le réseau neuronal à perceptron multicouche MLP (*Multi-Layer Perceptron*) (Colli & Trejos, 2022), la machine à vecteurs de support SVM (*Support Vector Machine*) (Tepe & Demir, 2022) et les réseaux neuronaux convolutionnels CNN (*Convolutional Neural Networks*) (Fajardo *et al.*, 2021).

Notre mémoire s'organise de la manière suivante : Le premier chapitre offre une introduction aux muscles squelettiques et aux signaux EMG, énonce la problématique ainsi que les objectifs du projet, et décrit les hypothèses et la méthodologie adoptée. Le deuxième chapitre est dédié à la présentation des techniques de prétraitement et d'extraction de caractéristiques fréquemment citées dans la littérature. Le troisième chapitre explore certains algorithmes de classification utilisés dans les recherches récentes. Le quatrième chapitre décrit les résultats obtenus avec la mise en œuvre dans des conditions réelles. Enfin, nous terminons avec une conclusion générale concernant le travail réalisé et les perspectives d'amélioration.

# CHAPITRE 1

## MUSCLES SQUELETTIQUES ET SIGNAUX EMG

Dans ce chapitre, nous explorons les concepts fondamentaux des muscles squelettiques et des signaux électromyographiques (EMG). Nous commençons par définir les muscles et discuter de leurs types, en mettant l'accent sur les muscles squelettiques et leurs fonctions. La relation entre le cerveau et les muscles dans la génération du mouvement, particulièrement celui de la main. Nous examinons également l'importance de l'analyse des signaux EMG pour l'étude de l'activité musculaire et du contrôle moteur, ainsi que le concept d'unités motrices. En outre, nous discutons des caractéristiques et des applications des signaux EMG de la main, ainsi que des méthodologies d'acquisition de ces signaux. Relever les défis de l'analyse des signaux EMG contribuera à des avancées dans des domaines tels que les prothèses, la rééducation et l'interaction homme-machine (HMI).

### 1.1 INTERACTION HOMME-MACHINE

La progression rapide de l'intelligence artificielle AI (*Artificial Intelligence*) et de l'internet des objets IoT (*Internet of Things*) transforme notre vie quotidienne de manière significative et essentielle. Au cœur de cette évolution se trouve l'interaction forte et précise entre les humains et les machines (Lin *et al.*, 2022).

L'interaction homme-machine HMI (*Human-Machine Interaction*) décrit la manière dont les personnes et les machines communiquent et interagissent. Il s'agit d'un vaste domaine qui inclut les interactions entre les êtres humains et les robots, les ordinateurs, et intègre des domaines tels que l'intelligence artificielle et la robotique. Traditionnellement, la technologie HMI a été utilisée dans des environnements industriels pour améliorer l'efficacité, la qualité

et la sécurité. Cependant, ses applications se sont étendues aux systèmes médicaux, de transport et de divertissement (Ke *et al.*, 2018).

L'électromyographie (EMG) est capable de détecter l'activité musculaire avant qu'elle ne devienne apparente, ce qui en fait un outil précieux dans l'interaction homme-machine (HMI) pour anticiper les intentions humaines (Xue & Lai, 2023). L'intégration de l'électromyographie (EMG) dans les systèmes HMI est motivée par sa capacité à permettre une interaction plus instinctive et naturelle avec les machines, ce qui est particulièrement important pour la gestion des systèmes robotiques et des technologies d'assistance (Ferreira *et al.*, 2008; Gomez-Gil *et al.*, 2011). Les systèmes HMI basés sur l'EMG fonctionnent en capturant et en traitant les signaux EMG, ce qui permet de créer un lien de communication entre les humains et les machines (Gomez-Gil *et al.*, 2011; Xiong *et al.*, 2021). Cette technologie est particulièrement utile pour les amputés, en améliorant leur interaction avec divers objets et équipements, notamment les prothèses.

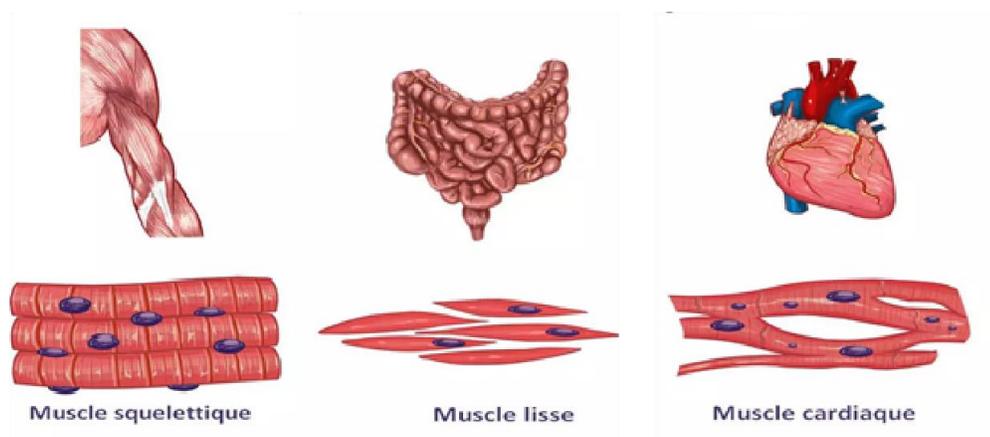
## **1.2 GENERALITES SUR LES MUSCLES**

### **1.2.1 DÉFINITION ET TYPES DE MUSCLES**

Le corps humain comprend trois réseaux principaux de tissus musculaires spécialisés qui jouent un rôle essentiel dans différentes fonctions corporelles. Ces réseaux sont responsables du mouvement de nos os, de nos vaisseaux sanguins, de nos organes internes et bien d'autres choses encore, fonctionnant de manière continue et ininterrompue. Avec plus de huit cents muscles répartis dans ces trois réseaux, ils génèrent du mouvement et de la chaleur par des contractions ou des relaxations, en fonction de leur utilisation (Colligan, 2010).

Il existe trois systèmes musculaires principaux dans le corps humain (voir Figure 1. 1): le système musculaire squelettique, le système musculaire lisse et le système musculaire cardiaque. Le système musculaire squelettique joue un rôle essentiel dans le maintien de la

position verticale de nos os. Il s'agit d'un système volontaire qui répond à nos commandes intentionnelles, ce qui nous permet de réaliser des actions telles que serrer le poing ou courir. Le système musculaire lisse, par contre, couvre plusieurs organes internes, y compris le foie et les reins. Il a pour fonction de faciliter le mouvement de substances telles que le sang, les aliments et les déchets dans l'ensemble du corps. Contrairement au système musculaire squelettique, le système musculaire lisse est involontaire, ce qui signifie que nous n'avons pas de contrôle conscient sur son fonctionnement. De même, le système musculaire cardiaque, également involontaire, est chargé d'aider le cœur à pomper le sang dans tout le corps. Les systèmes musculaires lisse et cardiaque fonctionnent automatiquement, indépendamment de notre influence consciente.



### 1.2.2 MUSCLES SQUELETTIQUES

Également appelés muscles somatiques ou volontaires, les muscles squelettiques représentent environ 40 % du poids total du corps humain. Leur principale fonction est de générer des mouvements en se contractant et en se relâchant de manière synchronisée. Ces muscles sont reliés aux os soit directement, soit, plus couramment, par des tendons. Lorsque le muscle se contracte, il transfère la tension aux os à travers une ou plusieurs articulations, ce qui entraîne un mouvement (Jarmey, 2018).

La Figure 1. 2 illustre la structure des muscles squelettiques, où le muscle s'attache à l'os par l'intermédiaire du tendon.

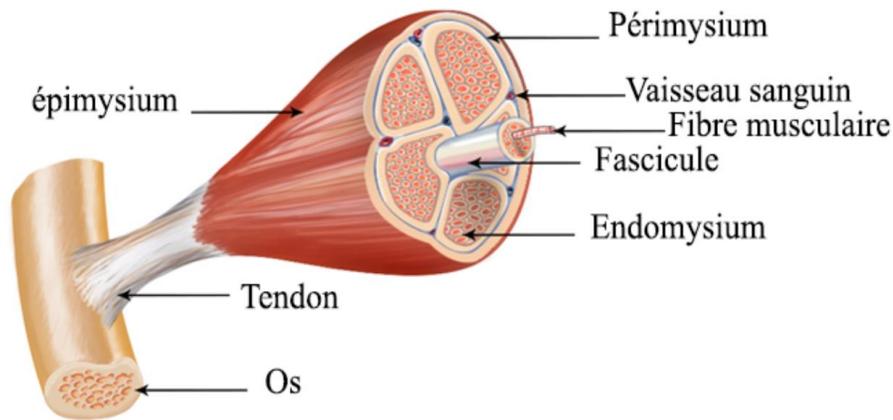


Figure 1. 2 Structure d'un muscles squelettique (SEER Training Modules, s. d.)

### 1.2.3 CERVEAU ET MUSCLES

Les muscles dépendent des neurones du cerveau pour recevoir des signaux et lancer des mouvements. Lorsque vous voulez bouger votre doigt, le cerveau envoie des impulsions électriques le long de la moelle épinière aux neurones moteurs situés près des muscles. À la jonction neuromusculaire, l'acétylcholine est libérée, déclenchant une charge électrique qui entraîne la contraction des fibres musculaires. Cela entraîne le mouvement de votre doigt (Colligan, 2010).

Les processus involontaires, comme la digestion, sont également contrôlés par le cerveau par l'intermédiaire des neurones moteurs. Ces processus se produisent automatiquement, indépendamment de la pensée consciente. La connexion complexe entre le cerveau et les muscles permet des mouvements volontaires et involontaires dans le corps. Dans la Figure 1. 3, nous pouvons observer la répartition du système nerveux somatique représentée dans le schéma de contour du cerveau humain.

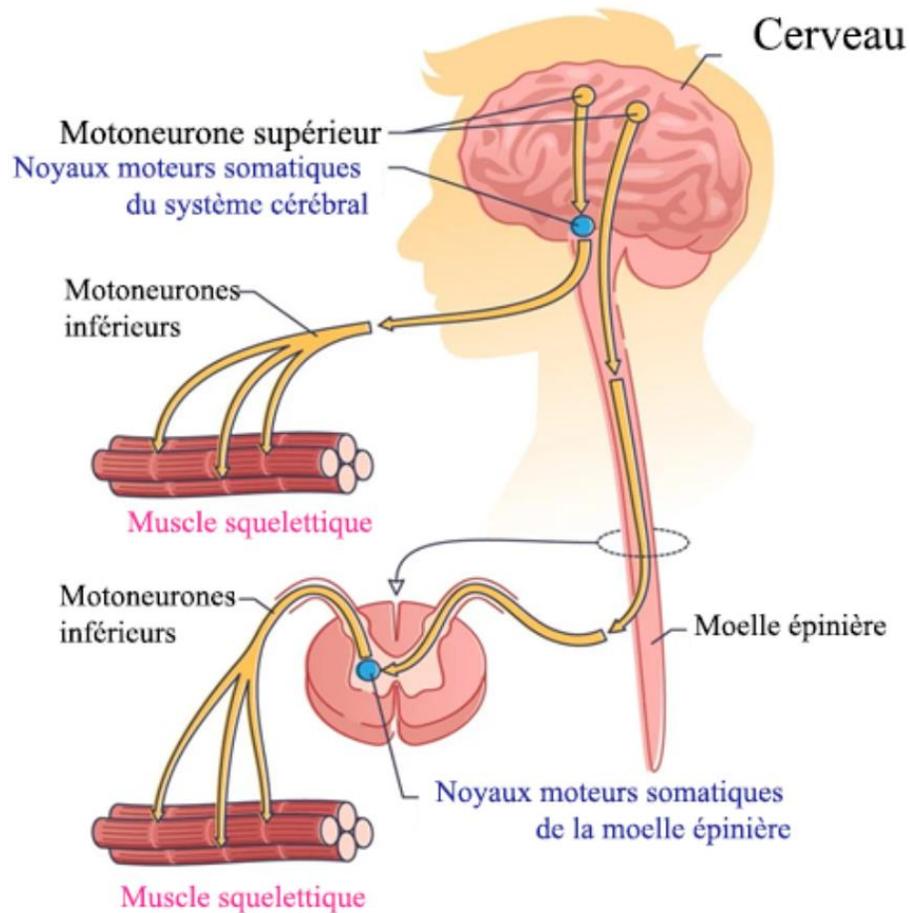


Figure 1. 3 Schéma simplifié de la division du système nerveux somatique en relation avec le cerveau et la moelle épinière humains (Evans, 2023)

#### 1.2.4 MOUVEMENT DES MAINS

Les mouvements complexes de la main résultent d'une interaction complexe entre les muscles extrinsèques et intrinsèques situés dans l'avant-bras et la main. Alors que les muscles extrinsèques de la main trouvent leur origine dans l'avant-bras, les muscles intrinsèques trouvent leur origine (et leur insertion) dans la main elle-même (Andreisek *et al.*, 2017).

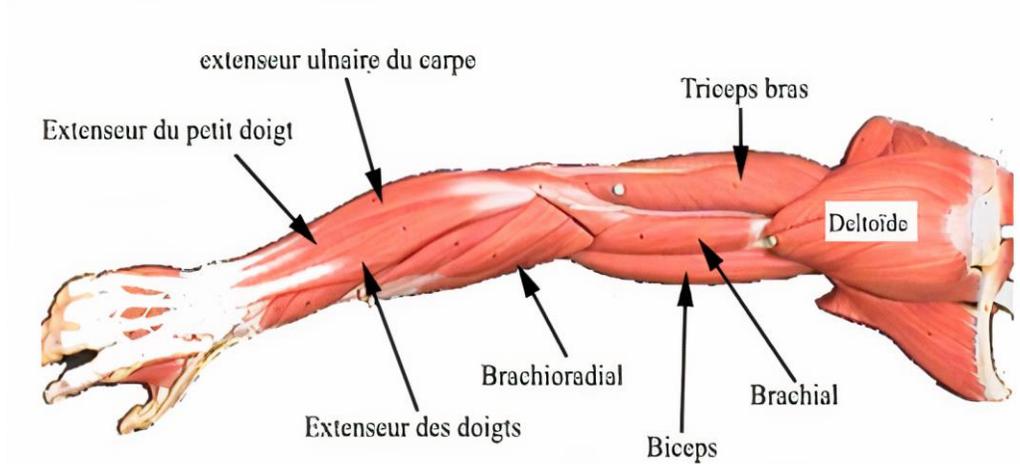


Figure 1. 4 Les muscles de l'avant-bras qui provoquent le mouvement (Alao, 2018)

Dans la Figure 1. 4, nous pouvons observer le schéma illustrant le fonctionnement des muscles de l'avant-bras lorsqu'ils génèrent un mouvement. Cette illustration met en évidence les différents muscles impliqués dans la flexion, l'extension et la rotation de l'avant-bras, ainsi que leur connexion avec les os et les tendons. Cette figure montre également comment ces muscles travaillent ensemble pour produire des mouvements souples et précis de l'avant-bras.

### 1.2.5 SIGNAL ÉLECTROMYOGRAPHIQUE (EMG)

L'électromyographie (EMG) est une technique expérimentale permettant de générer, d'enregistrer et d'analyser des signaux myoélectriques. Les changements physiologiques dans l'état des membranes des fibres musculaires produisent des signaux myoélectriques. En plus des études physiologiques et biomécaniques fondamentales, les signaux EMG se sont imposés comme un outil d'évaluation pour la recherche appliquée, la physiothérapie/rééducation, l'entraînement sportif, ainsi que pour l'étude des interactions du corps humain avec les produits industriels et les environnements de travail (Konard, 2005).

Les signaux électromyographiques (EMG) fournissent des informations sur les trains de potentiels d'action des unités motrices MUAPT (*Motor Unit Action Potential Trains*) générés par les unités motrices MU (*Motor Unit*) engagées pendant la contraction musculaire.

Chaque MUAPT est construit à partir d'une séquence de décharges intermittentes sous la forme de potentiels d'action d'unité motrice MUAP (*Motor Unit Action Potential*) individuels répartis dans l'espace (Ren *et al.*, 2018)

Dans la Figure 1. 5, nous pouvons observer un diagramme illustrant la technique d'enregistrement de l'activité électrique produite par les muscles.

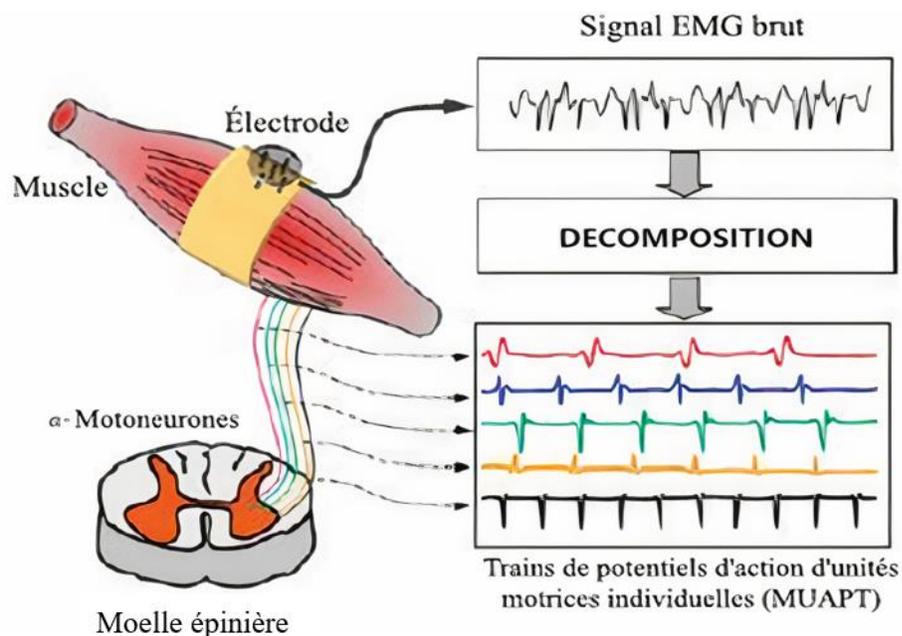


Figure 1. 5 Mesure des signaux électriques musculaires à l'aide de la technologie EMG (W. Lee *et al.*, 2021).

Les motoneurones alpha  $\alpha$  sont des motoneurones relativement grands qui excitent les muscles squelettiques et provoquent les contractions musculaires à l'origine du mouvement (Evans, 2023).

### 1.2.6 UNITE MOTRICE

Les signaux émis par un motoneurone provoquent la contraction des fibres musculaires squelettiques. Chaque fibre musculaire est excitée par un seul motoneurone, mais un seul motoneurone peut exciter de nombreuses fibres musculaires. La Figure 1. 6 représente l'unité

motrice qui est définie comme un seul motoneurone et toutes les fibres musculaires qu'il excite (Biga *et al.*, 2020).

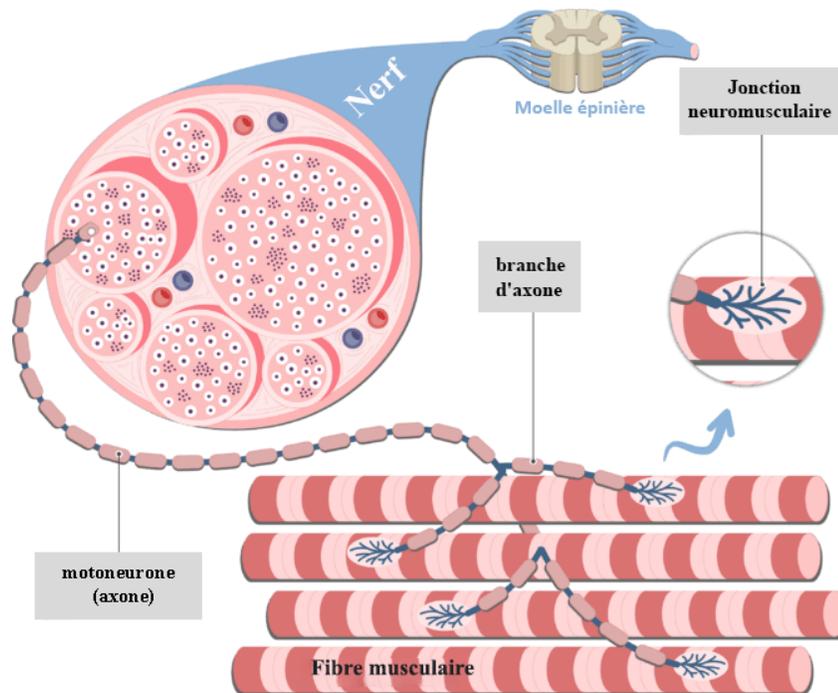


Figure 1. 6 Unité motrice (Sheffield, 2022).

### 1.2.7 SIGNAL ELECTROMYOGRAPHIQUE (EMG) DE LA MAIN

Le rôle de la main est essentiel pour l'exécution des tâches de la vie quotidienne. Son mouvement est régi par un système musculo-squelettique sophistiqué composé d'environ 38 muscles (Jarque-Bou *et al.*, 2021). Par conséquent, le suivi et la compréhension des signaux d'activation musculaire qui contrôlent le mouvement de la main sont essentiels dans de nombreux domaines scientifiques. La contraction des muscles de la main est provoquée par un signal électrique, appelé électromyogramme (EMG), transmis par les motoneurones. L'utilisation d'électrodes permet d'enregistrer ce signal (EMG) (Jarque-Bou *et al.*, 2021). La fréquence des impulsions des motoneurones peut modifier le degré de contraction et le nombre de fibres musculaires qui se contractent (Biga *et al.*, 2020).

### 1.2.8 ACQUISITION DU SIGNAL EMG

Le signal électromyographique (EMG) peut être collecté à l'aide d'un équipement de mesure invasif ou non invasif. Les techniques invasives utilisent des électrodes à fil ou à aiguille qui percent la peau pour atteindre le muscle concerné. L'électromyographie de surface (sEMG), quant à elle, est une approche non invasive qui utilise des électrodes de surface fixées sur la peau (Baygin *et al.*, 2022).

- Acquisition invasive de l'EMG, appelé EMG intramusculaire ou EMG à aiguille : Les électrodes aiguilles permettent de capter en vue d'analyser l'activité du signal émis par la fibre musculaire localement et sans perturbation extérieure. En revanche, elles présentent l'inconvénient d'endommager la peau ainsi que le tissu musculaire, (voir Figure 1. 7).



Figure 1. 7 Électromyographie (EMG) invasive (Brosset, 2018).

- Acquisition non invasive de l'EMG, appelé EMG de surface : Contrairement aux EMG à aiguille qui fournissent le signal émis par une seule fibre, les EMG de surface fournissent un signal émis par un ensemble de fibres musculaires. Les électrodes sont positionnées sur la peau au niveau du muscle concerné, comme le montre la Figure 1. 8. Il existe généralement deux types d'électrodes de surface : les premières sont en contact direct avec la peau et sont appelées électrodes sèches. Quant aux

autres, elles utilisent un gel électrolyte comme interface chimique entre la peau et la partie métallique de l'électrode et sont appelées électrodes à gel.

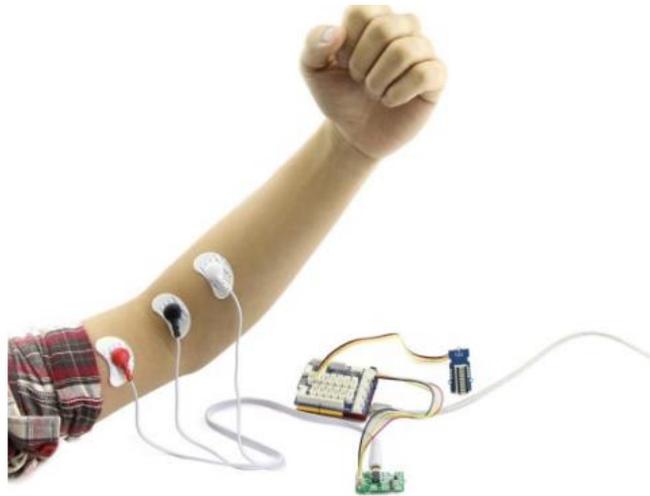


Figure 1. 8 Électromyographie (EMG) de surface (Artal-Sevil *et al.*, 2018).

### 1.2.9 BASES DE DONNEES POPULAIRES

Dans ce qui suit, nous allons décrire les bases de données, accessibles au public, les plus utilisées dans l'évaluation des systèmes de classification à base des signaux EMG de surface :

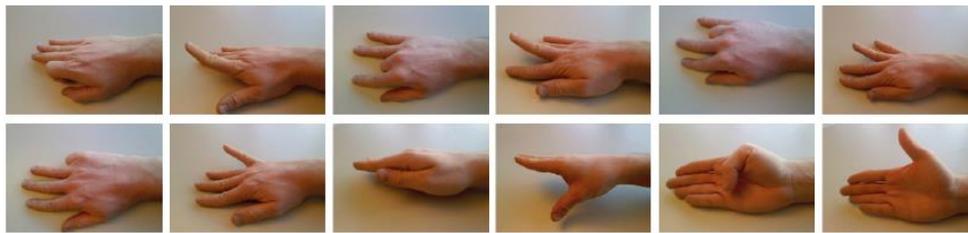
**a. NINAPRO :** La base de données NINAPRO (Atzori *et al.*, 2014) est considérée comme l'une des ressources les plus complètes, comprenant aujourd'hui 10 ensembles de données distincts collectés auprès de 234 sujets sains et 30 sujets amputés. NINAPRO contient un ensemble de données électromyographiques, cinématiques et dynamiques des membres supérieurs pour permettre aux chercheurs de tester des algorithmes d'apprentissage machine pour le contrôle des prothèses de mains. Les 10 ensembles de données comprennent de multiples répétitions de 50 mouvements de la main (voir Tableau 1. 1). Ces derniers ont été enregistrés avec différents protocoles et configurations d'acquisition, afin de permettre l'enregistrement de plusieurs signaux multimodaux, tels

que l'électromyographie de surface, l'accélération, la cinématique ainsi que la force. Malgré le fait que la base de données NINAPRO ne comprenne que les données de 30 amputés de la main, il a été démontré que les données de sujets sains peuvent également être utilisées comme mesure de substitution pour les amputés. Ce résultat justifie l'utilisation de sujets sains afin de réduire le stress et la douleur qui peuvent être causés chez les amputés (Palermo *et al.*, 2017). Le Tableau 1. 1 suivant résume le nombre de sujets et de mouvements ainsi que les capteurs utilisés pour chaque ensemble de données de NINAPRO. La Figure 1. 9 présente les détails des 52 mouvements de la main enregistrés dans la base de données NINAPRO DB1. La base de données est accessible au public par le lien suivant<sup>1</sup>.

Tableau 1. 1 Ensembles de données NINAPRO.

<b>Ensemble de données</b>	<b>Nombre de sujets</b>	<b>Nombre de mouvements</b>	<b>Capteur</b>
<b>DB1</b>	27	52	10 Otto Bock
<b>DB2</b>	40	49	12 Delsys Trigno
<b>DB3</b>	11 amputés	49	12 Delsys Trigno
<b>DB4</b>	10	52	12 Cometa
<b>DB5</b>	10	52	2 Thalmic Myo Armbands (16 électrodes)
<b>DB6</b>	10	7	14 Delsys Trigno
<b>DB7</b>	20 intacts, 2 amputés	6	12 Delsys Trigno
<b>DB8</b>	10 intacts, 2 amputés	9	16 Delsys Trigno
<b>DB9</b>	77	40	22 sensors Cyberglove-II
<b>DB10</b>	30 intacts, 15 amputés	10	12 Delsys Trigno

<sup>1</sup> <https://ninapro.hevs.ch>



(a) Mouvements de base des doigts (flexions et extensions).



(b) Configurations isométriques et isotoniques des mains ("postures des mains").



(c) Mouvements de base du poignet.



(d) La préhension et les mouvements fonctionnels.

Figure 1. 9 Détails des 52 mouvements de la main de NINAPRO DB1 (Atzori *et al.*, 2012).

**b. CSL-HDEMG:** détaillée dans (Amma *et al.*, 2015), elle présente des similitudes avec la base de données CapgMyo, car elle comprend également des signaux électromyographiques (EMG) à haute résolution, connus sous le nom de HD-sEMG. La base de données CSL-HDEMG contient principalement 27 gestes des doigts de la main effectués par cinq sujets différents. Ces gestes englobent divers mouvements d'extension et de flexion des doigts, ainsi que

d'autres gestes couramment utilisés dans les interactions homme-machine (voir Figure 1. 10). Pour capturer les signaux, une grille d'électrodes de 7 x 24 a été utilisée, ce qui a permis d'obtenir un total de 168 canaux de données. Les signaux HDEMG ont été échantillonnés à une fréquence de 2048 Hz. Les données de la base de données CSL-HDEMG ont été classées en trois catégories : les gestes de frappe, les gestes de flexion et les gestes multi-doigts. Chaque ensemble représente des types spécifiques de mouvements effectués pendant les enregistrements. La base de données est accessible sur le lien suivant<sup>2</sup>.

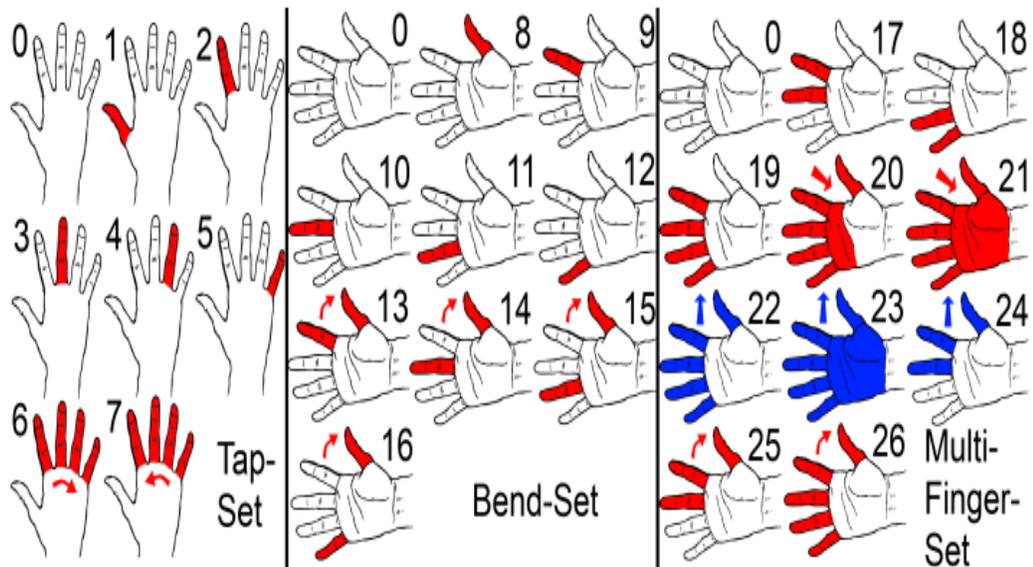


Figure 1. 10 Illustrations iconiques des 27 gestes effectués contenus dans la base de données SCL-HDEMG (Amma *et al.*, 2015).

- c. **Ensemble de données EMG pour les gestes de la main (UCI) :** Cet ensemble de données a été initialement présenté par (Lobov *et al.*, 2018) dans " *The UCI Machine Learning Repository* " <sup>3</sup>. Il comprend des signaux myographiques à 8

<sup>2</sup> <https://www.csl.uni-bremen.de/CorpusData/download.php?crps=cslhdemg>

<sup>3</sup> <https://archive.ics.uci.edu/dataset/481/emg+data+for+gestures>

canaux enregistrés par des bracelets Myo Gesture Armband portés sur les avant-bras de 36 sujets. Chaque sujet a effectué le même ensemble de six gestes deux fois de manière séquentielle. Avec un septième geste, mais il n'a pas été effectué par tous les sujets. Chaque geste a été effectué pendant 3 secondes avec une pause de 3 secondes entre les gestes.

- d. **CapgMyo** : décrit dans (Du *et al.*, 2017), cette base de données comprend des signaux électromyographiques (EMG) à haute résolution connus sous le nom de HD-sEMG. Les signaux ont été enregistrés chez 23 personnes en bonne santé et capturent l'activité électrique musculaire d'une zone spécifique de la peau. L'acquisition des données a nécessité huit modules, chacun équipé d'un réseau d'électrodes bidimensionnelles disposées dans une matrice de 8 x 16. L'échantillonnage des signaux EMG s'est fait à une fréquence de 1000 Hz. Cette technique a permis l'analyse et la modélisation des signaux EMG dans les domaines temporel et spatial, facilitant ainsi la recherche sur les futures interfaces muscle-ordinateur (MCI). Pour organiser la base de données CapgMyo, elle a été divisée en trois ensembles : DB-a, DB-b et DB-c, en fonction de la procédure d'acquisition. Les ensembles DB-a et DB-b englobent des gestes isométriques et isotoniques identiques de la main. En revanche, le DB-c se concentre sur 12 mouvements fondamentaux des doigts. Il est important de noter que les mouvements de chaque sous-ensemble de la base de données CapgMyo s'alignent sur ceux présents dans la base de données NINAPRO (voir Figure 1. 9). La base de données est accessible au public sur le lien suivant<sup>4</sup>.

---

<sup>4</sup> [http://zju-capg.org/research en electro capgmyo.html#download](http://zju-capg.org/research%20en%20electro%20capgmyo.html#download)

### 1.3 PROBLEMATIQUE DE RECHERCHE

La reconnaissance automatique des gestes de la main basée sur les signaux EMG est l'une des techniques permettant aux personnes à mobilité réduite notamment les personnes amputées des membres de contrôler et manipuler de façon efficace les prothèses et d'interagir avec les interfaces homme/machines. Cette approche, utilisant l'apprentissage machine, présente plusieurs défis qui doivent être relevés afin d'obtenir des résultats précis et fiables. L'un des principaux défis consiste à obtenir des signaux EMG de haute qualité qui seront idéalement acquis dans la population concernée. Cela peut être difficile en raison de facteurs tels que la variabilité des schémas d'activation musculaire (Benatti *et al.*, 2015), les différences d'impédance de la peau (Saelim *et al.*, 2019) et les effets de mouvement dans les signaux artéfacts. Un autre défi consiste à choisir les algorithmes d'apprentissage machine et les caractéristiques appropriés pour la reconnaissance, car différents algorithmes peuvent être plus performants pour différents types de gestes ou pour différentes populations d'utilisateurs. La taille et la diversité de l'ensemble de données d'apprentissage jouent également un rôle essentiel dans la détermination de l'exactitude du système de reconnaissance. En outre, l'exactitude de reconnaissance du système peut être affectée par des facteurs tels que la longueur de la fenêtre temporelle dans l'étape de prétraitement des données et le choix de la méthode d'extraction des caractéristiques. Par conséquent, un examen attentif de ces aspects est essentiel pour parvenir à une reconnaissance des gestes de la main basée sur l'EMG et utilisant l'apprentissage machine.

De nombreuses équipes de recherche travaillent actuellement sur des projets visant à développer et à réaliser des systèmes de reconnaissance des gestes de la main robuste et fiable. Il existe aujourd'hui un certain nombre d'ensembles de données accessibles au public, tels que NINAPRO (Atzori *et al.*, 2012) et UCI (Lobov *et al.*, 2018), qui utilisent différentes techniques d'acquisition et différents capteurs. Dans (Tepe & Demir, 2022), les performances de la classification par machine à vecteurs de support (SVM) pour les signaux EMG en temps réel et non réel ont été étudiées. De plus, Les auteurs dans (Zhang *et al.*, 2023) ont proposé une méthode de reconnaissance en ligne des gestes de la main utilisant des données

d'électromyographie de surface (sEMG). Cette méthode utilise un extracteur de caractéristiques basé sur le réseau convolutionnel temporel, qui est conçu pour apprendre des caractéristiques discriminantes de haut niveau à partir des signaux d'entrée. En outre, d'autres études ont mis en œuvre divers algorithmes d'apprentissage machine tels que les k-plus proches voisins k-NN (*k-Nearest Neighbors*) (Kim *et al.*, 2008; Shi *et al.*, 2018), les réseaux neuronaux artificiels ANN (*Artificial Neural Networks*) (Lee *et al.*, 2021) et les réseaux neuronaux convolutionnels CNN (*Convolutional Neural Networks*) (Ozdemir *et al.*, 2020) en tant qu'algorithmes de classification, atteignant une précision d'au moins 90 % dans la classification de quatre à dix gestes de la main.

#### **1.4 OBJECTIFS**

Le présent projet de recherche vise principalement à concevoir et développer un système robuste et fiable de classification des gestes de la main basé sur les signaux EMG en utilisant des algorithmes d'apprentissage machine et à évaluer ses performances. Le système sera testé à l'aide d'un ensemble de données de gestes de la main recueillies auprès d'un groupe diversifié de sujets, et les performances seront mesurées à l'aide de différents critères d'évaluation. En plus, des tests seront effectués dans des conditions réelles pour valider l'efficacité du système en environnement pratique. La recherche vise également à comparer les performances de différents algorithmes d'apprentissage machine avec celles rapportées dans d'autres études similaires dans la littérature.

#### **1.5 HYPOTHESE**

La captation de plusieurs signaux EMG provenant de certaines régions (muscles) de l'avant-bras permet de recueillir suffisamment d'information pour discriminer différents gestes de la main, offrant ainsi une solution technologique innovante pour améliorer la qualité de vie des amputés. Ce projet de recherche émet l'hypothèse que l'utilisation des signaux EMG enregistrés à partir des muscles de l'avant-bras permettra une reconnaissance précise

et efficace des gestes de la main. Nous anticipons que les formes distinctives présents dans les signaux EMG, correspondant à différents mouvements de la main, peuvent être identifiés et utilisés afin de contribuer de manière significative à la mise au point de prothèses plus intuitives. Cela permettrait aux amputés, par exemple, de contrôler efficacement des gestes variés et ainsi favoriser leur réintégration dans les activités quotidiennes.

## 1.6 METHODOLOGIE

Pour atteindre ces objectifs, notre approche est structurée en plusieurs étapes essentielles, illustrées dans la Figure 1. 11 :

- a) **Acquisition des signaux EMG:** La première étape consiste à collecter des signaux EMG de bonne qualité auprès d'un groupe diversifié de sujets. Il existe plusieurs capteurs pour l'acquisition du signal EMG, on peut citer *OttoBock MyoBock electrodes*, *Delsys Trigno sensors*, *Cometa Wave Plus*, et le brassard *Thalmic Myo Gesture Armband*. Nous avons utilisé le brassard Myo Gesture Armband pour l'acquisition car c'est un capteur d'électromyographie (EMG) portable sans fil conçu pour acquérir l'activité musculaire de l'avant-bras. Le Myo Gesture Armband dispose de 8 canaux de capteurs, offrant ainsi des signaux EMG de bonne qualité. Il a été utilisé dans diverses applications, en particulier dans les domaines de l'interaction homme-machine et de la reconnaissance des gestes. Les données collectées doivent être fiables, cohérentes et représentatives de la population concernée.
- b) **Prétraitement des signaux EMG:** Cette étape implique le prétraitement des signaux EMG collectés en utilisant la segmentation (fenêtrage).
- c) **Extraction des caractéristiques:** Les signaux prétraités sont ensuite utilisés pour extraire les caractéristiques appropriées, sélectionnées en fonction de leur valeur pour la tâche de reconnaissance des gestes de la main.

**d) Classification:** Les modèles d'apprentissage machine pour la reconnaissance des gestes de la main sont sélectionnés et entraînés à l'aide de l'ensemble de données traitées. Les performances des systèmes développés sont évaluées à l'aide de différentes mesures d'évaluation, telles que l'exactitude.

**e) Application dans des conditions réelles:** La dernière étape consiste à tester et à valider le système de reconnaissance de gestes de la main dans les conditions réelle. Le système de reconnaissance des gestes de la main peut être intégré à d'autres systèmes, tels que la robotique ou les prothèses, selon les besoins.

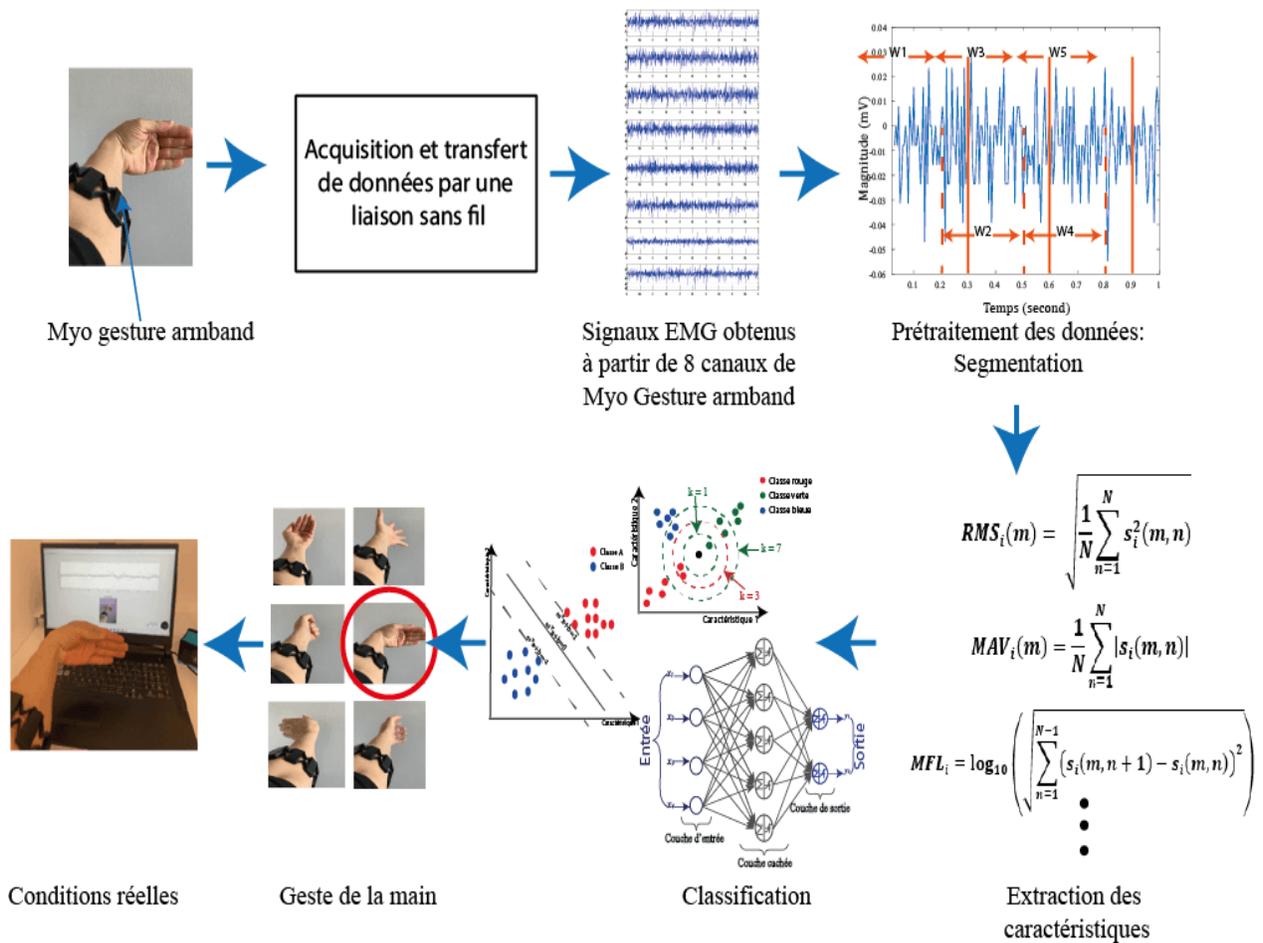


Figure 1. 11 Processus de la reconnaissance des gestes de la main.

## 1.7 CONTRIBUTION

Dans cette recherche, nous avons apporté des contributions significatives à la reconnaissance des gestes de la main basée sur les signaux électromyographie (EMG) en étudiant plusieurs aspects clés. Tout d'abord, des expériences ont été menées pour optimiser la longueur de la fenêtre et le chevauchement, ce qui a permis d'améliorer la reconnaissance des gestes. Ensuite, plusieurs techniques d'extraction de caractéristiques dans le domaine temporel ont été testées. De plus, une approche a été conçue pour combiner trois techniques d'extraction de caractéristiques les plus performantes. En outre, des algorithmes de classification par apprentissage machine ont été appliqués, et le modèle le plus performant a été sélectionné et appliqué dans des conditions réelles. Au final, nous avons comparé nos résultats avec des études qui ont utilisé le brassard Myo Gesture Armband pour l'acquisition des données. Cette approche a également été testée sur un autre ensemble de données populaire et les résultats obtenus ont été comparés à des études qui utilisaient également cet ensemble de données. Ce qui a permis une évaluation plus précise de l'exactitude de classification automatique des gestes de la main par le modèle proposé.



## CHAPITRE 2

### PRÉTRAITEMENT DU SIGNAL EMG ET EXTRACTION DES CARACTÉRISTIQUES

Dans ce chapitre, nous allons étudier les processus essentiels de prétraitement du signal électromyographique (EMG) et d'extraction des caractéristiques. Le prétraitement consiste à améliorer les données EMG brutes, en utilisant des techniques telles que le filtrage pour rendre le signal plus clair. Ensuite, l'extraction des caractéristiques permet de récupérer les éléments significatifs des signaux traités.

#### 2.1 PRETRAITEMENT

Le prétraitement du signal EMG implique une série de techniques appliquées aux données brutes afin d'en améliorer la qualité et l'interprétabilité en vue d'une analyse ultérieure. Ce processus comprend généralement le filtrage du bruit, la segmentation et le fenêtrage. L'objectif principal du filtrage est d'éliminer les informations inutiles afin de rendre les signaux plus adaptés à une analyse précise dans des applications telles que la reconnaissance des gestes de la main.

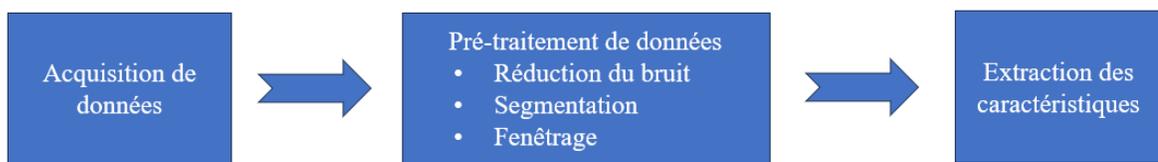


Figure 2. 1 Représentation des étapes de traitement du signal : acquisition des données, pré-traitement des données et extraction des caractéristiques.

La Figure 2. 1 montre les étapes du traitement du signal. La première étape est l'acquisition des données, où les données brutes sont recueillies. La deuxième étape est le pré-traitement des données, qui inclut la réduction du bruit, la segmentation et le fenêtrage. La dernière étape est l'extraction des caractéristiques, où les informations pertinentes sont extraites des données.

### 2.1.1 REDUCTION DU BRUIT

Dans le domaine du traitement des signaux biomédicaux, le débruitage des signaux EMG (électromyographie) joue un rôle important dans la qualité résultats de leur analyse. Les signaux EMG sont souvent perturbés par diverses sources de bruit, notamment les artefacts de mouvement, les interférences des lignes électriques et le bruit des électrodes. Il est essentiel de traiter ces facteurs de bruit pour garantir une interprétation et une analyse précises et fiables des données EMG. Diverses techniques ont été proposées et étudiées dans la littérature pour débruiter les signaux EMG. L'une de ces techniques est la transformée en ondelettes discrète DWT (*Discrete Wavelet Transform*), qui utilise des fonctions à base d'ondelettes pour décomposer le signal en différentes bandes de fréquence. En appliquant un seuillage aux coefficients d'ondelettes et en reconstruisant le signal, il est possible de réduire le bruit tout en préservant les caractéristiques importantes du signal (Bhobhriya *et al.*, 2023). Une autre méthode de débruitage efficace est l'approche des moyennes non locales NLM (*Non-Local Means*), qui exploite les similitudes entre les parcelles du signal pour estimer le signal propre. Cette technique permet de préserver la structure du signal tout en supprimant le bruit (Bhobhriya *et al.*, 2023). En outre, la décomposition variationnelle des modes VMD (*Variational Mode Decomposition*) s'est révélée prometteuse pour le débruitage des signaux EMG. Elle décompose le signal en modes à bande étroite et le reconstruit en utilisant les modes sélectionnés pour séparer les composantes de bruit. L'approche basée sur la VMD a démontré une amélioration des performances de débruitage par rapport aux méthodes traditionnelles (Ma *et al.*, 2020).

### 2.1.2 SEGMENTATION DES SIGNAUX

Dans cette étape, les signaux EMG sont divisés en tranches (segments) temporels afin d'extraire des caractéristiques pour chaque segment. De nombreuses méthodes, dont les techniques de segmentation adjacent (Wang *et al.*, 2013) et de chevauchement (Jia *et al.*, 2020), sont utilisées pour diviser les signaux EMG. Chaque nouveau segment dans la segmentation adjacent commence là où se termine le segment précédent. Dans la

segmentation par chevauchement, un nouveau segment commence avant la fin du segment précédente, permettant d’avoir des fenêtres avec un certain taux de chevauchement. De nombreuses études (Colli & Trejos, 2022; Hassan *et al.*, 2020; Murciego *et al.*, 2022) ont choisi un chevauchement de 50%. La Figure 2. 2 ci-dessous montre la différence entre les deux techniques utilisées dans la segmentation.

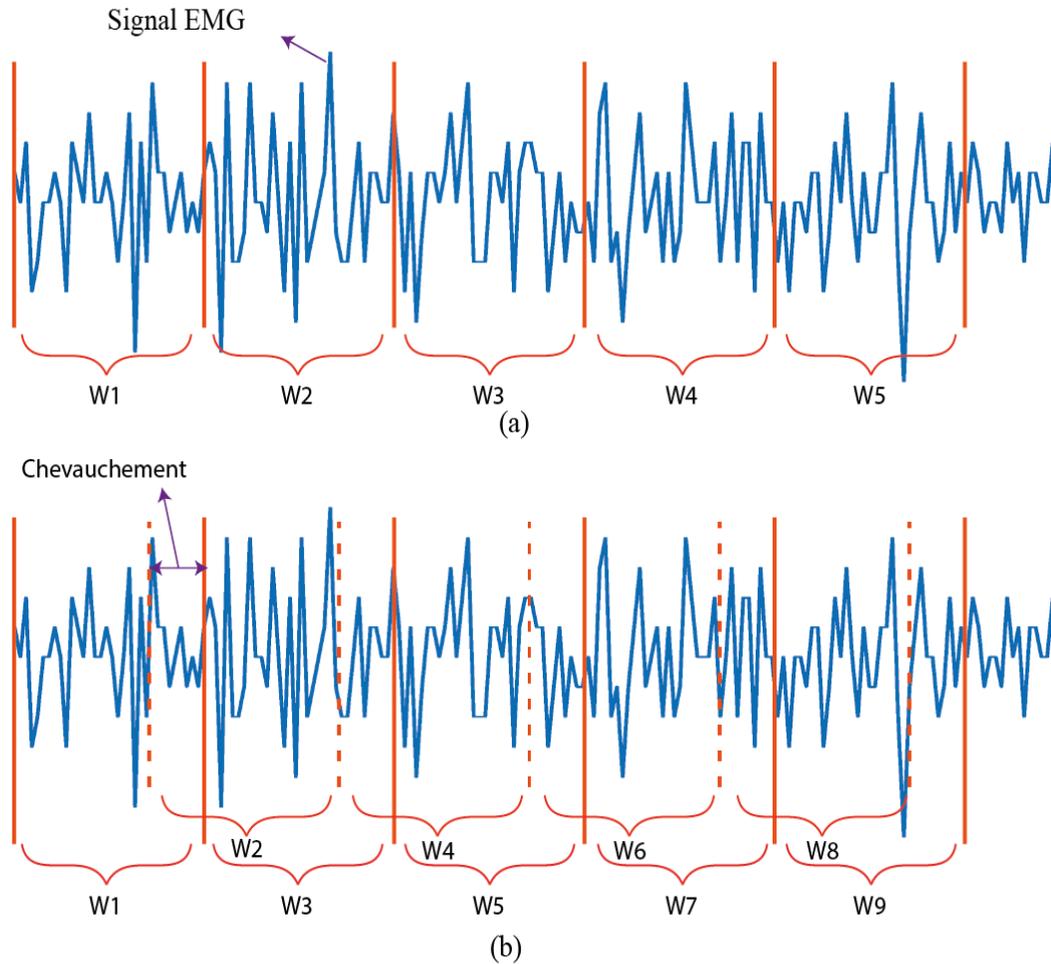


Figure 2. 2 Segmentation du signal par: (a) sans chevauchement; (b) avec chevauchement

Le signal EMG brute  $s_i(n)$  du  $i^{\text{ème}}$  canal est divisé en segments plus courts obtenus en le multipliant par une fenêtre rectangulaire glissante  $w(n)$  :

$$s_i(m, n) = s_i(n)w(n - mL) \quad (2. 1)$$

où  $m$  représente l'indice de la fenêtre,  $L$  représente le pas d'avancement et  $w(n)$  est la fenêtre rectangulaire de taille  $N$ . Le taux de chevauchement est défini par  $(N - L) / N$ .

La longueur de la fenêtre des signaux EMG est déterminante pour la performance de la classification et le temps nécessaire à l'analyse des signaux EMG en mode en ligne. Comme les informations ne sont pas suffisamment représentatives de la forme à classer lorsque la longueur de la fenêtre est trop petite, l'exactitude de la classification diminue. Pour cette raison, il est difficile de récupérer les informations pertinentes de chaque fenêtre. En revanche, l'augmentation de la longueur de la fenêtre entraîne d'augmenter progressivement le temps de latence ainsi que l'exactitude de la classification jusqu'à un certain point, mais au-delà, il n'y a pas de gain de précision du système. Par conséquent, un compromis entre l'exactitude et le temps de retard doit être ajusté pour garantir une haute exactitude et un faible temps de retard (Hassan *et al.*, 2020). Une longueur de fenêtre comprise entre 125 ms et 300 ms a été généralement considérée dans la littérature (Tepe & Demir, 2022). Dans (Hassan *et al.*, 2020; Tepe & Demir, 2021), les auteurs ont choisi une longueur de fenêtre de 240 ms.

### 2.1.3 FENETRAGE

Le processus de fenêtrage dans le traitement des signaux consiste à appliquer une fonction fenêtre glissante au signal afin de limiter la taille de ses bords. Ce processus permet les artefacts lors de l'analyse spectrales du signal. Les fenêtres de *Hanning*, *Hamming* et *Blackman* sont bien connues et largement utilisées à cette fin (Chithra & Aparna, 2015). Le choix du type et de la taille de la fenêtre peut affecter la résolution en temps et en fréquence lors de l'analyse du signal. Par exemple, la fenêtre de *Hamming* est préférée par beaucoup en raison de la largeur relativement étroite de son lobe principal et de la bonne suppression des lobes secondaires. Le signal fenêtré peut ensuite être traité pour diverses applications, telles que la détection d'enveloppe et l'extraction de caractéristiques (Venugopal *et al.*, 2014). Le fenêtrage peut être combiné à la segmentation en remplaçant la fenêtre rectangulaire par une autre fenêtre de pondération (*Hanning*, *Hamming* ou *Blackman*).

## 2.2 EXTRACTION DES CARACTERISTIQUES

L'extraction des caractéristiques est une étape permettant de récupérer des informations importantes de chaque segment du signal EMG (Too *et al.*, 2019). L'extraction de caractéristiques peut être réalisée dans le domaine temporel (TD), le domaine fréquentiel (FD) ou le domaine temps-fréquence (TFD). Les caractéristiques EMG les plus largement utilisées sont celle extraites dans le domaine temporel (TD).

### 2.2.1 CARACTERISTIQUES DU DOMAINE TEMPOREL

Dans le domaine temporel, les données brutes sont analysées directement dans la séquence temporelle. Quelques fonctions courantes d'extraction de caractéristiques TD sont décrites ci-dessous :

- **RMS** : la racine carrée moyenne RMS (*Root Mean Square*). Dans le traitement des signaux EMG, le RMS est l'une des caractéristiques du domaine temporel les plus souvent utilisées dans le traitement des signaux EMG (Karabulut *et al.*, 2017). La formulation mathématique du RMS est définie comme suit :

$$RMS_i(m) = \sqrt{\frac{1}{N} \sum_{n=1}^N s_i^2(m, n)} \quad (2.2)$$

où  $i$  désigne le canal et  $m$  est l'indice de la fenêtre ou du segment analysé de taille  $N$ .

- **MAV** : la valeur absolue moyenne MAV (*Mean Absolut Value*) est définie comme suit (Phinyomark *et al.*, 2011):

$$MAV_i(m) = \frac{1}{N} \sum_{n=1}^N |s_i(m, n)| \quad (2.3)$$

- **WL** : La longueur de la forme d'onde WL (*Wave Length*) désigne le degré de complexité de chaque segment du signal EMG (Karabulut *et al.*, 2017). Son expression mathématique est la suivante :

$$WL_i(m) = \sum_{n=1}^{N-1} |s_i(m, n+1) - s_i(m, n)| \quad (2.4)$$

- **MFL** : La longueur fractale maximale MFL (*Maximum Fractal Length*) est une composante EMG récente utilisée pour quantifier l'activation de la contraction musculaire de bas niveau (Too *et al.*, 2019). Mathématiquement, le MFL est défini par:

$$MFL_i = \log_{10} \left( \sqrt{\sum_{n=1}^{N-1} (s_i(m, n+1) - s_i(m, n))^2} \right) \quad (2.5)$$

- **SSC** : Le changement de signe de la pente SSC (*Slope Sign Changes*) possède des propriétés liées à la fréquence qui permettent de compter les changements du signe de la pente du signal sEMG (Hassan *et al.*, 2020). L'expression mathématique du SSC est la suivante :

$$SSC_i(m) = \sum_{n=2}^{N-1} f \left( (s_i(m, n) - s_i(m, n-1))(s_i(m, n) - s_i(m, n+1)) \right) \quad (2.6)$$

$$f(x) = \begin{cases} 1 & \text{si } x \geq \text{seuil} \\ 0 & \text{sinon} \end{cases} \quad (2.7)$$

- **ZC** : la fonction passages par zéro ZC (*Zéro Crossings*) contient des caractéristiques liées à la fréquence qui comptent le nombre de fois où l'amplitude d'un signal traverse l'amplitude zéro sur un segment de temps donné. Elle affiche la quantité de fluctuations du signe du signal tout en mesurant le décalage de fréquence (Hassan *et al.*, 2020). Le ZC est représenté mathématiquement comme suit :

$$ZC_i(m) = \sum_{n=1}^{N-1} \text{sgn}(s_i(m, n)s_i(m, n + 1)) \cap |s_i(m, n) - s_i(m, n + 1)| \geq \alpha \quad (2.8)$$

où  $\alpha$  est un seuil, avec

$$\text{sgn}(x) = \begin{cases} 1 & \text{si } x \geq \alpha \\ 0 & \text{sinon} \end{cases} \quad (2.9)$$

- **IEMG** : Le terme EMG intégré IEMG (*Integrated Electromyography*) fait référence à la moyenne des valeurs d'amplitude du signal EMG pour chaque segment du signal (Karabulut *et al.*, 2017). C'est ainsi que l'IEMG est décrit :

$$IEMG_i(m) = \sum_{n=1}^N |s_i(m, n)| \quad (2.10)$$

- **DASDV** : Différence Écart-type absolu Valeur DASDV (*Difference Absolute Standard Deviation Value*). Elle est représentée mathématiquement comme suit (Tepe & Demir, 2021):

$$DASDV_i(m) = \sqrt{\frac{1}{N-1} \sum_{n=1}^{N-1} (s_i(m, n+1) - s_i(m, n))^2} \quad (2.11)$$

- **MMAV**: La valeur absolue moyenne modifiée MMAV (*Modified Mean Absolute Value*) est une version améliorée de la valeur absolue moyenne (MAV), qui intègre une fonction de fenêtre pondérée. La formule mathématique de la MMAV peut être exprimé comme suit (Too, Abdullah, & Saad, 2019):

$$MMAV_i(m) = \frac{1}{N} \sum_{n=1}^N f_n |s_i(m, n)| \quad (2.12)$$

$$f_n = \begin{cases} 1, & \text{si } 0.25N \leq n \leq 0.75N \\ 0.5, & \text{sinon} \end{cases}$$

- **MMAV2:** La valeur absolue moyenne modifiée 2 MMAV2(*Modified Mean Absolute Value type 2*) est une extension supplémentaire de la caractéristique de la valeur absolue moyenne (MAV), qui utilise une fonction de fenêtre de pondération continue. Le calcul de MMAV2 peut être exprimé mathématiquement comme suit :

$$MMAV2_i(m) = \frac{1}{N} \sum_{n=1}^N f_n |s_i(m, n)|$$

$$f_n = \begin{cases} 1, & \text{si } 0.25N \leq n \leq 0.75N \\ \frac{4n}{N}, & \text{si } n < 0.25N \\ \frac{4(n-N)}{N}, & \text{sinon} \end{cases} \quad (2.13)$$

où  $m$  représente l'indice du segment.  $N$  est la longueur du segment et  $i$  représente le numéro du canal,  $i = 1, \dots, 8$ . Et  $s_i(m, n)$  est segment analysé.

### 2.2.2 CARACTERISTIQUES DU DOMAINE FREQUENTIEL

Dans le domaine des fréquences, les données brutes sont converties en leur représentation fréquentielle par des méthodes telles que la transformée de Fourier, ce qui permet d'extraire les caractéristiques fréquentielles du signal. Diverses techniques d'extraction de caractéristiques sont employées dans ce domaine, parmi lesquelles on trouve celles obtenue à partir de la densité spectrale de puissance PSD (Power Spectral Density).

La PSD illustre la distribution de la puissance d'un signal sur différentes composantes de fréquence, ce qui donne une idée de l'énergie présente dans ces éléments de fréquence. Pour calculer la densité spectrale de puissance d'un signal, il faut d'abord effectuer sa transformée de Fourier (Bahoura & Simard, 2012). La transformée de Fourier est une technique mathématique utilisée pour analyser les signaux et leurs composantes de fréquence. Appliquée à un signal à temps discret  $s_i(n)$ , la transformée de Fourier à court terme STFT (*Short-Time Fourier Transform*) est calculée en effectuant une transformée de Fourier sur des images continues du signal.

$$S_i(m, k) = \sum_{n=0}^{N-1} s_i(n) \omega(n - mL) e^{\frac{-j2\pi nk}{N}} \quad (2.14)$$

où  $\omega(n)$  est une fonction de fenêtrage de pondération de  $N$  échantillons, centrée sur la position temporelle  $m$  et  $N$  représente aussi le nombre de fréquences discrètes,  $L$  est le pas d'avancement,  $k$  est l'indice des fréquences et  $i$  représente le numéro du canal,  $i = 1, \dots, 8$ . La densité spectrale de puissance  $P_i(m, k)$  peut être calculée en utilisant :

$$P_i(m, k) = \frac{1}{N} [S_i(m, k)]^2 \quad (2.15)$$

Certaines caractéristiques du domaine fréquentiel sont données ci-dessous avec leur expression mathématique (Tepe & Demir, 2021) :

- **Fréquence de pic**

$$PKF_i = \max(P_{i,j}) \quad (2.16)$$

- **Fréquence Moyenne**

$$MNF_i = \frac{\sum_{j=1}^N P_{i,j} f_j}{\sum_{j=1}^N P_{i,j}} \quad (2.17)$$

- **Puissance moyenne**

$$MNP_i = \frac{1}{N} \sum_{j=1}^N P_{i,j} \quad (2.18)$$

- **Moment spectral**

$$SM_i = \sum_{j=1}^N P_{i,j} f_j \quad (2.19)$$

où  $P_{i,j}$  représente la  $j^{\text{ème}}$  densité spectrale de puissance et  $f_j$  est la fréquence dans une fenêtre.  $N$  est le nombre des fréquence discrètes.

## 2.3 APPROCHE PROPOSÉ

Le processus d'extraction des caractéristiques est détaillé et illustré à la Figure 2. 3.

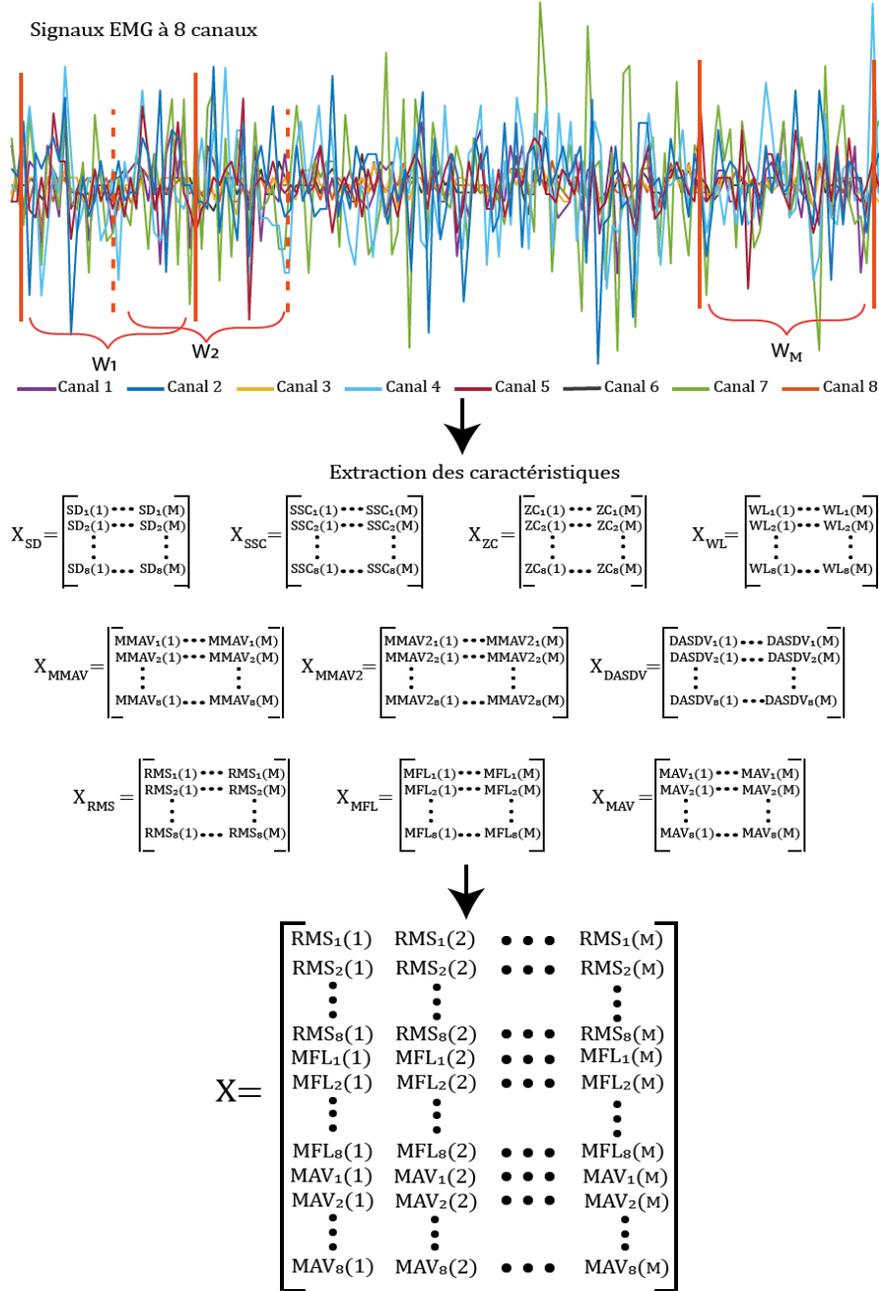


Figure 2. 3 Illustration du processus d'extraction des caractéristiques du signal EMG.

Dans ce projet de recherche, nous allons tester plusieurs techniques d'extraction de caractéristiques dans le domaine temporel afin de combiner trois parmi elles qui fournissent la meilleure exactitude de classification. Il n'y a pas de limitations de nombre de caractéristiques, nous aurions pu choisir plus de 3 caractéristiques, mais cela entraînerait une augmentation de la complexité du système en termes de temps de calcul. La procédure d'extraction des caractéristiques commence par la segmentation temporelle du signal EMG en fenêtres ou segments. À partir de chaque segment, nous extrayons la matrice des caractéristiques spécifique comme illustré dans la Figure 2. 3, par exemple la matrice  $X_{RMS}$  contient les caractéristiques *RMS*, puis nous la testons pour déterminer le niveau d'exactitude qu'elle apporte. Enfin, les caractéristiques les plus performantes sont combinées afin d'améliorer l'exactitude globale de notre système de classification, comme illustré dans la figure ci-dessus, la matrice  $X$  combine les caractéristiques *RMS*, *MFL* et *MAV*.



### **CHAPITRE 3**

## **ALGORITHMES D'APPRENTISSAGE MACHINE POUR LES TACHES DE CLASSIFICATION AUTOMATIQUE**

Après avoir extrait les caractéristiques importantes du signal EMG, nous passerons à l'étape de la classification automatique des gestes de la main en utilisant certains algorithmes d'apprentissage machine. La classification automatique adoptée se base sur l'approche par apprentissage supervisé dans lequel un système est formé pour prédire la classe ou la catégorie d'un point de données d'entrée où sa classe d'appartenance est connue.

L'apprentissage machine est une stratégie qui améliore les performances d'un système en utilisant des algorithmes qui tirent des enseignements de l'expérience. Les données sont la principale source d'expérience dans les systèmes informatiques, et l'objectif fondamental de l'apprentissage machine est de développer des algorithmes d'apprentissage qui créent des modèles à partir des données. En fournissant des données d'expérience à l'algorithme d'apprentissage, nous sommes en mesure de créer un modèle qui peut anticiper les résultats des observations futures (Zhou, 2021c).

Les modèles d'apprentissage profond (*deep learning*), comme les réseaux de neurones convolutifs (CNN), ont besoin de beaucoup de données et d'un bon équilibre entre les classes de données pour atteindre un taux de classification élevé. En comparaison, les modèles d'apprentissage automatique traditionnels (*machine learning*), comme les réseaux MLP, sont un peu moins exigeants à cet égard. Les réseaux CNN sont très performants avec des images, mais quand il s'agit des signaux monodimensionnels, comme les signaux audio ou les signaux biomédicaux, qui nécessitent généralement une transformation en images (spectrogrammes), ces réseaux ne donnent pas toujours les meilleures performances. Donc, bien que les deux approches puissent être utilisées pour la classification, chacune a ses forces et ses faiblesses selon le type de données (Semmad & Bahoura &, 2024).

### 3.1 MACHINE A VECTEURS DE SUPPORT (SVM)

La machine à vecteurs de support SVM (*Support Vector Machine*) est l'un des algorithmes d'apprentissage automatique les plus populaires dans les tâches de reconnaissance des formes (classification). L'algorithme utilise une variété de noyaux pour traiter les fonctions linéaires et non linéaires (Noble, 2006). L'objectif initial de l'algorithme SVM était de résoudre les problèmes de classification binaire (Ukil, 2007). Cet algorithme considère chaque vecteur de caractéristique comme un point dans un espace à  $n$  dimensions et classe les données en deux classes en utilisant un hyperplan. L'idée principale du système SVM est de trouver cet hyperplan qui sépare deux classes de telle sorte que la distance entre les points de données les plus proches de chaque classe soit maximale.

#### 3.1.1 CLASSIFICATION LINEAIRE

La problématique de classification linéaire est de créer des frontières de décision linéaires qui permettent de manière consciente de catégoriser les données en différents groupes. L'hyperplan de séparation dans un espace bidimensionnel serait une ligne droite qui est généralement spécifiée par le vecteur de poids  $\omega$  et le vecteur de biais  $b$ . Par conséquent, il peut être décrit de manière générale comme (Boujelben & Bahoura, 2016) :

$$\omega^T x + b = 0 \quad (3.1)$$

où l'exposant  $T$  indique la transposition et  $x$  est un vecteur quelconque de caractéristiques.

Cette droite paramétrée, lorsqu'elle est appliquée à un ensemble de données, crée deux sous-ensembles, que nous pouvons appeler le sous-ensemble gauche "A" et le sous-ensemble droit "B" comme la montre la Figure 3. 1, et que nous pouvons décrire plus précisément comme suit :

$$\begin{aligned} \text{si } x \in A &\rightarrow \omega^T x + b \geq 0 \\ \text{si } x \in B &\rightarrow \omega^T x + b < 0 \end{aligned} \quad (3.2)$$

Les étiquettes 1 pour le sous-domaine  $A$  et  $-1$  pour le sous-domaine  $B$  peuvent être utilisées pour différencier les points appartenant à ces sous-domaines. Par conséquent, l'objectif de paramétrage de la machine à vecteurs de support (SVM) peut être décrit comme suit (Suthaharan, 2016):

$$\begin{aligned} \omega^T x + b &= 1, x \in A \\ \omega^T x + b &= -1, x \in B \end{aligned} \quad (3.3)$$

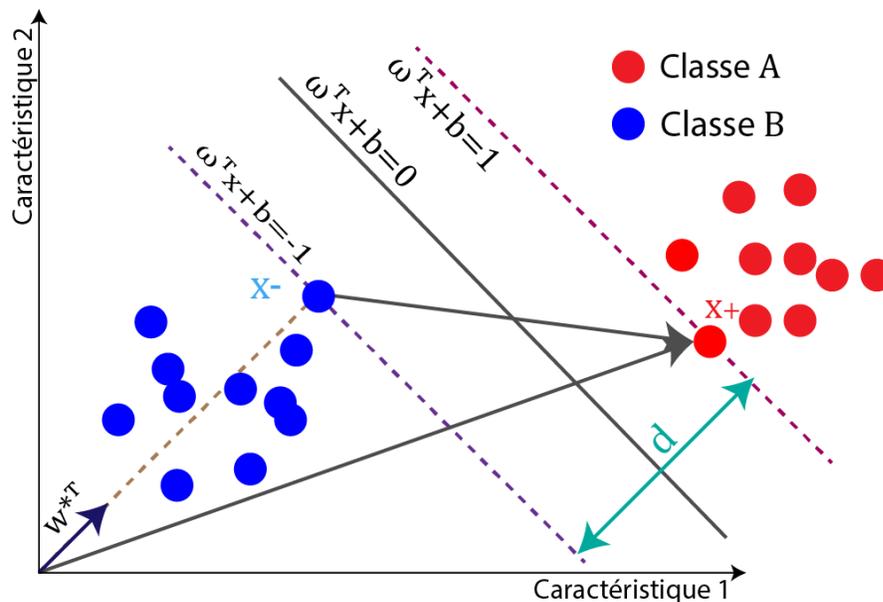


Figure 3. 1 Exemple de classification linéaire. Les deux caractéristiques représentent les composantes du vecteur  $x$ .

Deux lignes droites (ou hyperplans) qui peuvent être utilisées pour établir les frontières des classes ont été modélisées dans l'objectif de paramétrage. Le but de l'optimisation est de trouver les valeurs des paramètres qui maximisent une fonction objective donnée, dans ce cas la distance entre les lignes droites. Ces deux lignes sont parallèles l'une à l'autre. La distance entre ces lignes est définie par (Ukil, 2007):

$$d = \omega^{*T} (x_+ - x_-) = \omega^T \cdot \frac{x_+ - x_-}{|\omega|} = \frac{2}{|\omega|} \quad (3.4)$$

où  $x_+$  et  $x_-$  représentent, respectivement,  $x \in A$  et  $x \in B$ . Et  $\omega^{*T} = \frac{\omega^T}{|\omega|}$  est le vecteur unitaire de  $\omega^T$ . Cette distance, appelée "marge", est un paramètre clé pour choisir l'hyperplan optimal, c'est-à-dire celui qui offre la marge maximale possible. La Figure 3. 1 illustre cette marge définie par l'expression ci-dessus (3. 4). L'effet de la marge apparaît dans la Figure 3. 2.

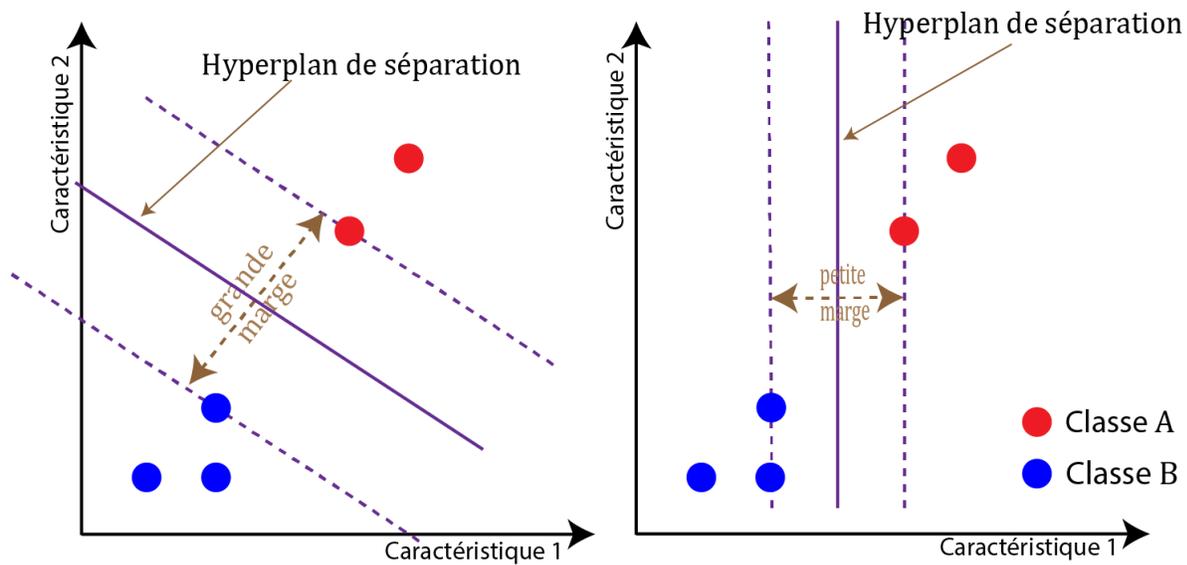


Figure 3. 2 Les hyperplans de séparation et les marges.

En considère que nous avons  $n$  points  $(x_i, i = 1 \dots n)$  dans l'espace des caractéristiques. En étiquetant  $y_i = 1$  et  $y_i = -1$ , respectivement, pour  $x_i \in A$  et  $x_i \in B$ , l'expression (3. 2) devient (Zhou, 2021d) :

$$y_i(\omega^T x_i + b) \geq 1 \quad (3. 5)$$

Par conséquent, l'objectif ici est de maximiser (3. 4) par rapport à  $(\omega, b)$  sous la contrainte de (3. 5). Autrement dit il s'agit de minimiser  $\frac{|\omega|}{2}$  par rapport à  $(\omega, b)$  sous la contrainte de (3. 5). Comme la minimisation de  $f$  et  $\sqrt{f}$  sont identiques (car la fonction racine carrée est une fonction monotone), nous aurons :

$$\max_{\omega, b} \frac{1}{2} |\omega|^2 \quad (3.6)$$

*Sous réserve de  $y_i(\omega^T x_i + b) \geq 1, i = 1, \dots, n$*

L'objectif est de trouver le modèle d'hyperplan séparateur à marge maximale en résolvant l'équation (3.6). Pour ce faire, on applique la méthode des multiplicateurs de Lagrange à l'équation (3.6), ce qui conduit à son problème dual. Pour être plus précis, on introduit un multiplicateur de Lagrange positif noté  $\alpha_i \geq 0$  pour chaque contrainte de l'équation (3.6), ce qui aboutit à la formulation de la fonction de Lagrange suivante (Zhou, 2021d):

$$L(\omega, b, \alpha) = \frac{1}{2} |\omega|^2 + \sum_{i=1}^n \alpha_i (1 - y_i(\omega^T x_i + b)) \quad (3.7)$$

où  $\alpha = (\alpha_1; \alpha_2; \dots; \alpha_n)$ . En fixant à 0 les dérivées partielles de  $L(\omega, b, \alpha)$  par rapport à  $\omega$  et  $b$ , on obtient :

$$\omega = \sum_{i=1}^n \alpha_i y_i x_i \quad (3.8)$$

$$0 = \sum_{i=1}^n \alpha_i y_i \quad (3.9)$$

La substitution de (3.9) dans (3.7) élimine  $\omega$  de  $L(\omega, b, \alpha)$ . Ensuite, avec la contrainte (3.9) nous avons le problème dual de (3.6) comme (Boujelben & Bahoura, 2018) :

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j^T \quad (3.10)$$

*Sous réserve de  $\sum_{i=1}^n \alpha_i y_i = 0, \quad C \geq \alpha_i \geq 0, \quad i = 1, 2, \dots, n.$*

La pénalité de mauvaise classification, notée  $C$ , représente le compromis entre la maximisation de la marge et la minimisation de l'erreur. La résolution de ce problème

d'optimisation permet d'obtenir les valeurs de  $\alpha$ , qui nous permettent à leur tour de déterminer les valeurs de  $\omega$  et de  $b$ . Nous obtenons ainsi le modèle souhaité

$$f(x) = \sum_{i=1}^n \alpha_i y_i x_i^T + b \quad (3.11)$$

La décision de classification pour le classificateur SVM linéaire est présentée comme suit :

$$d(x) = \text{sign}(f(x)) \quad (3.12)$$

### 3.1.2 CLASSIFICATEUR SVM NON LINEAIRE

Lorsque l'on traite des données qui ne sont pas linéairement séparables, les classificateurs SVM traditionnels peuvent rencontrer des limites. Pour relever ce défi, la méthode du noyau a été introduite comme solution pour traiter les données non linéairement séparables avec un classificateur SVM linéaire. L'idée centrale de la méthode du noyau est d'utiliser un mappage non linéaire pour convertir les entrées non linéairement séparables de leur espace original de faible dimension ( $\mathbb{R}^l$ ) en un espace de dimension supérieure ( $\mathbb{R}^h$ ), où  $h \geq l$ . Dans ce contexte, le vecteur d'entrée  $x$  subit une transformation en  $\phi(x)$  (Boujelben & Bahoura, 2018) :

$$\begin{aligned} \phi: \mathbb{R}^l &\rightarrow \mathbb{R}^h \\ x &\rightarrow \phi(x) \end{aligned} \quad (3.13)$$

Ensuite, une fonction noyau est établie comme  $K(x_i, x_j) = \phi^T(x_i)\phi(x_j)$ . Sous cette condition, la fonction optimale, désignée par (3.10), est alors transformée :

$$\begin{aligned} \max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) \\ \text{Sous réserve de } \sum_{i=1}^n \alpha_i y_i = 0, \quad C \geq \alpha_i \geq 0, \quad i = 1, 2, \dots, n. \end{aligned} \quad (3.14)$$

L'équation ci-dessous définit la fonction de décision dans le contexte de données non linéaires

$$d(x) = \text{sign} \left( \sum_{i=1}^n \alpha_i y_i K(x_i, x) + b \right) \quad (3.15)$$

### 3.1.3 FONCTION NOYAU

Une fonction noyau (*Kernel function*) transfère les données d'un espace à faible dimension vers un espace à plus haute dimension (Patle & Chouhan, 2013). Autrement dit, passer d'un espace non linéairement séparable à un espace linéairement séparable. Quatre noyaux typiques sont généralement utilisés (Boujelben & Bahoura, 2018).

- **Fonction noyau linéaire**

$$K(x_i, x_j) = x_i^T x_j \quad (3.16)$$

- **Fonction noyau polynomiale** Type equation here.

$$K(x_i, x_j) = (x_i^T x_j + 1)^d \quad (3.17)$$

où  $d$  est le degré de polynôme.

- **Noyau à fonction de base radiale (RBF)**

$$K(x_i, x_j) = \exp \left( -\frac{\|x_i - x_j\|^2}{2\sigma^2} \right) \quad (3.18)$$

- **Noyau Sigmoidale**

$$K(x_i, x_j) = \tanh(\alpha x_i^T x_j + \beta) \quad (3.19)$$

La Figure 3. 3 illustre l'utilisation de la classification linéaire pour séparer les classes non linéaires en les projetant dans un espace de plus grande dimension, passant ainsi de la bi-dimension à la tri-dimension.

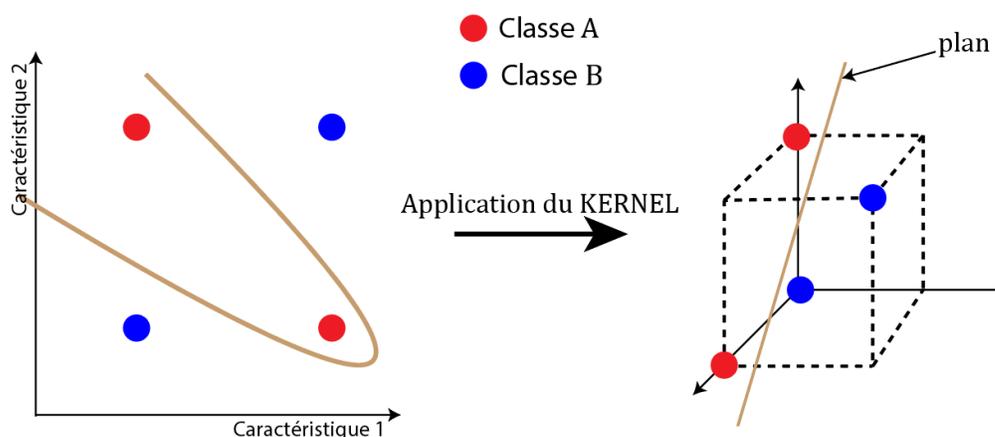


Figure 3. 3 Exemple de noyau pour deux classes.

### 3.1.4 MULTI-CLASSIFICATION PAR SVM

Comme le classificateur SVM, en complément d'autres algorithmes d'apprentissage automatique, a été conçu pour la classification binaire. Ce qui signifie que le nombre de classes à séparer est de deux. En pratique, Nous avons besoin d'une méthode ou d'une technique qui permette de l'étendre à des fins de multi-classification (plus de deux classes). Dans la littérature, deux approches sont généralement proposés pour atteindre cet objective (Hsu et Lin, 2002; Vural et Dy, 2004). Il s'agit de la technique « un contre tous » et de la technique « un contre un » :

#### a. Un contre tous (*One vs All/rest*):

La stratégie « un contre tous » consiste à crée un SVM par catégorie. Le nombre de classificateur est égale au nombre de classes. Chaque classificateur binaire est conçu pour séparer chaque classe par rapport aux restes des classes. Comme vous pouvez le voir dans la Figure 3. 4, pour classer les données en 3 classes (rouge, bleu et vert), nous devons créer trois classificateurs, tels que :

- Classificateur 1 : pour séparer la classe bleue de la rouge et la verte.
- Classificateur 2 : pour séparer la classe verte de la rouge et la bleue.

- Classificateur 3 : pour séparer la classe rouge de la bleue et la verte.

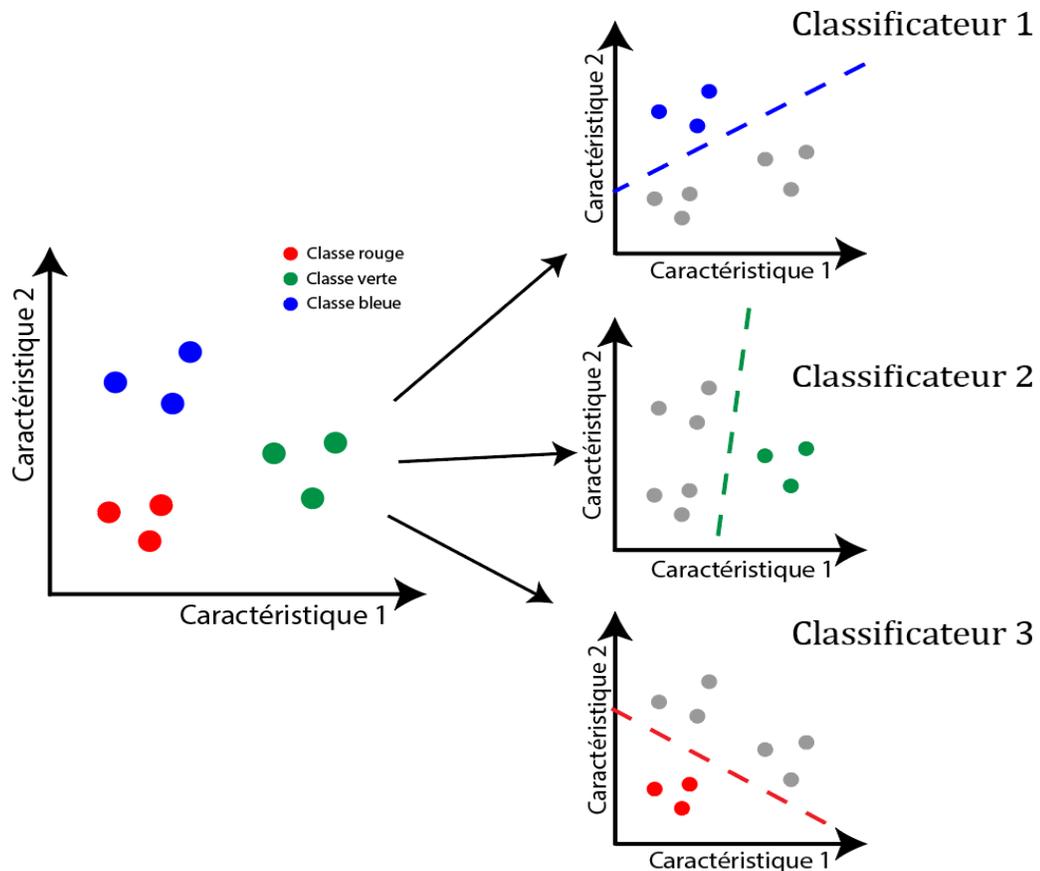


Figure 3. 4 Exemple de multi classification SVM utilisant la stratégie un contre tous.

Pour trouver la classe à laquelle appartient un nouveau point de données, nous le faisons passer par les trois classificateurs. En fonction du score de probabilité (calculée à partir du score SVM (Platt & others, 1999) représentant la distance signée de l'observation par rapport à la frontière de décision) de chaque classificateur, nous décidons quelle classe nous choisissons. Par exemple, si un nouveau point de données donne un score de probabilité de 0.9, 0.6 et 0.4, respectivement, pour le classificateur 1, le classificateur 2 et le classificateur 3, nous pouvons alors dire que cette donnée appartient à la classe bleue. Dans notre cas, nous avons besoin de classer les gestes de la main en 6 classes. Ce que nécessite 6 classificateurs « un contre tous ». L'inconvénient de cette approche est que les tailles des données d'entraînement des classificateurs ne sont pas équilibrées.

**b. Un contre un (*One vs one*) :**

Le principe de cette approche est de créer une frontière en prenant deux classes à la fois. Dans le cas d'une problématique de  $N$  classes, le nombre de classificateurs « un contre un » requis est défini par :

$$\text{Nombre de classificateurs} = \binom{N}{2} = \frac{N!}{2!(N-2)!} = \frac{N(N-1)}{2} \quad (3.20)$$

Dans la multi-classification « un contre un », chaque classificateur binaire prédit une étiquette de classe. Lorsque nous introduisons de nouvelles données dans tous les classificateurs, chaque modèle de classification binaire peut prédire une étiquette de classe et le modèle avec le plus de prédictions ou de votes est prédit comme l'étiquette de classe. Par exemple, dans Figure 3. 5, qui représente une multi-classification « un contre un » d'un ensemble de données à trois étiquettes de classe, nous avons besoin de trois classificateurs :

- Classificateur 1 : bleu contre vert.
- Classificateur 2 : bleu contre rouge.
- Classificateur 3 : vert contre rouge.

Si la nouvelle donnée est prédite comme bleue, bleue et verte, respectivement, dans le classificateur 1, le classificateur 2 et le classificateur 3, elle sera alors classée comme bleue. En cas d'égalité, une stratégie de départage peut-être adoptée. La plupart du temps, cette stratégie consiste à sélectionner au hasard aléatoirement l'une des étiquettes de classe qui sont à égalité (Pal, 2008).

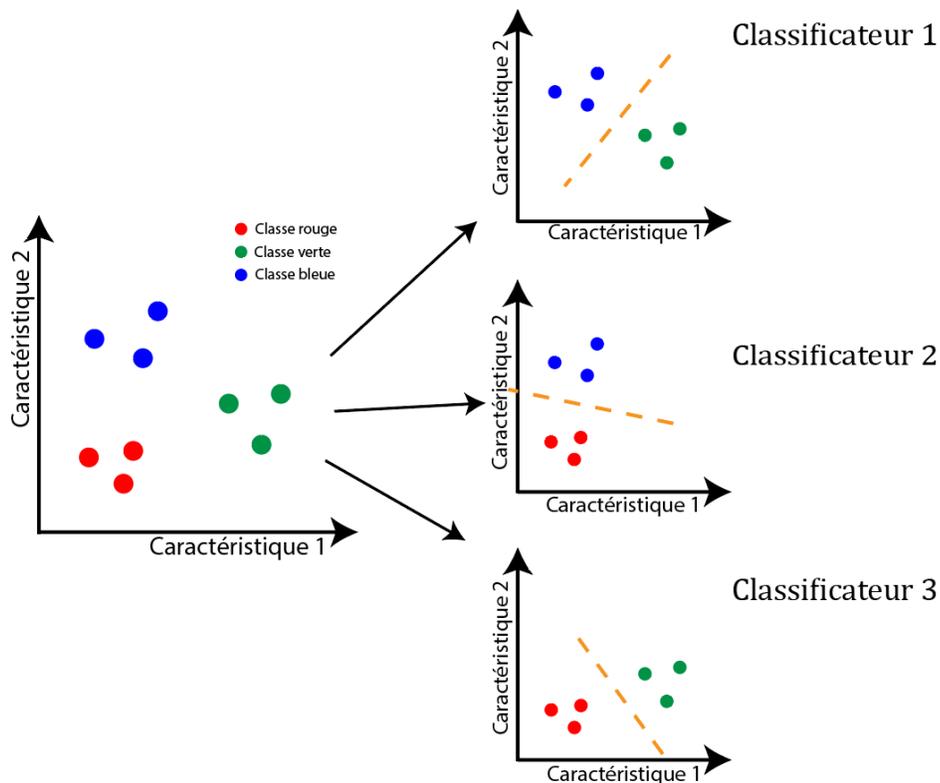


Figure 3. 5 Exemple de multi classification SVM utilisant la stratégie un contre un.

Dans notre cas de 6 classes de gestes de la main, nous aurons besoin de  $\frac{6 \times 5}{2} = 15$  classificateurs « un contre un ».

### 3.2 PERCEPTRON MULTICOUCHES (MLP)

Le perceptron multicouche MLP (*Multi-Layer Perceptron*) est une classe de réseaux de neurones artificiels ANN (*Artificial Neural Network*). Le réseau MLP contient des neurones organisés en couches selon une topologie *feedforward*, dans laquelle les entrées d'un neurone d'une couche quelconque sont fournies par les sorties des neurones de la couche qui le précède. Le réseau MLP contient une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Dans le réseau MLP à 3 couches illustré en Figure 3. 6, la première couche correspond au vecteur de caractéristiques. Le nombre de neurones à la couche d'entrée ( $N_e$ ) est défini par la taille du vecteur de caractéristiques, tandis que celui à

la couche de sortie ( $N_s$ ) est déterminé par le nombre de classes. Cependant, le nombre de neurones dans chaque couche cachée ( $N_c$ ) dépend de la complexité de la tâche de classification, ce qui affecte également le choix du nombre de couches cachées. Dans ce type de réseau, l'apprentissage s'effectue selon l'approche supervisée, où le résultat souhaité doit être connu à l'avance afin de définir les poids des connexions internes entre les couches. L'algorithme de rétropropagation est le plus utilisé pour l'apprentissage du réseau MLP. Dans cet algorithme, la sortie de chaque entrée d'apprentissage est calculée afin de déterminer l'erreur entre la sortie d'apprentissage souhaitée et la sortie calculée. Les poids des connexions sont continuellement modifiés, durant l'apprentissage, jusqu'à ce que le gradient d'erreur atteigne une petite valeur appropriée, indiquant que la sortie réelle est proche de la cible (Hussain *et al.*, 2020).

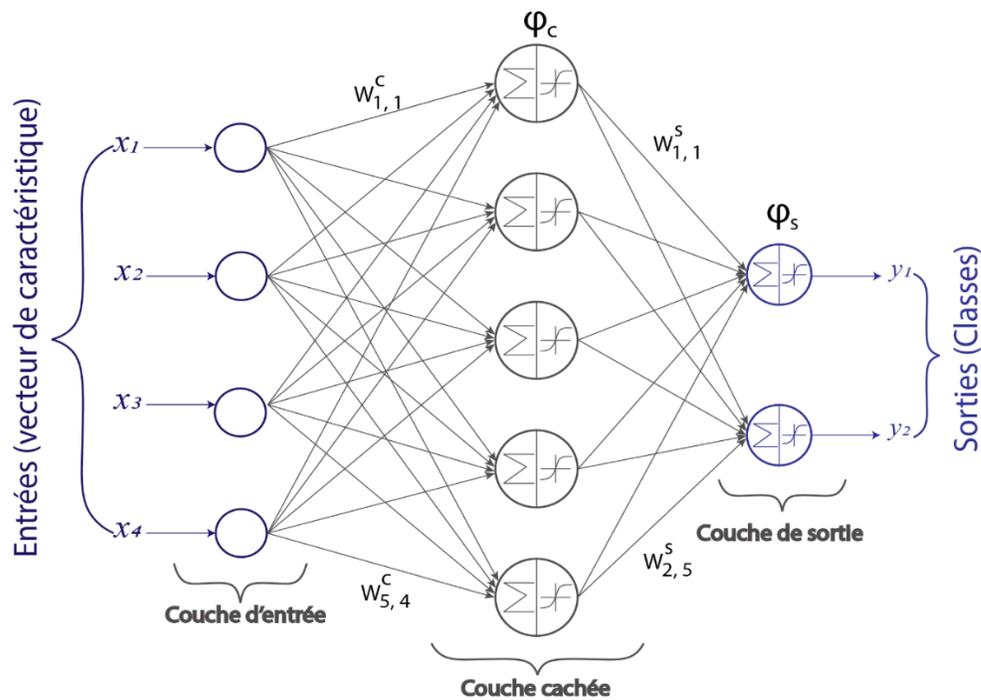


Figure 3. 6 Architecture générale d'un réseau de neurones avec une seule couche cachée. Il contient  $N_e = 4$  entrées,  $N_c = 5$  neurones cachés et  $N_s = 2$  neurones de sortie.

### 3.2.1 PERCEPTRON OU NEURONE ARTIFICIEL

Le terme "perceptron" désigne le réseau neuronal le plus basique. Il n'y a qu'une couche d'entrée et un nœud de sortie dans ce réseau neuronal (Aggarwal, 2018). La Figure 3. 7 présente la conception fondamentale du perceptron.

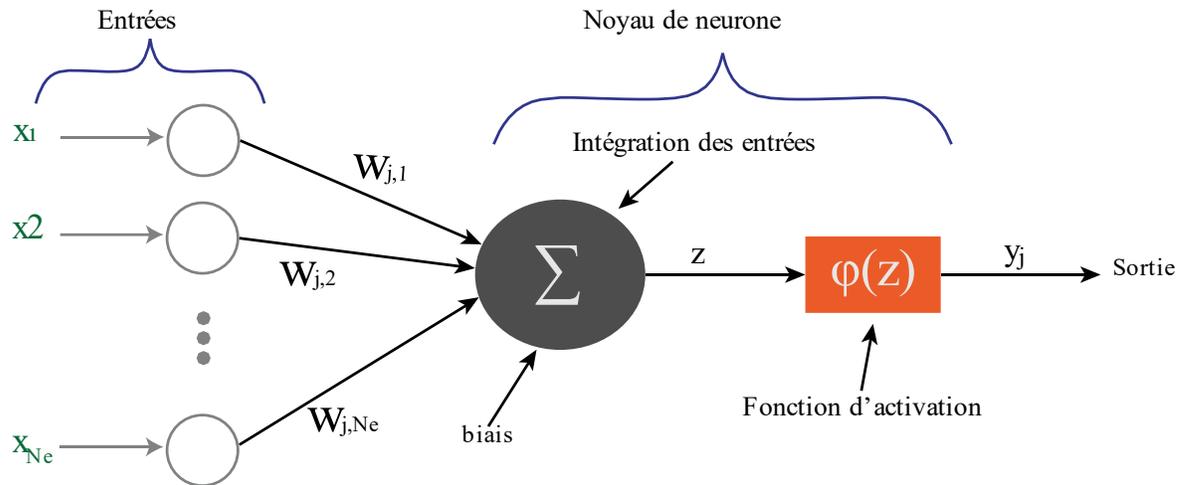


Figure 3. 7 Représentation d'un perceptron.

La sortie  $y_j$  du perceptron est définie par :

$$y_j = \varphi(z) = \varphi\left(\sum_{i=1}^{N_e} w_{j,i} \cdot x_i + b\right) \quad (3. 21)$$

où  $N_e$  est le nombre d'entrées,  $b$  est le biais lié au perceptron,  $w_{j,i}$  représente le poids de la connexion entre l'entrée  $x_i$  et le perceptron  $j$  et  $\varphi(z)$  est la fonction d'activation.

### 3.2.2 FONCTION D'ACTIVATION

Tout comme les neurones biologiques, les fonctions d'activation dans les réseaux de neurones artificiels régulent la sortie de chaque neurone, en la maintenant dans des limites prédéterminées. Un trait commun à plusieurs de ces fonctions d'activation est leur capacité à limiter ou à contrôler l'amplitude de la sortie du neurone.

- **Fonction sigmoïde**

La fonction sigmoïde, qui est une fonction continuellement différentiable (voir Figure 3. 8), est la fonction d'activation la plus utilisée en pratique (Graupe, 2013) :

$$\text{sig}(x) = \frac{1}{1 + \exp(-x)} \quad (3. 22)$$

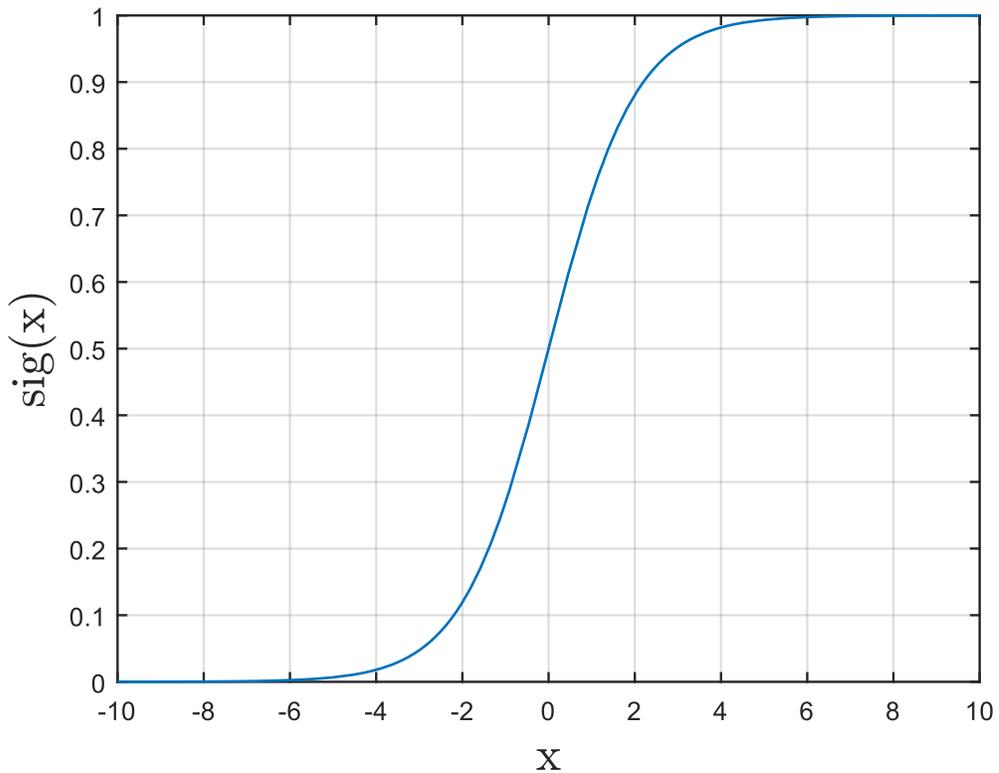


Figure 3. 8 Fonction sigmoïde.

Pour une entrée  $x$  de n'importe quelle valeur réelle, la sortie du neurone sera toujours comprise entre 0 et 1.

- **Fonction de tangente hyperbolique**

La fonction tangente hyperbolique, communément appelée *tanh*, est une fonction d'activation utilisée dans les réseaux neuronaux artificiels. Elle est similaire

à la fonction sigmoïde, mais sa plage de sortie est comprise entre -1 et 1, ce qui en fait une fonction centrée sur zéro (Figure 3. 9). Généralement, la fonction *tanh* est utilisée dans la couche cachée. Elle est définie par la formule suivante :

$$\tanh(x) = \frac{\exp(2x) - 1}{\exp(2x) + 1} \quad (3. 23)$$

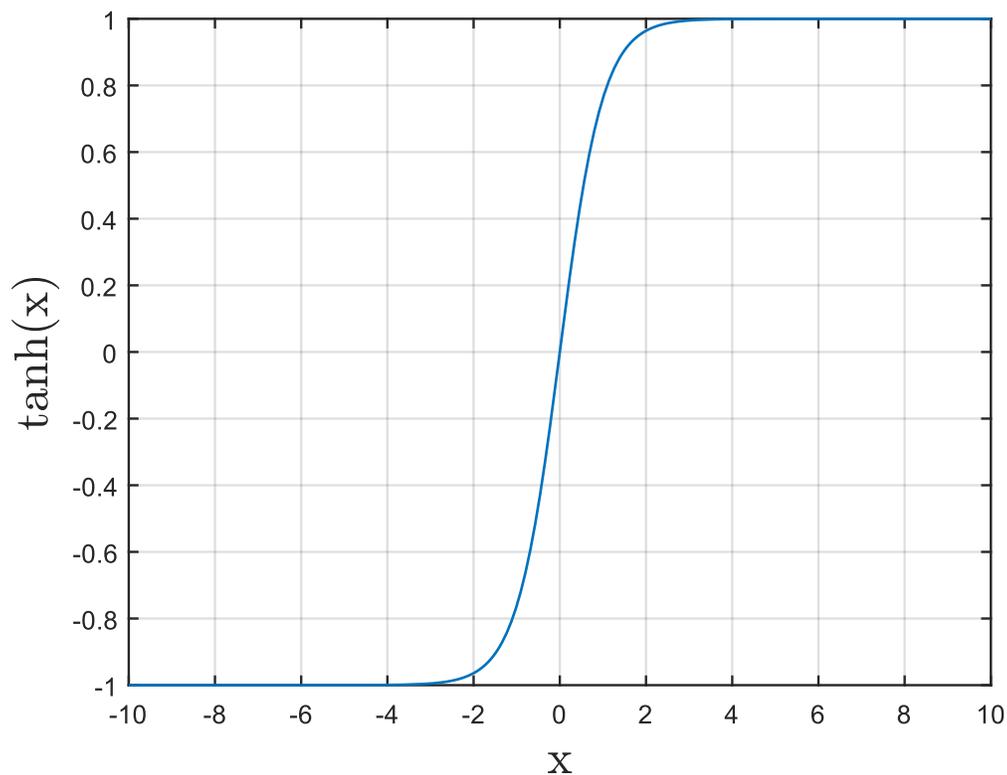


Figure 3. 9 Fonction tangente hyperbolique.

- **Fonction ReLU (Rectified Linear Unit)**

ReLU est une fonction d'activation très utilisée dans les réseaux neuronaux artificiels. Il s'agit d'une fonction simple mais efficace qui a été largement adoptée pour sa capacité à introduire la non-linéarité dans un modèle (Figure 3. 10). La fonction ReLU est définie comme suit :

$$f(x) = \max(0, x) \quad (3.24)$$

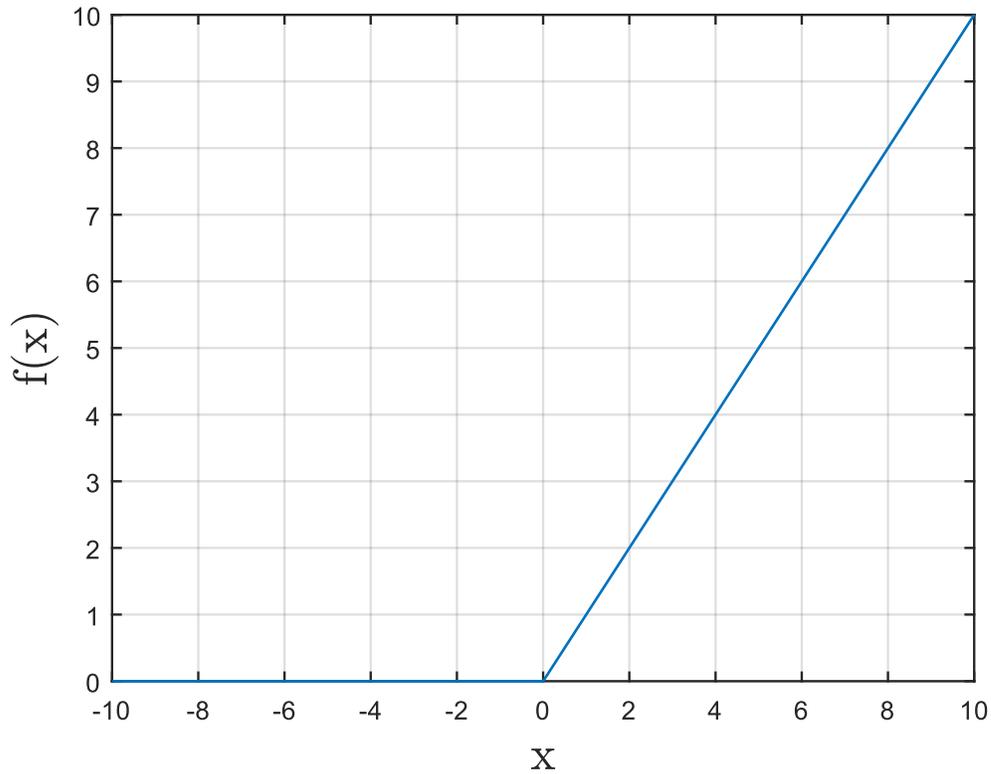


Figure 3. 10 Fonction ReLU.

C'est une fonction qui remplace les valeurs négatives à son entrée par des zéros.

- **Fonction Softmax**

La fonction Softmax est une version étendue du sigmoïde. Elle est utilisée dans les problèmes impliquant une classification multi-classes comme couche finale. Car elle produit des probabilités pour chaque entrée dans quelle classe elle se situerait :

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{N_s} e^{x_j}}, (i = 1 \dots N_s) \quad (3.25)$$

où  $x_i$  est la valeur d'entrée de la couche softmax et  $f(x_i)$  est la valeur de sortie, qui indique la probabilité que l'échantillon tombe dans la  $i^{th}$  catégorie (Aggarwal, 2018).  $N_s$  est le nombre de neurones dans la couche de sortie qui correspond au nombre de classes.

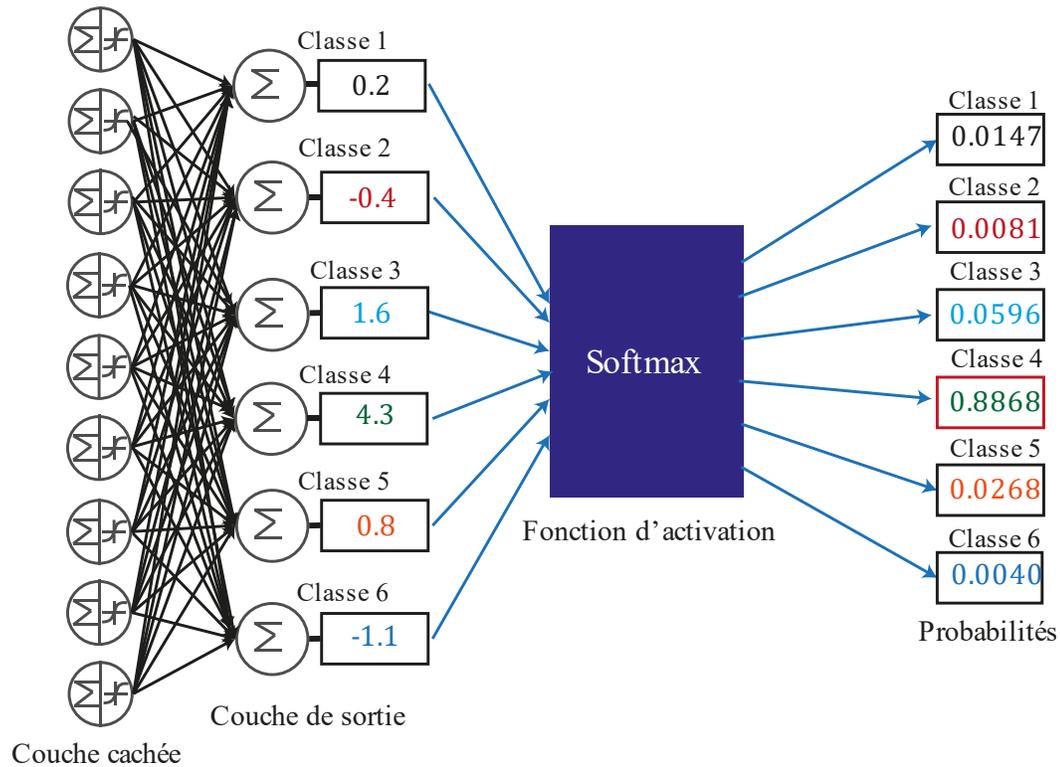


Figure 3. 11 Représentation schématique de l'application de la fonction d'activation Softmax dans un réseau de neurones pour la catégorisation multi-classes.

La Figure 3. 11 illustre un exemple d'utilisation du Softmax. Les valeurs du côté droit représentent les probabilités attribuées à chaque classe par la fonction Softmax. Dans ce cas, la "Classe 4" a la probabilité la plus élevée (0.8668), ce qui indique que le réseau a prédit la "Classe 4" avec un degré de probabilité élevé pour l'entrée présentée au réseau de neurones.

### 3.2.3 L'ALGORITHME DE RETROPROPAGATION

L'algorithme de rétropropagation est un processus d'apprentissage supervisé où la sortie doit être connue à l'avance pour mettre à jour les poids des connexions entre les

neurones des différentes couches. Les poids des connexions sont continuellement ajustés durant la phase d'apprentissage de manière à ce que l'erreur quadratique moyenne entre la sortie souhaitée et la sortie obtenue atteigne la plus faible valeur possible (Hussain *et al.*, 2020). La descente de gradient est l'approche la plus couramment utilisée dans les réseaux MLP pour mettre à jour les poids des connexions. Par conséquent, les fonctions d'activation du perceptron neurone doivent être différentiables afin d'appliquer la descente de gradient. Elle peut être implémentée de deux manières différentes : le mode séquentiel et le mode par lot (Bahoura & Park, 2012). La mise à jour de la valeur des poids dans la descente de gradient dépend des valeurs précédentes et du gradient de l'erreur quadratique moyenne par rapport au poids concerné. La formulation de l'algorithme mentionné est la suivante :

$$w_{j,i}(n + 1) = w_{j,i}(n) - \alpha \frac{\delta E(n)}{\delta w_{j,i}(n)} \quad (3. 26)$$

Avec  $E(n)$  représente l'erreur quadratique moyenne, elle est exprimée comme suit :

$$E(n) = \frac{1}{K} \sum_{k=1}^K (d_k(n) - y_k(n))^2 \quad (3. 27)$$

où  $d_k$ ,  $y_k$  représentent la sortie désirée et la sortie réelle du perceptron  $k$  de la couche de sortie. Le paramètre  $n$  représente la  $n^{\text{ème}}$  donnée d'apprentissage et  $\alpha$  est le taux d'apprentissage.  $K$  est le nombre de neurones de sortie qui correspond au nombre de classes.

### 3.3 METHODE DU K PLUS PROCHES VOISINS (K-NN)

La méthode des  $k$  plus proches voisins  $k$ -NN (*k-Nearest Neighbors*) est une technique de classification à apprentissage supervisée non paramétrique, ce qui signifie qu'elle ne fait aucune hypothèse sur les données et la distribution sous-jacentes. En d'autres termes, la structure du modèle est déterminée par les données (Guo *et al.*, 2003).

Le principe de l'algorithme k-NN est de trouver les k éléments de données d'entraînement les plus proches d'un échantillon de test en utilisant une métrique de distance, puis d'utiliser ces k "voisins" pour faire des prédictions. Le vote peut typiquement être utilisé pour prévoir l'échantillon de test comme l'étiquette de classe la plus répandue dans les k proches pour les problèmes de classification, et la moyenne peut typiquement être utilisée pour prévoir l'échantillon de test comme la moyenne des k sorties à valeurs réelles pour les problèmes de régression. En outre, les échantillons peuvent être pondérés en fonction de leur distance, un échantillon plus proche se voyant attribuer un poids plus important (Zhou, 2021a).

L'objectif du classificateur k-NN est de classer une nouvelle observation  $x$  provenant d'un espace de vecteurs de caractéristiques à  $D$  dimensions dans la bonne classe de l'ensemble  $\{\omega_1, \dots, \omega_K\}$ , où  $K$  représente le nombre total de classes. Considérons un ensemble de données d'apprentissage avec  $N$  observations étiquetées  $O_N = \{\{x_1, \omega_1\}, \dots, \{x_N, \omega_N\}\}$ , où  $x_i$  et  $\omega_i$  désigne les vecteurs de caractéristiques et les étiquettes de classe qui leur sont associées. Lors de la classification d'une nouvelle observation  $x$  avec une classe indéterminée, un volume  $V$  est désigné autour de  $x$  pour encapsuler  $k$  prototypes. Définissons  $k_j$  comme le nombre d'échantillons étiquetés avec l'étiquette de classe  $\omega_j$  qui sont inclus dans ce volume (de sorte que  $\sum_{j=1}^K k_j = k$ ), la probabilité conjointe peut être estimée comme suit (Bahoura & Simard, 2012) :

$$p_N(x, \omega_j) = \frac{k_j/V}{k} \quad (3.28)$$

et permet d'obtenir une estimation de la probabilité postérieure.

$$p_N(\omega_j|x) = \frac{p_N(x, \omega_j)}{\sum_{j=1}^M p_N(x, \omega_j)} = \frac{k_j}{k} \quad (3.29)$$

Cette nouvelle observation  $x$  sera classée par vote majoritaire dans la classe  $\omega$ .

$$\omega = \underset{1 \leq j \leq K}{\operatorname{arg\,max}} \{k_j\} \quad (3.30)$$

En raison de l'importance de la précision des prédictions, le choix de la valeur de  $k$  est un aspect très important de l'algorithme  $k$ -NN et ne doit pas être considéré comme un élément acquis. Si l'algorithme apprend à partir d'un ensemble de données bruitées, le choix de petites valeurs pour  $k$  conduira probablement à une précision réduite, et si la valeur de  $k$  est trop grande, le modèle peut être sur ajusté et fournir une mauvaise précision. Afin d'éviter les égalités dans le vote majoritaire, un nombre impair  $k$  devrait également être utilisé si le nombre de classes est pair (Kotsiantis *et al.*, 2006).

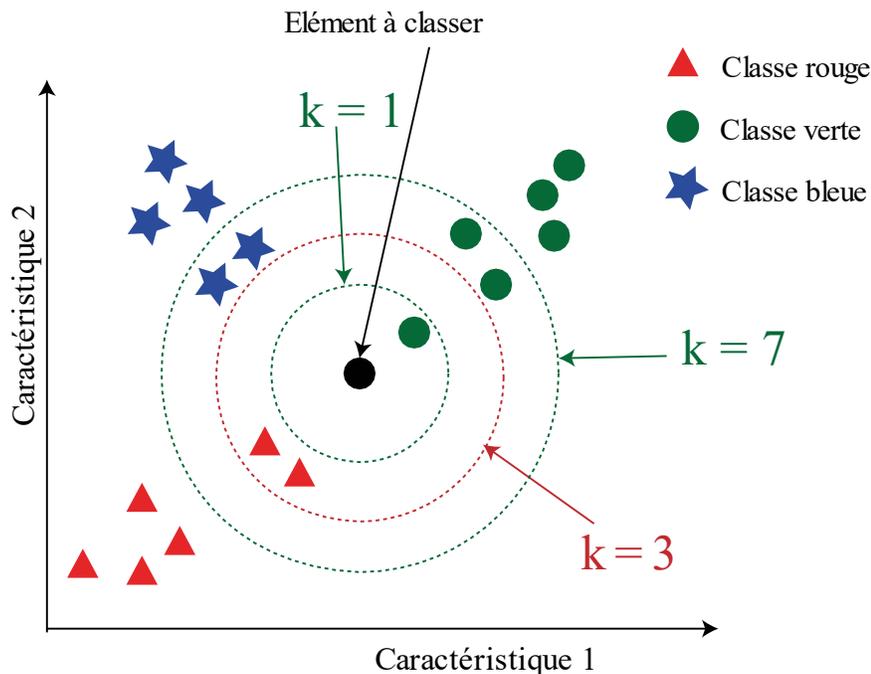


Figure 3. 12 Règle de décision  $k$ -NN pour les valeurs  $k = 1, k = 3$  et  $k = 7$ .

Considérons que nous avons un problème de multi-classification avec le rouge, le bleu et le vert comme étiquettes de classes, tel que le présente la Figure 3. 12, et que nous devons trouver à quelle classe appartient l'échantillon noir. Nous classons le nouvel échantillon avec l'algorithme  $k$ -NN en regardant les échantillons les plus proches de lui (c'est-à-dire les "plus proches voisins"). Si le  $k$  est égal à 1, alors nous n'utilisons que le plus

proche voisin pour définir la catégorie. Dans ce cas, la catégorie est le vert. Si  $k = 3$ , nous utilisons les 3 plus proches voisins. Comme le montre la Figure 3. 12, nous aurons 2 voisins rouges et 1 voisin vert. Par conséquent, le nouvel échantillon est classé comme rouge puisqu'il obtient le plus de votes.

### 3.3.1 CALCUL DE LA DISTANCE

Il existe de nombreuses façons de calculer la distance entre un point de données inconnu et ses plus proches voisins dans l'espace des caractéristiques. Quatre techniques sont couramment utilisées : distance Euclidienne, distance de Manhattan, distance de Minkowski et distance de Mahalanobis.

Supposons deux points de données de vecteurs  $x$  et  $y$  dans un espace de caractéristiques à  $N$  dimensions :

- **Distance Euclidienne:**

$$d(x, y) = \sqrt{\sum_{i=1}^N (x_i - y_i)^2} \quad (3. 31)$$

- **Distance de Manhattan:**

$$d(x, y) = \sum_{i=1}^N |x_i - y_i| \quad (3. 32)$$

- **Distance de Minkowski:**

$$d(x, y) = \left( \sum_{i=1}^N |x_i - y_i|^p \right)^{\frac{1}{p}} \quad (3. 33)$$

Cela équivaut à la distance euclidienne lorsque  $p = 2$ , et à la distance de Manhattan lorsque  $p = 1$ .

- **Distance de Mahalanobis:**

$$d(x, y) = \sqrt{(x - y)^T A (x - y)} \quad (3. 34)$$

où  $A$  est une matrice définie positive. Si  $A = I$  (matrice d'identité), cela se réduit à la distance euclidienne.

### 3.4 ANALYSE DISCRIMINANTE LINEAIRE LDA

L'analyse discriminante linéaire LDA (*Linear Discriminant Analysis*) est une méthode de classification à apprentissage supervisé. L'objectif de l'analyse discriminante linéaire est de trouver une combinaison linéaire de caractéristiques qui maximise simultanément la dispersion entre les classes et minimise la dispersion au sein de chacune des classes. Les échantillons d'une même classe apparaissent donc aussi proches que possible dans l'espace projection, tandis que les échantillons des autres classes apparaissent aussi éloignés que possible (Zhu *et al.*, 2022). Lors de la catégorisation de nouveaux échantillons, ceux-ci sont projetés sur le même ligne, et les positions projetées des échantillons établissent les classes auxquelles ils appartiennent. Un exemple de caractérisation bidimensionnel est présenté à la Figure 3. 13.

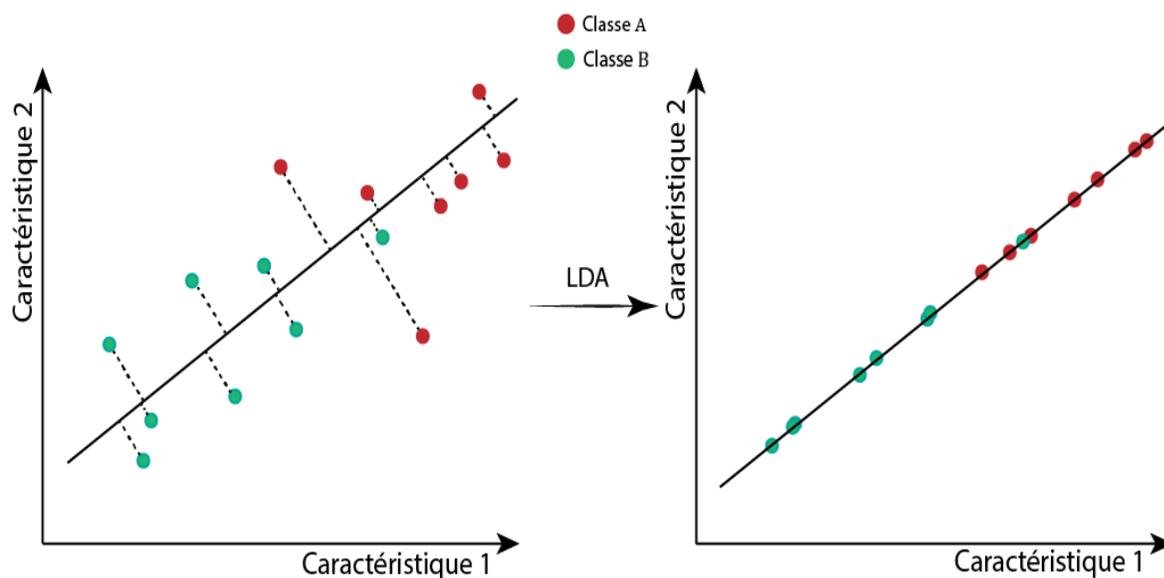


Figure 3. 13 Exemple d'utilisation de l'analyse discriminante linéaire.

Soit  $D = \{(x_i, y_i)\}, i = 1, \dots, N$  un ensemble de données, avec  $y_i \in \{0, 1\}$ . Soit  $X_i$ ,  $\mu_i$  et  $\Sigma_i$  désignent, respectivement, l'ensemble des échantillons, le vecteur moyen et la matrice de covariance de la  $i^{\text{ème}}$  classe. Nous visons à projeter les données sur la droite définie par  $\omega$ . Après cette projection, les centres des échantillons des deux classes sont représentés respectivement par  $\omega^T \mu_0$  et  $\omega^T \mu_1$ . De même, les covariances associées à ces deux classes sont respectivement  $\omega^T \Sigma_0 \omega$  et  $\omega^T \Sigma_1 \omega$  (Zhou, 2021b).

Pour rendre les points de projection des échantillons de même classe aussi proches que possible, nous devons minimiser la covariance des points de projection des échantillons de même classe aussi petite que possible, c'est-à-dire minimiser  $\omega^T \Sigma_0 \omega + \omega^T \Sigma_1 \omega$ . Pour que les points de projection des échantillons de différentes classes soient aussi éloignés que possible, nous devons augmenter la distance entre les centres de classe, c'est-à-dire maximiser  $\|\omega^T \mu_0 - \omega^T \mu_1\|$ . En combinant les deux critères, nous avons le critère suivant à maximiser (Zhou, 2021b) :

$$J = \frac{\|\omega^T \mu_0 - \omega^T \mu_1\|_2^2}{\omega^T \Sigma_0 \omega + \omega^T \Sigma_1 \omega} \quad (3.35)$$

La Figure 3.14 illustre la différence entre la projection des données en se concentrant seulement sur la maximisation de la distance entre les moyennes et la projection des données en maximisant la distance entre les moyennes tout en minimisant la covariance. Lorsque nous optimisons simultanément ces deux critères, maximiser l'écart entre les moyennes et minimiser la covariance, nous obtenons une séparation plus efficace.

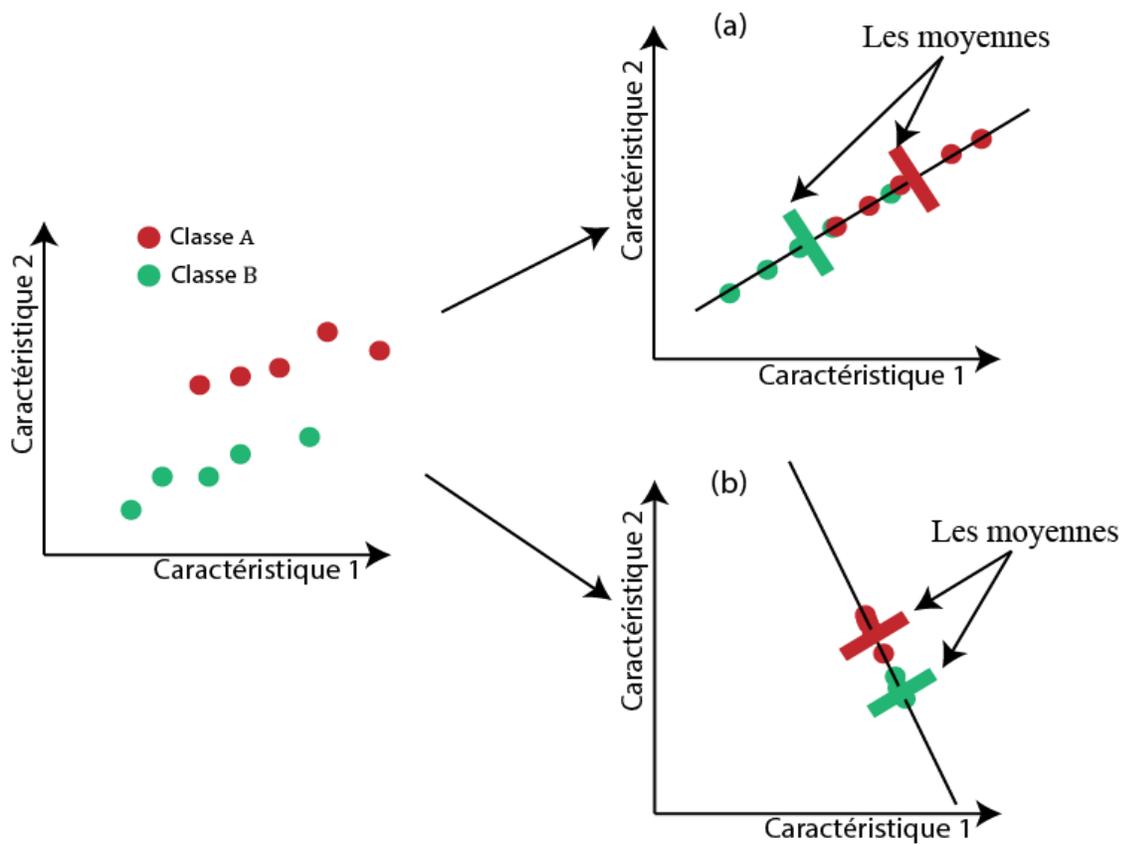


Figure 3. 14 Différence entre les modes de projection dans LDA: (a) Maximisation seulement de la distance entre les moyennes; (b) Maximisation la distance entre les moyennes et minimiser la covariance.

### 3.5 MATRICE DE CONFUSION

Un tableau appelé matrice de confusion est fréquemment employé pour évaluer l'efficacité d'un modèle de classification. Il fournit un résumé tabulaire des résultats de classification réelle et prédite, de sorte que les performances du modèle peuvent être facilement examinées. En comparant les résultats de la classification prédite à ceux de la classification réelle, la matrice de confusion est créée en comptant le nombre de vrais positifs

(TP), de faux positifs (FP), de vrais négatifs (TN) et de faux négatifs (FN). Le Tableau 3. 1 montre une matrice de confusion de la classification binaire (deux classes).

Tableau 3. 1 Matrice de confusion en classification binaire

		Classe réelle	
		Classe 0	Classe 1
Classe prédite	Classe 0	TP	FP
	Classe 1	FN	TN

L'exactitude est calculée comme suit :

$$Exactitude = \frac{TN + TP}{TN + TP + FP + FN} \quad (3. 36)$$

Dans le cas de la multi-classification, disons 3 classes, le tableau est défini par (Tharwat, 2018):

Tableau 3. 2 Matrice de confusion en multiclasse.

		Classe réelle		
		Classe A	Classe B	Classe C
Classe prédite	Classe A	$TP_A$	$E_{BA}$	$E_{CA}$
	Classe B	$E_{AB}$	$TP_B$	$E_{CB}$
	Classe C	$E_{AC}$	$E_{BC}$	$TP_C$

La matrice de confusion du Tableau 3. 2 présente les résultats d'un problème de classification multi-classes à trois classes (A, B et C).  $TP_A$  désigne le nombre d'échantillons correctement classés dans la classe A, également appelés échantillons vrais positifs.  $E_{AB}$  représente les échantillons mal classés de la classe B en tant que classe A. Le faux négatif ( $FN_A$ ) de la classe A est la somme de  $E_{AB}$  et  $E_{AC}$ , c-à-d,  $FN_A = E_{AB} + E_{AC}$ , qui indique le nombre total d'échantillons de la classe A incorrectement classés en tant que classe B ou C.

Le faux négatif de toute classe dans une colonne peut être calculé en additionnant les erreurs de cette colonne. Le faux positif de toute classe prédite dans une ligne représente la somme de toutes les erreurs de cette ligne. Par exemple, le faux positif ( $FP_A$ ) de la classe A est calculé comme suit :  $FP_A = E_{BA} + E_{CA}$ .

L'exactitude de la classification utilisée dans ce mémoire est calculée comme suit :

$$Exactitude = \frac{TP_A + TP_B + TP_C}{TP_A + E_{AB} + E_{AC} + TP_B + E_{BA} + E_{BC} + TP_C + E_{CA} + E_{CB}} \quad (3.37)$$

### 3.6 APPROCHE PROPOSEE POUR LA CLASSIFICATION

Pour l'étape de classification automatique des signaux EMG, notre projet consiste à développer quatre modèles de classification distincts : SVM, k-NN, LDA et MLP. Le premier modèle proposé est le SVM, qui sera réalisé en utilisant l'algorithme *one-vs-one* pour la classification de 6 classes. Le deuxième modèle à tester est le système k-NN qui utilise la distance euclidienne pour mesurer le voisinage. Nous allons aussi effectuer des tests avec différentes valeurs de k ( $k = 1, \dots, 8$ ). Le troisième modèle à tester est le système LDA, utilisant une projection quadratique. Finalement, le modèle basé sur le réseau de neurones MLP sera optimisé en menant une étude paramétrique sur le nombre de neurones et de couches cachées utilisés par le réseau.

## CHAPITRE 4

### RESULTATS EXPERIMENTAUX

Dans ce chapitre, nous présentons les résultats de notre recherche sur la reconnaissance automatique des gestes de la main par analyse des signaux électromyographiques (EMG). Nous détaillons également les outils et l'équipement utilisées durant cette expérience : la base de données utilisée, les méthodes de prétraitement et d'extraction des caractéristiques appliquées, ainsi que l'évaluation des performances des divers algorithmes de classification. En outre, les résultats obtenus sont comparés à ceux obtenus par d'autres chercheurs et la faisabilité d'une mise en œuvre dans les conditions réelles a été étudiée. Ce chapitre fournit également des indications précieuses sur l'efficacité de la reconnaissance automatique des gestes de la main utilisant les signaux EMG et sur ses applications potentielles du système de classification développé.

#### 4.1 OUTILS ET MATERIEL UTILISEES

Au début de ce chapitre, nous présentons les spécificités techniques et logicielles utilisées pour les expériences décrites dans les sections suivantes. Cette section inclue les détails sur l'équipement informatique, les dispositifs de capture des signaux, ainsi que les logiciels et bibliothèques employés.

- **Configuration de l'ordinateur :** Un ordinateur muni d'un processeur Intel i7 9300H, d'une carte graphique NVIDIA 1650GTX, d'une mémoire RAM de 16 Go et d'un système d'exploitation Windows 11.
- **Capture des signaux EMG :** brassard Myo Gesture Armband de Thematic Lab.
- **Traitement des données et simulations :** Logiciel MATLAB R2022b avec les boites à outils (*toolboxes*) nécessaires pour le traitement.

- **Lecture du signal EMG du brassard Myo Gesture Armband** : Utilisation de la bibliothèque MyoMex (*Myo SDK MATLAB MEX Wrapper*) créée par (Tomaszewski, 2023) ; Kit de développement logiciel SDK (*Software Development Kit*) pour le brassard MyoArmband (Rosenstein, 2019).

## 4.2 MYO GESTURE ARMBAND

Le brassard Myo Gesture Armband est un dispositif qui utilise des techniques non invasives pour enregistrer le signal EMG (électromyographie de surface). En outre, il se compose également d'une unité de mesure inertielle IMU (*Inertial Measurement Unit*) qui comprend un magnétomètre, un gyroscope et un accéléromètre. Il a été développé par Thematic Lab en 2014 (Rawat *et al.*, 2017). Vous pouvez le porter sur votre avant-bras, sous le coude. Le brassard est rechargeable et offre une journée complète d'utilisation au sujet. Il possède 8 capteurs EMG pour détecter les mouvements musculaires autour de l'avant-bras et de la main. Il est également doté d'un port Micro-USB, d'une LED de logo et d'une LED d'état. Vous pouvez le relier à un ordinateur via une liaison sans fil afin (Bluetooth) de recueillir les informations (signaux EMG, gyroscope, etc.). La Figure 4. 1 suivante représente le Brassard Myo Gesture Armband et ses composants. La Figure 4. 2 montre les traces des 8 signaux EMG enregistrés à partir de 8 canaux du brassard Myo Gesture Armband utilisant l'environnement MATLAB.

Le brassard a une circonférence initiale de 19 cm, pouvant être étendue jusqu'à 34 cm pour s'adapter à la taille de l'avant-bras du sujet. Il est équipé de 8 capteurs EMG de surface active, dont l'écartement peut varier entre 0.5 et 1.8 cm. Le brassard doit être positionné sur la partie supérieure de l'avant-bras afin d'optimiser la couverture de la zone où les muscles sont les plus présents, comme expliqué par (Raurale *et al.*, 2021). La Figure 4. 3 montre le brassard placé au hasard sur la surface de la peau de l'avant-bras et la vue en coupe intérieure des muscles de la partie supérieure de l'avant-bras (Raurale *et al.*, 2021).

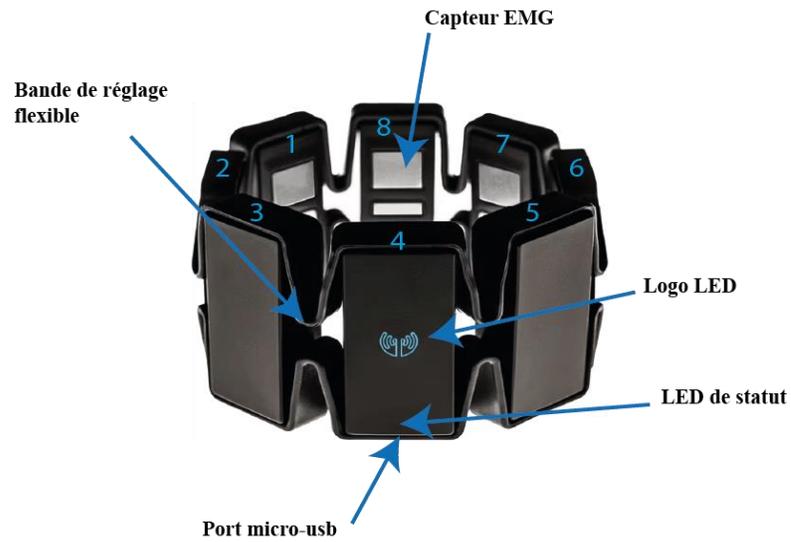


Figure 4. 1 Myo Gesture Armband développé par Thematic Lab.

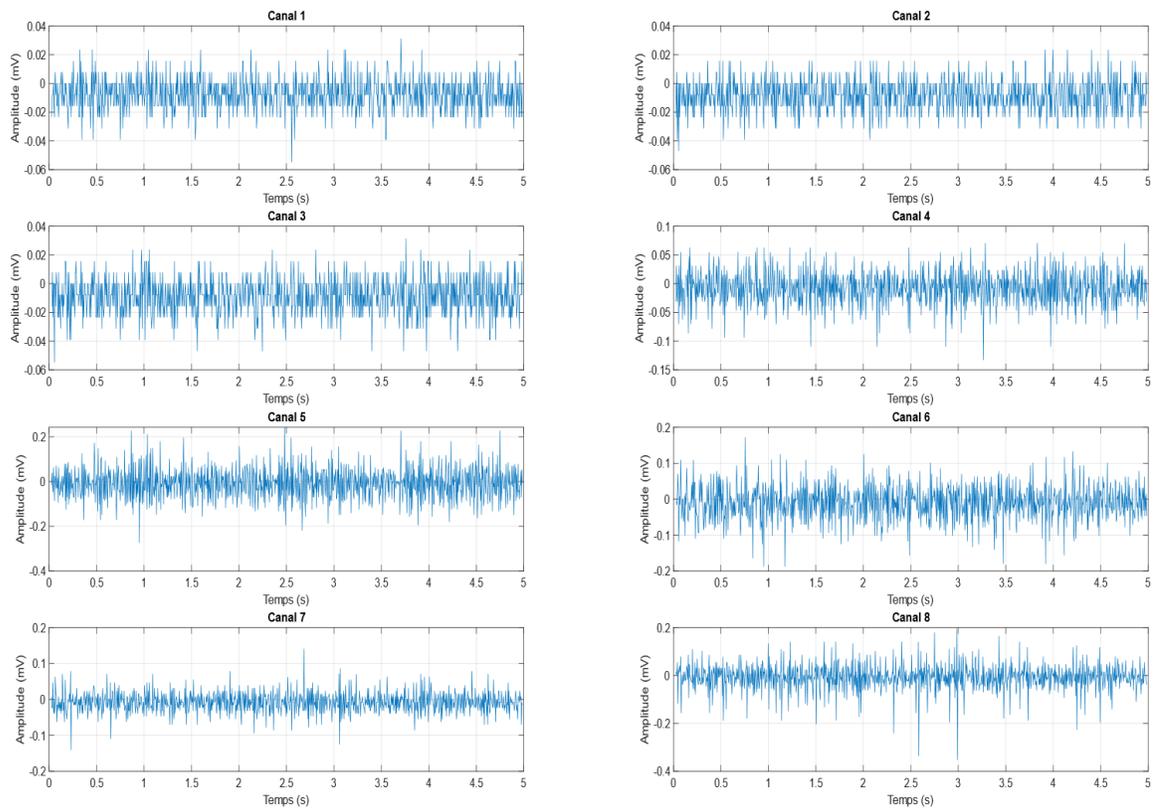


Figure 4. 2 Représentation des signaux EMG à 8 canaux collectés à partir du brassard Myo Gesture Armband.



Figure 4. 3 Placement du brassard Myo Gesture Armband avec vue en coupe des muscles de l'avant-bras (Raurale *et al.*, 2020).

### 4.3 BASE DE DONNEES

Nous avons créé notre propre base de données à l'aide du brassard Myo Gesture Armband qui appartient à la catégorie des capteurs des signaux EMG de surface en utilisant l'environnement MATLAB. Tout d'abord, nous avons téléchargé l'outil SDK (Rosenstein, 2019). Nous avons ensuite ajouté le chemin du dossier "bin" contenu dans le dossier SDK aux variables d'environnement du système Windows. Après avoir effectué ces étapes, nous avons utilisé la bibliothèque MyoMex créée par (Tomaszewski, 2023) pour pouvoir lire les données captées par le brassard Myo Armband. Par la suite, nous avons procédé à la création d'un petit programme, utilisant la classe MATLAB MyoMex, qui nous permet de lire les signaux EMG du brassard Myo Armband et de le sauvegarder dans un fichier ".mat". Les données collectées comprennent six mouvements de la main gauche, à savoir : position de repos, mouvement vers la gauche, mouvement vers la droite, main ouverte, main fermée et mouvement vers le haut. La collecte des données a été effectuée auprès de 10 participants sains, composés de 8 hommes et 2 femmes, âgés de 25 à 32 ans. Chacun des participants a réalisé la même série de six gestes, répétant chaque geste dix fois de manière séquentielle. La durée de chaque geste était de 5 secondes, comme illustré dans la Figure 4. 4. Chaque

geste contient 1000 échantillons ( $200 \text{ Hz} \times 5 \text{ s} = 1000$ ) par canal. Ainsi, au total, nos données contiennent 600000 échantillons ( $5 \text{ s} \times 200 \text{ Hz} \times 6 \text{ gestes} \times 10 \text{ répétitions} \times 10 \text{ personnes}$ ) pour chaque canal. Donc, le fichier '.mat' contient une variable MATLAB de type cellule comprenant 600 ( $6 \text{ gestes} \times 10 \text{ répétitions} \times 10 \text{ personnes}$ ) lignes et 2 colonnes. La première colonne représente le signal EMG, tandis que la seconde colonne indique le geste de la main, variant de 1 à 6. Ainsi, chaque ligne contient le signal EMG sous la forme d'une matrice MATLAB de 8 lignes, correspondant à 8 canaux, et de 1000 colonnes correspondant aux données collectées pour un seul geste ( $200 \text{ Hz} \times 5 \text{ s} = 1000$ ) avec le geste de la main associé.

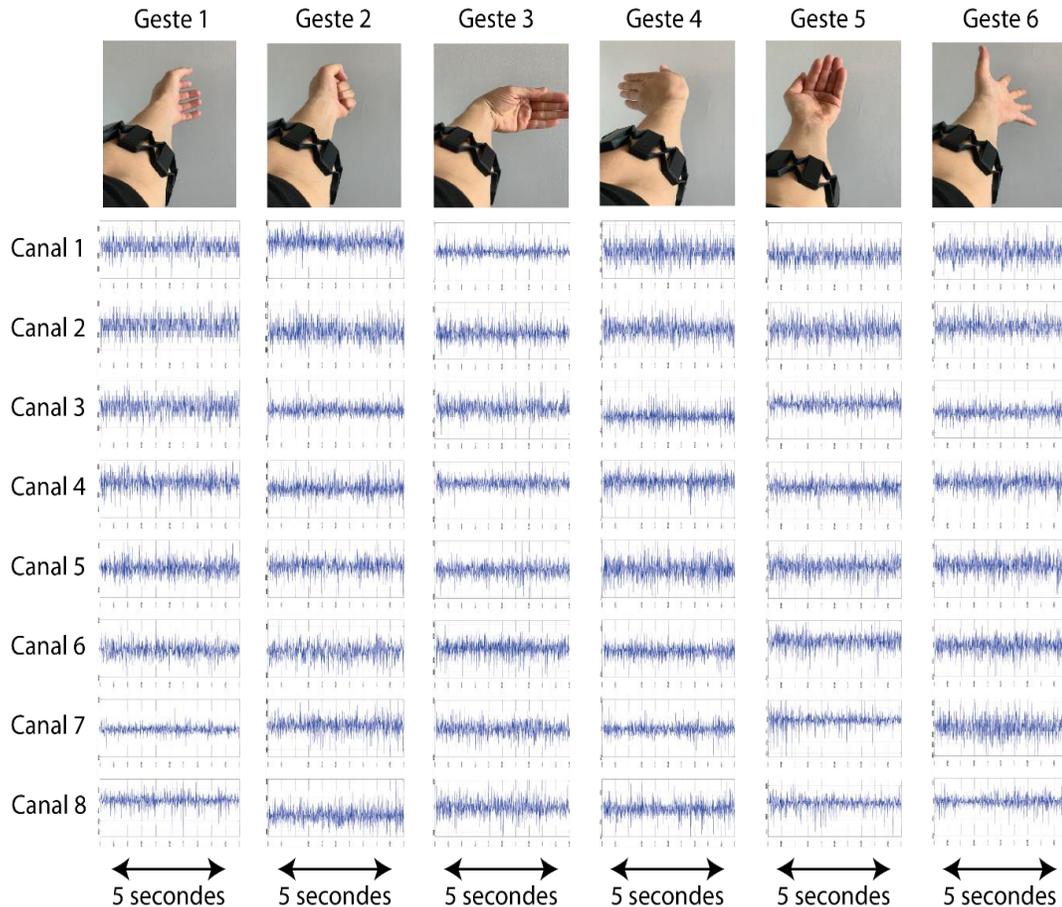


Figure 4. 4 Illustrations des gestes effectués et les signaux EMG associés dans les 8 canaux.

#### 4.4 PRETRAITEMENT ET EXTRACTION DES CARACTERISTIQUES

Tout d'abord, nous devons préparer nos données pour la classification. C'est-à-dire extraire des caractéristiques représentatives des signaux EMG et les classes cibles d'apprentissage qui seront transmises à un algorithme d'apprentissage machine. Ainsi les signaux EMG du même geste de la main a été rassemblés en un seul (voir Figure 4. 5). Chaque geste (classe) contient 100000 échantillons par canal, correspondant à  $(5 s \times 200 Hz \times 10 \text{ répétitions} \times 10 \text{ personnes})$ .

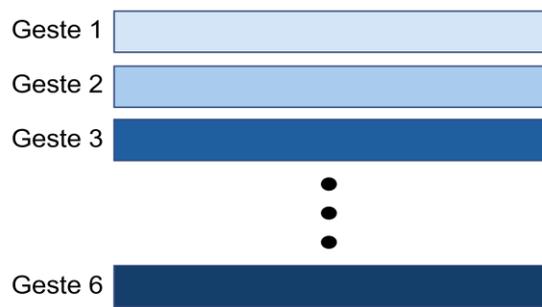


Figure 4. 5 Représentation de la préparation de la base de données.

##### 4.4.1 PRETRAITEMENT ET D'EXTRACTION DES CARACTERISTIQUES

Pour un geste (classe de la main), la Figure 4. 6 illustre la segmentation et le processus d'extraction des caractéristiques des signaux EMG. Lors de l'acquisition d'un geste (classe) d'une personne, un enregistrement de 5 secondes est effectué sur 8 canaux, à une fréquence de  $200 Hz$ , ce qui donne un total de  $5s \times 200Hz = 1000$  échantillons pour chaque canal. Cette acquisition se traduit par une matrice de dimensions  $8 \times 1000$  échantillons. Si les signaux sont segmentés avec une fenêtre de 50 échantillons ( $250 ms$ ) et un chevauchement de 50 %, on obtient 39 segments par canal. Le remplacement de chaque segment par ses caractéristiques conduit à une matrice de dimensions  $8 \times 39$ . L'extraction d'une seule caractéristique par segment, RMS par exemple, produit une matrice de caractéristiques de

taille  $8 \times 39$ . Si on extrait trois caractéristiques (RMS, MAV et MFL) par segment, on obtient une matrice de caractéristiques de taille  $24 \times 39$ .

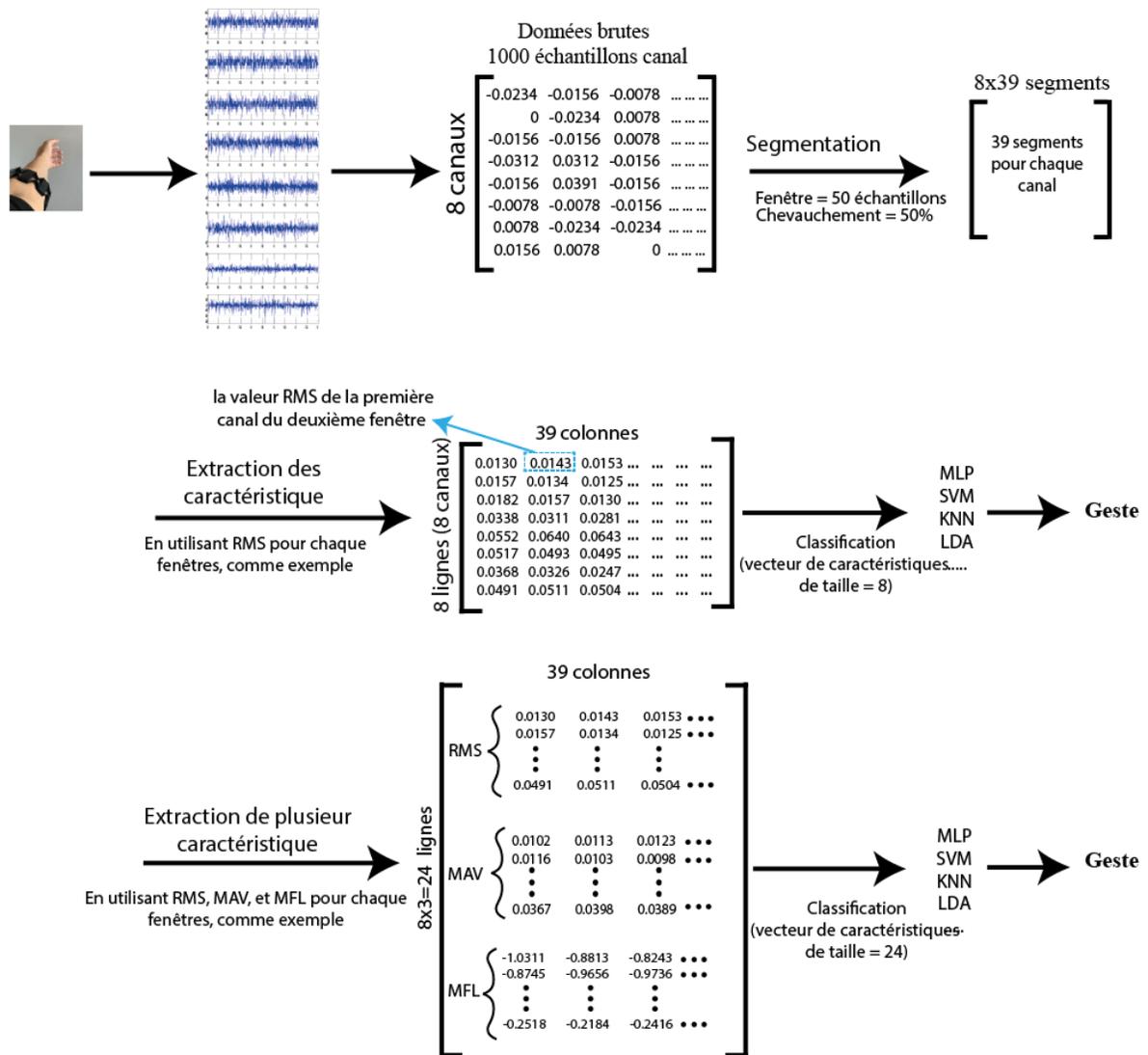


Figure 4. 6 Illustration du processus de fenêtrage et d'extraction des caractéristiques

#### 4.4.2 VALIDATION CROISEE

La validation croisée *k-fold* est une technique utilisée en apprentissage automatique pour évaluer les performances d'un modèle de classification sur toute la base de données, généralement de taille limitée. Elle consiste à diviser les données disponibles en *k* sous-

ensembles de taille égale, ou plis (*fold*), à entraîner le modèle sur  $k-1$  *fold* et à le tester sur le *fold* restant. Ce processus est répété  $k$  fois, chaque pli étant utilisé une fois comme ensemble de test. Les résultats sont ensuite combinés pour obtenir une estimation plus robuste des performances du modèle. La validation croisée à  $k$  *fold* est couramment utilisée pour ajuster les hyperparamètres et tester toute la base de données.

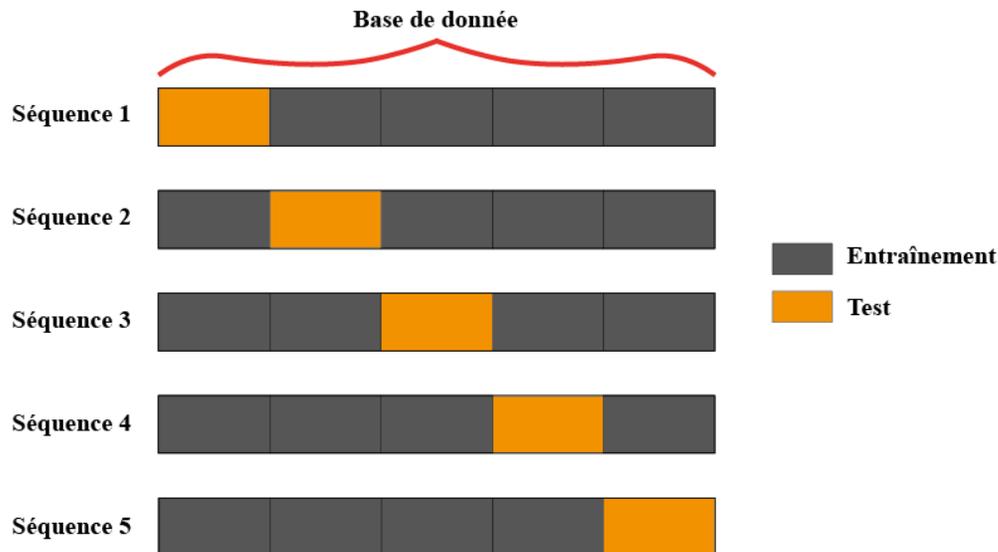


Figure 4. 7 Représentation d'un exemple illustratif de la validation croisée pour  $k = 5$ .

Admettons que nous disposions d'un ensemble de données de 100 observations. Tout d'abord, nous mélangeons les données pour garantir le caractère aléatoire et une distribution représentative. Ensuite, nous les divisons en 5 *fold*s (Figure 4. 7), chacun contenant 20 observations. Pour chacune de 5 itérations (séquence), nous entraînons notre modèle sur une combinaison de quatre *fold*s tout en utilisant un *fold* pour le test. Cette méthode garantit que chaque observation fait partie de l'ensemble de test exactement une fois. Après chaque itération, nous évaluons les performances de notre modèle à l'aide d'une mesure d'évaluation choisie. Ce processus rigoureux nous permet d'évaluer la capacité de notre modèle à s'adapter à différents sous-ensembles de données, en nous aidant à identifier les problèmes potentiels tels que le surajustement ou le sous-ajustement et en fournissant une estimation complète de ses performances globales.

#### 4.4.3 CHOIX DE LA METHODE D'EXTRACTION DES CARACTERISTIQUES

Afin de choisir la longueur de fenêtre et le taux de chevauchement qui donnent la meilleure précision de classification, ainsi que l'extraction de caractéristiques à utiliser, nous avons d'abord testé quelques méthodes d'extraction de caractéristique avec une longueur de fenêtre de 240 ms et un chevauchement de 50%. Ensuite, nous avons fait une combinaison de 3 méthodes d'extraction de caractéristiques qui ont donné la meilleure exactitude de la classification. Après avoir choisi ces techniques d'extractions de caractéristiques, nous avons fait varier la longueur de la fenêtre et le pourcentage de chevauchement.

Une fenêtre de 240 ms contient 48 échantillons, avec un chevauchement de 50 %, ce qui donne un total de 4165 segments par geste et par canal dans toute la base de données.

$$\left[ \frac{100000}{48} \right] + \left[ \frac{100000 - (1 + 48 \times 50\%) + 1}{48} \right] = 2082 + 2083 = 4165.$$

où les crochets carrés représentent la partie entière du nombre qui se trouve entre eux.

À cette étape de test, nous avons utilisé la méthode de validation croisée à 10-*fold*. Les paramètres choisis pour les algorithmes de classification utilisés sont les suivants : pour le k-NN, le nombre de voisins  $k = 3$  et une métrique Euclidienne ; pour le SVM, un noyau polynomial d'ordre 3 ; pour le MLP, ayant une couche cachée de 40 neurones, utilise la tangente hyperbolique comme fonction d'activation et la Softmax en sortie.

D'après les résultats du Tableau 4. 1, nous pouvons voir que MFL, RMS, MAV, SD et WL sont les caractéristiques qui ont donné la meilleure exactitude de classification. Par conséquent, une combinaison de 3 caractéristiques du domaine temporel (MFL, RMS, et MAV) sera utilisée dans l'étape l'extraction de caractéristiques du système proposé.

Tableau 4. 1 Effet du choix du technique de caractérisation sur les performances de classification.

Technique de caractérisation	Exactitude %				
	SVM	k-NN	LDA		MLP
			Linéaire	Quadratique	
<b>RMS</b>	<b>98.47</b>	<b>98.96</b>	<b>78.77</b>	86.68	<b>94.57</b>
<b>MAV</b>	98.25	98.81	78.26	<b>87.10</b>	<b>93.25</b>
<b>SD</b>	<b>98.60</b>	<b>99.08</b>	<b>78.99</b>	<b>86.83</b>	88.82
<b>SSC</b>	66.61	58.78	59.58	64.52	67.08
<b>ZC</b>	78.58	73.61	67.64	74.53	80.21
<b>DASDV</b>	98.32	98.79	78.78	86.51	90.42
<b>WL</b>	98.31	98.80	78.52	86.97	92.13
<b>MMAV</b>	98.04	98.65	77.95	86.84	92.05
<b>MMAV2</b>	95.40	95.35	74.90	83.74	86.46
<b>MFL</b>	<b>98.94</b>	<b>99.12</b>	<b>83.93</b>	<b>93.67</b>	<b>98.22</b>

#### 4.4.4 CHOIX DE LA LONGUEUR DE FENETRE ET DU TAUX DE CHEVAUCHEMENT

Par la suite, nous allons étudier l'effet de la longueur de la fenêtre et du pourcentage de recouvrement des fenêtres. Pour ce faire, nous avons modifié les valeurs de la longueur de la fenêtre de 120 ms à 300 ms et du taux de chevauchement de 20 % à 80 %. L'extraction des caractéristiques utilisée est la combinaison de (MFL, RMS et MAV). Nous avons utilisé un classificateur k-NN ayant les 3 plus proches voisins. De plus, pour la validation croisée, nous avons pris des *k-folds* égaux à 10-*fold*. Le Tableau 4. 2 ci-dessous montre que l'exactitude de classification tend à augmenter avec la taille des fenêtres. Cela s'explique par le fait qu'une fenêtre plus longue capture davantage d'informations sur le signal. Toutefois, l'utilisation de fenêtres plus longues peut s'avérer désavantageuse en raison du temps qu'elles nécessitent, qui influe sur la vitesse de classification, en particulier dans les implémentations en ligne. De même, l'exactitude s'améliore avec des taux de chevauchement plus élevés.

Tableau 4. 2 Étude de l'effet de variation de la taille des fenêtres et du taux de chevauchement sur l'exactitude de la classification.

		<b>EXACTITUDE (%)</b>						
<b>FENÊTRE</b>	<b>(MS)</b>	<b>Chevauchement (%)</b>						
		20	30	40	50	60	70	80
<b>120</b>		97.694	97.815	97.951	97.942	98.200	98.581	98.969
<b>130</b>		97.851	97.963	98.154	98.214	98.428	98.689	99.133
<b>140</b>		98.126	98.136	98.254	98.411	98.623	98.896	99.126
<b>150</b>		98.224	98.362	98.497	98.530	98.718	98.980	99.292
<b>160</b>		98.357	98.507	98.518	98.682	98.858	99.035	99.389
<b>170</b>		98.506	98.440	98.703	98.790	98.922	99.193	99.429
<b>180</b>		98.622	98.658	98.808	98.980	99.083	99.241	99.561
<b>190</b>		98.800	98.789	98.980	98.967	99.152	99.375	99.511
<b>200</b>		98.810	98.950	98.896	99.080	99.250	99.412	99.593
<b>210</b>		98.855	98.975	98.950	99.181	99.249	99.452	99.644
<b>220</b>		99.078	99.085	99.120	99.211	99.307	99.482	99.677
<b>230</b>		98.951	99.045	99.108	99.241	99.382	99.449	99.667
<b>240</b>		99.157	99.099	99.188	99.212	99.373	99.545	99.705
<b>250</b>		99.060	99.212	99.210	99.400	99.430	99.560	99.738
<b>260</b>		99.223	99.124	99.261	99.345	99.419	99.584	99.782
<b>270</b>		99.305	99.278	99.280	99.365	99.479	99.594	99.736
<b>280</b>		99.175	99.200	99.286	99.337	99.527	99.634	99.763
<b>290</b>		99.187	99.337	99.381	99.405	99.563	99.643	99.770
<b>300</b>		99.392	99.370	99.340	99.435	99.568	99.670	99.800

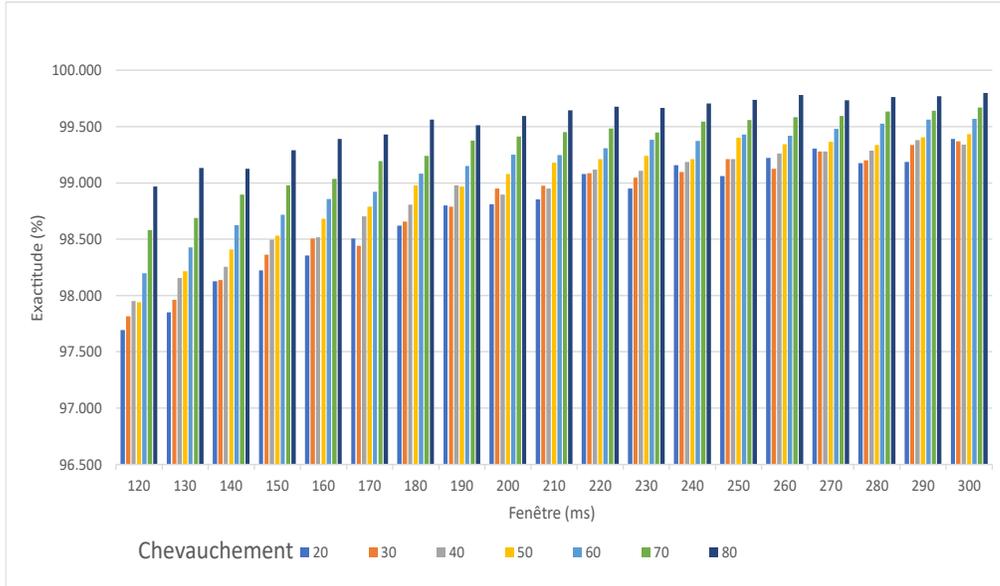


Figure 4. 8 Variation de l'exactitude en fonction de la taille de la fenêtre et de chevauchement en utilisant k-NN

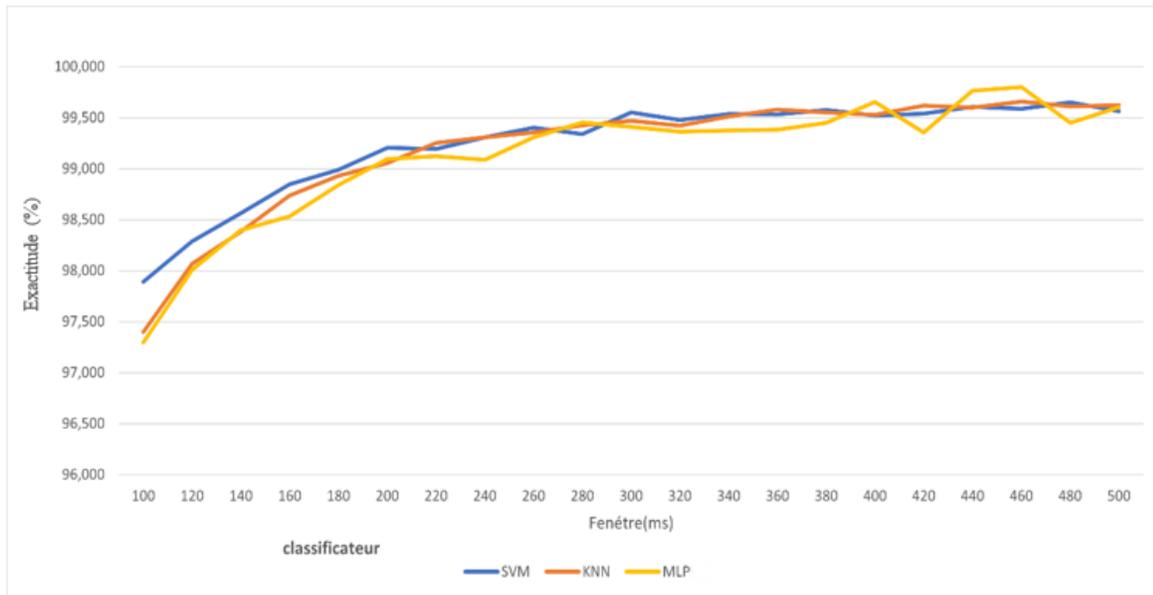


Figure 4. 9 Variation de l'exactitude en fonction de la taille des fenêtres en utilisant classificateur SVM, k-NN et MLP.

La Figure 4. 8, présentée sous forme de diagramme à barres, illustre le pourcentage d'exactitude de l'algorithme k-NN en fonction de différents taux de chevauchement et de tailles de fenêtre en millisecondes. De plus, la Figure 4. 9 illustre la variation de l'exactitude des trois algorithmes de classification : SVM, k-NN et MLP en fonction de la taille de la fenêtre (en millisecondes).

D'après le Tableau 4. 2 et les Figures 4. 8 et 4. 9, nous pouvons observer qu'en augmentant la taille de la fenêtre, l'exactitude augmente, et vice versa. Il en va de même pour le pourcentage de chevauchement. Pour notre travail, nous avons choisi une longueur de fenêtre de 250 ms au lieu de 300 ms car l'augmentation de la longueur de la fenêtre entraîne une augmentation du temps de latence, aussi, une fenêtre de 250 ms offre une très bonne exactitude. Avec un chevauchement de fenêtre de 50 %, ainsi que la technique d'extraction de caractéristiques sélectionnée précédemment. Avec une taille de fenêtre de 50 échantillons et un chevauchement de 50 %, on obtient 3999 segments par geste et par canal  $\left[ \frac{100000}{50} \right] + \left[ \frac{100000 - (1 + 50 \times 50 \%) + 1}{50} \right] = 2000 + 1999 = 3999$ .

## 4.5 RESULTATS DE LA CLASSIFICATION

Dans cette partie, nous allons procéder à la classification en utilisant 4 algorithmes : SVM, k-NN, LDA et MLP. Dans les trois premiers algorithmes mentionnés, nous avons appliqué la validation croisée *k-fold* à 10 plis. Pour le dernier algorithme, la validation croisée a été ajustée pour inclure les données d'apprentissage, de validation et de test.

### 4.5.1 MACHINE A VECTEUR DE SUPPORT (SVM)

Un classificateur SVM a été formé sur notre base de données avec une validation croisée 10 *folds*, en utilisant un noyau polynomial de degré 3 et en appliquant l'algorithme *one-vs-one*. La matrice de confusion correspondante est présentée à la Figure 4. 10. L'exactitude obtenue est de 99,34 %.

Dans notre ensemble de données, il existe six classes distinctes, chacune représentant précisément  $\frac{1}{6} = 16,6\%$  du total des observations. Pour commencer, prenons l'exemple de la classe 1. La machine à vecteur de support (SVM) a effectué des classifications précises pour la classe 1, avec 3955 observations correctement identifiées comme appartenant à cette classe. Cependant, il a attribué 1 observation à la classe 2 et 43 observations à la classe 6. Le même principe s'applique aux autres classes.

**Matrice de confusion**

Classe de sortie	1	3955 16.5%	0 0.0%	0 0.0%	1 0.0%	0 0.0%	31 0.1%	99.2% 0.8%
	2	1 0.0%	3994 16.6%	0 0.0%	0 0.0%	0 0.0%	2 0.0%	99.9% 0.1%
	3	0 0.0%	1 0.0%	3968 16.5%	0 0.0%	0 0.0%	31 0.1%	99.2% 0.8%
	4	0 0.0%	0 0.0%	0 0.0%	3989 16.6%	3 0.0%	0 0.0%	99.9% 0.1%
	5	0 0.0%	2 0.0%	1 0.0%	7 0.0%	3994 16.6%	0 0.0%	99.8% 0.2%
	6	43 0.2%	2 0.0%	30 0.1%	2 0.0%	2 0.0%	3935 16.4%	98.0% 2.0%
		98.9% 1.1%	99.9% 0.1%	99.2% 0.8%	99.7% 0.3%	99.9% 0.1%	98.4% 1.6%	99.3% 0.7%
	Classe cible							
	1	2	3	4	5	6		

Figure 4. 10 Matrice de confusion pour SVM de noyau polynomial d'ordre 3.

## 4.5.2 K-PLUS PROCHES VOISINS (K-NN)

### 4.5.2.1 Choisir le nombre de voisins

Le Tableau 4. 3 ci-dessous montre l'exactitude de la classification en fonction du nombre de voisins.

Tableau 4. 3 Exactitude de la classification à différents nombres de voisins k.

<b>k</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>
<b>Exactitude (%)</b>	99.32	99.11	99.36	99.25	99.30	99.30	99.29	99.23

Après une analyse approfondie, nous avons constaté que l'utilisation de  $k=3$  permettait d'obtenir la meilleure exactitude de classification. Le modèle a atteint une performance impressionnante de 99.36% en utilisant 3 voisins.

### 4.5.2.2 Résultats de classification

Nous avons utilisé la validation croisée à 10 *fold* pour entraîner un classificateur k-NN sur notre jeu de données. Pour ce classificateur, nous avons utilisé  $k = 3$  et une métrique Euclidienne. La matrice de confusion est présentée à la Figure 4. 11. L'exactitude obtenue est de 99,36%.

**Matrice de confusion**

<b>Classe de sortie</b>	1	3966 16.5%	0 0.0%	0 0.0%	1 0.0%	0 0.0%	36 0.2%	99.1% 0.9%
	2	1 0.0%	3996 16.7%	1 0.0%	0 0.0%	1 0.0%	1 0.0%	99.9% 0.1%
	3	0 0.0%	0 0.0%	3977 16.6%	0 0.0%	1 0.0%	45 0.2%	98.9% 1.1%
	4	0 0.0%	0 0.0%	0 0.0%	3990 16.6%	2 0.0%	0 0.0%	99.9% 0.1%
	5	0 0.0%	1 0.0%	0 0.0%	5 0.0%	3995 16.6%	1 0.0%	99.8% 0.2%
	6	32 0.1%	2 0.0%	21 0.1%	3 0.0%	0 0.0%	3916 16.3%	98.5% 1.5%
			99.2% 0.8%	99.9% 0.1%	99.4% 0.6%	99.8% 0.2%	99.9% 0.1%	97.9% 2.1%
		<b>Classe cible</b>						
		1	2	3	4	5	6	

Figure 4. 11 Matrice de confusion pour k-NN à 3 plus proches voisins.

### 4.5.3 ANALYSE DISCRIMINANTE LINEAIRE (LDA)

L'exactitude de la classification obtenue par le classificateur LDA avec un hyperplan linéaire était de 86,83 %. Cela est dû au fait que nos données ne sont pas linéairement séparables. En utilisant la projection des données sur un hyperplan quadratique au lieu d'un hyperplan linéaire, nous obtenons une exactitude de 90,06 %. La matrice de confusion est présentée dans la Figure 4. 12.

**Matrice de confusion**

Classe de sortie	1	3939 16.4%	11 0.0%	143 0.6%	3 0.0%	1 0.0%	1602 6.7%	69.1% 30.9%
	2	8 0.0%	3958 16.5%	25 0.1%	0 0.0%	0 0.0%	12 0.1%	98.9% 1.1%
	3	5 0.0%	7 0.0%	3740 15.6%	0 0.0%	5 0.0%	266 1.1%	93.0% 7.0%
	4	0 0.0%	1 0.0%	0 0.0%	3906 16.3%	21 0.1%	0 0.0%	99.4% 0.6%
	5	4 0.0%	0 0.0%	3 0.0%	90 0.4%	3971 16.5%	24 0.1%	97.0% 3.0%
	6	43 0.2%	22 0.1%	88 0.4%	0 0.0%	1 0.0%	2095 8.7%	93.2% 6.8%
			98.5% 1.5%	99.0% 1.0%	93.5% 6.5%	97.7% 2.3%	99.3% 0.7%	52.4% 47.6%
		Classe cible						
		1	2	3	4	5	6	

Figure 4. 12 Matrice de confusion pour LDA à hyperplan quadratique.

#### 4.5.4 PERCEPTRON MULTICOUCHE (MLP)

##### 4.5.4.1 Validation croisée sur MLP

Pour des raisons de fiabilité, nous avons entraîné le MLP sur des sous-ensembles différents d'entraînement, de validation et de test. Les données sont divisées en 10 *folds* comme pour les trois autres classificateurs. Cependant, pour chaque séquence, 8 *folds* sont utilisés

pour l'entraînement, un pour la validation, et un pour le test. La Figure 4. 13 montre comment les données sont réparties.

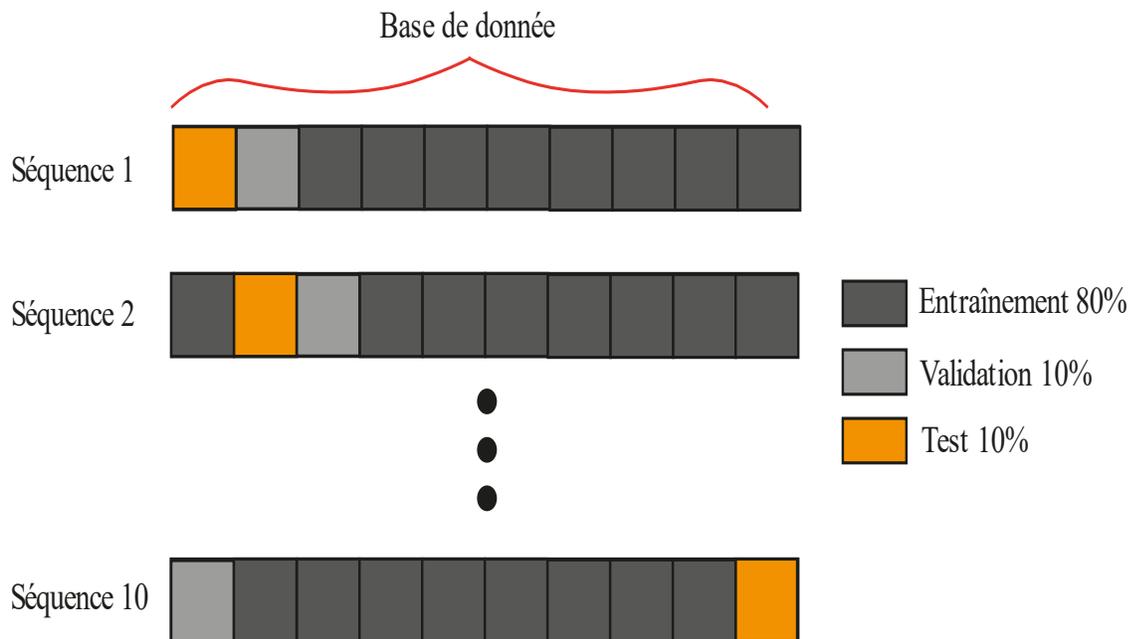


Figure 4. 13 Représentation de processus Cross-Validation  $k = 10$  avec le classificateur MLP.

#### 4.5.4.2 Effets du nombre de couches cachées sur les performances de classification

Dans cette section, nous allons évaluer les performances de classification du réseau MLP en fonction du nombre de couches cachées et du nombre de neurones par couches cachées afin de trouver la meilleure topologie du réseau. Le Tableau 4. 4 résume l'exactitude moyenne de la classification obtenue pour chaque combinaison à partir de 50 tests. Nous avons donc divisé les données en 80% d'entraînement, 10% de validation et 10% de test. De plus, le processus est répété 50 car les poids de connexion sont initialisés aléatoirement. Pour les fonctions d'activation, nous avons choisi la tangente hyperbolique dans toutes les couches cachées, tandis que la fonction Softmax a été utilisée pour la couche de sortie.

Tableau 4. 4 Exactitude de la classification avec différents nombres de couches cachées et de perceptrons.

<b>Nombre de couches cachées</b>	<b>Nombres de perceptrons</b>	<b>Exactitude (%)± Écart-type</b>
<b>1</b>	10	96.42 ± 0.57
<b>1</b>	20	97.84 ± 0.33
<b>1</b>	30	98.29 ± 0.29
<b>1</b>	40	98.47 ± 0.25
<b>1</b>	50	98.52 ± 0.28
<b>2</b>	10, 10	97.02 ± 0.51
<b>2</b>	20, 20	98.40 ± 0.27
<b>2</b>	30, 30	98.77 ± 0.22
<b>2</b>	40, 40	98.97 ± 0.17

Ces résultats mettent en évidence l'impact de l'ajustement du nombre de couches cachées et du nombre de perceptrons sur les performances de classification du modèle. Notamment, les modèles comportant plusieurs couches cachées ont tendance à être plus performants que ceux comportant une seule couche. En outre, l'augmentation du nombre de perceptrons dans chaque couche entraîne généralement une amélioration de sa performance de classification.

Sur la base de ces résultats, la combinaison de deux couches cachées avec 40 perceptrons dans chaque couche a obtenu la plus grande exactitude, soit 98.97 %. Ces informations précieuses peuvent nous aider à affiner l'architecture de nos réseaux neuronaux pour obtenir des performances optimales dans notre tâche.

#### 4.5.4.3 Résultats de classification

Nous avons formé un classificateur MLP sur notre base de données. En présentant le processus de la validation croisée *10-folds*. Nous avons divisé les données de chaque séquence en 80% d'entraînement, 10% de validation et 10% de test. De plus, chaque séquence est répétée 50 pour réduire le biais causé par l'initialisation aléatoires des poids. Les hyperparamètres suivants ont donné les meilleures performances sur l'ensemble de validation

- Nombre de couches cachées : 2
- Nombre d'entrées : 24
- Nombre de neurones dans chaque couche cachée : 40
- Fonction d'activation des couches cachées : Tangente hyperbolique sigmoïde dans les deux couches cachées.
- Fonction d'activation de la couche de sortie : Softmax

La matrice de confusion pour les résultats obtenu par MLP sur toute la base de données est présentée dans la Figure 4. 14. La performance totale de l'application de validation croisées pour MLP est de 98.997%.

**Matrice de confusion**

<b>Classe de sortie</b>	1	3920 16.3%	0 0.0%	0 0.0%	0 0.0%	1 0.0%	73 0.3%	98.1% 1.9%
	2	0 0.0%	3997 16.7%	0 0.0%	0 0.0%	0 0.0%	2 0.0%	99.9% 0.1%
	3	0 0.0%	0 0.0%	3967 16.5%	0 0.0%	1 0.0%	35 0.1%	99.1% 0.9%
	4	0 0.0%	0 0.0%	0 0.0%	3989 16.6%	7 0.0%	0 0.0%	99.8% 0.2%
	5	0 0.0%	0 0.0%	0 0.0%	8 0.0%	3990 16.6%	0 0.0%	99.8% 0.2%
	6	79 0.3%	2 0.0%	32 0.1%	2 0.0%	0 0.0%	3889 16.2%	97.1% 2.9%
			98.0% 2.0%	99.9% 0.1%	99.2% 0.8%	99.7% 0.3%	99.8% 0.2%	97.2% 2.8%
		1	2	3	4	5	6	
		<b>Classe cible</b>						

Figure 4. 14 Matrice de confusion pour MLP.

#### 4.6 COMPARAISON DES RESULTATS

Après avoir évalué les différents algorithmes d'apprentissage automatique, il a été observé que le modèle LDA était le moins performant (Tableau 4. 5). À l'inverse, les modèles k-NN, SVM et MLP ont donné des résultats exceptionnels, avec une exactitude supérieure à 99 %. En raison de la complexité des modèles SVM et MLP, nous avons conclu que k-NN était l'algorithme d'apprentissage automatique le mieux adapté à notre projet.

Tableau 4. 5 Comparaison des modèles proposés dans cette recherche.

<b>Méthode</b>	<b>SVM</b>	<b>k-NN</b>	<b>LDA</b>	<b>MLP</b>
<b>Exactitude</b>	99.34%	99.36%	90.06%	98.97%

#### **4.7 COMPARAISON DES RESULTATS AVEC D'AUTRE TRAVAUX UTILISANT LE MEME CAPTEUR MYO GESTURE ARMBAND**

Le Tableau 4. 6 présente une comparaison des résultats hors ligne de cette étude avec ceux d'autres recherches. Chaque étude dans le Tableau 4. 6 présente des aspects de la méthode expérimentale utilisée, notamment le nombre de gestes, la taille de la fenêtre, l'extraction des caractéristiques, le type de classificateur utilisé, et l'exactitude de la classification obtenue.

Ces études utilisent le même capteur, le brassard Myo Gesture Armband, pour acquérir les signaux EMG à la même fréquence d'échantillonnage de 200Hz. Cependant, elles diffèrent par les techniques d'extraction des caractéristiques et de classification, ainsi que par la population ayant servi à construire chacune des bases de données. Toutefois, notre approche présente les meilleures performances de classification.

Tableau 4. 6 Performance de classification obtenue par des chercheurs utilisant le brassard Myo Gesture Armband avec une fréquence d'échantillonnage de 200Hz.

Étude	Nombre de Gestes	Taille de la fenêtre	Extraction des caractéristiques	Classificateur	Exactitude
(Krishnan <i>et al.</i> , 2017)	5	50	SSI, Max. frequency, Min. frequency, Mean Power, Mean Frequency,	SVM	92.40
(Tepe & Demir, 2021)	7	60	DASDV, WAMP, AAC	SVM	98.38
(Wahid <i>et al.</i> , 2018)	3	60	MAV, ZC, WL, SSC	RF	96.40
(Hassan <i>et al.</i> , 2020)	7	48	MAV, RMS, WL, AR(4), ZC, SSC	SVM	95.26
				LDA	92.58
				k-NN	86.41
<b>Cette étude</b>	6	50	RMS, MFL, MAV	k-NN	99.37
				SVM	99.36
				LDA	90.06
				MLP	98.97

#### 4.8 COMPARAISON DES RESULTATS DE CLASSIFICATION AVEC CEUX DES AUTRES CHERCHEURS EN UTILISANT LA BASE DE DONNES UCI

Dans cette sous-section, nous allons comparer les performances de classification de nos algorithmes avec des systèmes de classification existants en utilisant la même base de données UCI constituée de signaux EMG captés par le brassard Myo Gesture Armband sur 36 sujets effectuant 6 mouvements différents de la main. Pour évaluer et comparer les performances, nous utiliserons les quatre algorithmes de classification suivants : SVM, k-NN, LDA et MLP proposé dans ce mémoire.

Tableau 4. 7 Performances de classification obtenues par des études utilisant la base de données de l'UCI.

Étude	Sujet et geste	Extraction des caractéristique	Classificateur	L'exactitude (%)
(Cao <i>et al.</i> , 2021)	Les 36 sujets Gestes 3-6	RMS+MAV+IAV+WL+AR	AMPSO-SVM	97.50
			PSO-SVM	94.63
			GS-SVM	90.93
(Rabbat, 2022)	Premier sujet, geste 1-6	RMS	Bagging classifieur	87.3
			RF	85.6
			Régression	63.5
(Mora-Rubio <i>et al.</i> , 2022)	Les 36 sujets Gestes 1-6	Aucune extraction de caractéristiques n'a été utilisée.	k-NN	95.06
			LR	22.25
			GNB	63.27
			MLP	72.68
			RF	96.25
			DT	94.59
			ANN	96.09
CNN	95.48			
<b>Dans cette étude</b>	Les 36 sujets Gestes 1-6	MFL+RMS+MAV	k-NN	96.62
			LDA	87.01
			MLP	96.26
			SVM	97.54

Le Tableau 4. 7 donne des indications comparatives sur les différentes approches de la classification des gestes, en mettant en évidence l'efficacité des différentes techniques d'extraction des caractéristiques et des classificateurs. Cette étude présente l'une des exactitudes les plus élevées parmi les recherches citées, ce qui indique une approche potentiellement efficace de la reconnaissance des gestes de la main.

Dans cette étude comparative, nous avons observé une petite diminution de l'exactitude de classification lors de l'évaluation de la base de données UCI de 36 sujets, par rapport à notre précédente base de données de 10 sujets. Cette diminution de l'exactitude pourrait être attribuée au fait qu'un plus grand nombre de sujets introduit une plus grande diversité de caractéristiques. Cette variabilité peut également inclure des différences dans les schémas d'activation musculaire (Benatti *et al.*, 2015) et des variations dans l'impédance de la peau (Saelim *et al.*, 2019). Ces facteurs contribuent collectivement à augmenter la variabilité des données.

Nous avons calculé l'exactitude en utilisant différents nombres de sujets. Les résultats sont présentés dans le Tableau 4. 8. Nous rappelons que le système k-NN a permis d'obtenir une exactitude de 99.36% avec notre base de données à 10 sujets. Comme vous pouvez le voir, lorsque le nombre de sujets augmente, l'exactitude diminue.

Tableau 4. 8 Variation de l'exactitude avec l'algorithme k-NN en fonction du nombre de sujets de la base de données UCI, en sélectionnant les premiers sujets de la base de données.

<b>Nombre de sujets</b>	<b>Exactitude (%)</b>
<b>10</b>	99.52
<b>15</b>	99.04
<b>20</b>	98.61
<b>25</b>	98.00
<b>30</b>	97.64
<b>36</b>	96.62

#### 4.9 RECONNAISSANCE DES GESTES DANS LES CONDITIONS REELLES

L'évaluation du système de reconnaissance des gestes de la main par analyse des signaux EMG dans des conditions réelles est une étape importante dans l'intégration du système à des applications pratiques. Par conséquent, une implémentation est réalisée à l'aide de brassard Myo Gesture Armband et l'environnement MATLAB (Figure 4. 15). Le système a été développé pour fonctionner dans des conditions réelles, ce qui permet une reconnaissance quasi-immédiate et précise des gestes de la main. Le brassard Myo Gesture Armband a été utilisé pour collecter les signaux EMG de l'avant-bras de l'utilisateur, les signaux ont ensuite été traités et classés en temps quasi réel à l'aide d'algorithmes d'apprentissage machine dans l'entraînement MATLAB. Les performances dans les conditions réelle du système ont été évaluées en termes de temps de traitement et de délai d'attente, qui sont des paramètres importants pour s'assurer que le système est réactif et précis. Nos résultats ont montré que le système avait un temps de traitement moyen de 20 ms (le système prend en moyenne 20 ms pour traiter chaque fenêtre de données qu'il reçoit) et une durée de latence de 310 ms, ce qui indique que le système est capable de fonctionner en temps réel.

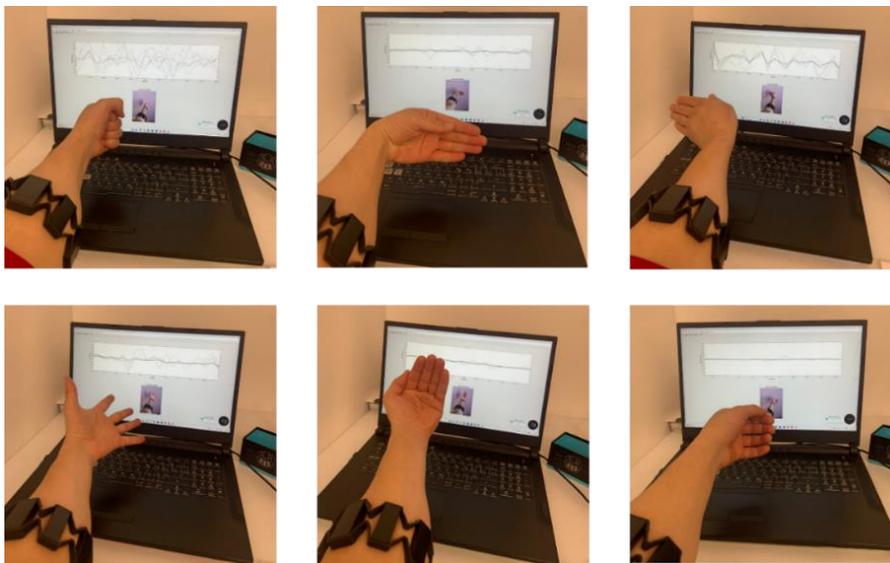


Figure 4. 15 Représentation du résultat dans les conditions réelles d'utilisation.

## CONCLUSION GÉNÉRALE

Dans ce mémoire, nous avons montré les différentes étapes de conception et de réalisation d'un système de classification des gestes de la main basée sur les signaux électromyographiques (EMG) en utilisant des modèles d'apprentissage machine. Nous avons débuté la réalisation du système en explorant les signaux EMG et les méthodes couramment utilisées pour acquérir ces signaux. Nous avons également discuté de diverses bases de données des signaux EMG populaires relatives aux gestes de la main. Pour exploiter efficacement les informations contenues dans ces signaux, nous avons appliqué la technique du fenêtrage et extrait des caractéristiques pertinentes à partir de chaque fenêtre.

Ensuite, nous nous sommes concentrés sur l'apprentissage machine et avons examiné en détail quatre algorithmes spécifiques : SVM, MLP, k-NN et LDA. Nous avons décrit leurs fonctionnements et leurs paramètres afin de comprendre comment ils peuvent être appliqués à la reconnaissance des gestes de la main basée sur l'EMG. Nous avons réalisé une étude comparative en utilisant notre propre base de données ainsi qu'une base de données disponible dans le référentiel d'apprentissage machine de l'UCI (*University of California at Irvine*), ce qui nous a permis d'évaluer la capacité de généralisation des algorithmes.

De plus, nous avons mis en place une plateforme de test dans des conditions réelles en connectant le brassard Myo Gesture Armband et l'environnement MATLAB, permettant ainsi d'évaluer leurs performances dans des scénarios d'applications réelles. Les résultats obtenus ont démontré que l'utilisation d'algorithmes d'apprentissage machine pour la reconnaissance des gestes de la main basée sur les signaux EMG est une approche prometteuse, offrant une précision élevée dans la reconnaissance des gestes en temps réel. Cette étude a mis en évidence le potentiel des algorithmes d'apprentissage machine pour améliorer l'exactitude de la reconnaissance des gestes de la main par rapport à d'autres recherches, ouvrant ainsi la voie au développement d'interfaces homme-machine innovantes et efficaces.

Cependant, il reste des défis à surmonter, particulièrement en ce qui touche à l'acquisition des signaux EMG, l'extraction des caractéristiques et l'optimisation des algorithmes pour améliorer leurs performances dans des conditions réelles. Des recherches supplémentaires sont nécessaires pour perfectionner l'approche de reconnaissance des gestes de la main basée sur les signaux EMG.

## RÉFÉRENCES BIBLIOGRAPHIQUES

- Aggarwal, C. C. (2018). *Neural Networks and Deep Learning*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-94463-0>
- Ahsan, M. R., Ibrahimy, M. I., & Khalifa, O. O. (2011). Electromyography (EMG) signal based hand gesture recognition using artificial neural network (ANN). *2011 4th International Conference on Mechatronics (ICOM)*, Kuala Lumpur, Malaysia, 2011, pp. 1-6. <https://doi.org/10.1109/ICOM.2011.5937135>
- Alao, A. (2018). How muscles of the fore arm brings about movement. steemit. <https://steemit.com/promo-steem/@alaoabduhaleem/how-muscles-of-the-fore-arm-brings-about-movement>. Visité le 02 mai 2023.
- Amma, C., Krings, T., Böer, J., & Schultz, T. (2015). Advancing muscle-computer interfaces with high-density electromyography. *33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 929-938. <https://doi.org/10.1145/2702123.2702501>
- Andreisek, G., Kislinger, B., Dessouky, R., & Chhabra, A. (2017). MRI of the Intrinsic Muscles of the Hand. *Seminars in musculoskeletal radiology*, 21 (04), pp. 392-402. <https://doi.org/10.1055/s-0037-1604006>
- Artal-Sevil, J. S., Acon, A., Montanes, J. L., & Dominguez, J. A. (2018). Design of a Low-Cost Robotic Arm controlled by Surface EMG Sensors. *2018 XIII Technologies Applied to Electronics Teaching Conference (TAEF)*, La Laguna, Spain, 2018, pp. 1-8. <https://doi.org/10.1109/TAEF.2018.8476126>
- Atzori, M., Gijsberts, A., Castellini, C., Caputo, B., Hager, A. G. M., Elsig, S., Giatsidis, G., Bassetto, F., & Müller, H. (2014). Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Scientific Data*, 1 (1), pp. 1-13. <https://doi.org/10.1038/sdata.2014.53>
- Atzori, M., Gijsberts, A., Heynen, S., Hager, A. G. M., Deriaz, O., Van Der Smagt, P., Castellini, C., Caputo, B., & Muller, H. (2012). Building the Ninapro database: A resource for the biorobotics community. *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, Rome, Italy, 2012, pp. 1258-1265. <https://doi.org/10.1109/BioRob.2012.6290287>
- Bahoura, M., & Park, C. W. (2012). FPGA-implementation of dynamic time delay neural network for power amplifier behavioral modeling. *Analog Integrated Circuits and Signal Processing*, 73(3), pp. 819-828. <https://doi.org/10.1007/s10470-012-9857-0>

- Bahoura, M., & Simard, Y. (2012). Expert Systems with Applications Serial combination of multiple classifiers for automatic blue whale calls recognition. *Expert Systems With Applications*, 39 (11), pp. 9986-9993. <https://doi.org/10.1016/j.eswa.2012.01.156>.
- Baygin, M., Barua, P. D., Dogan, S., Tuncer, T., Key, S., Acharya, U. R., & Cheong, K. H. (2022). A Hand-Modeled Feature Extraction-Based Learning Network to Detect Grasps Using sEMG Signal. *Sensors*, 22 (5), pp. 1-18. <https://doi.org/10.3390/s22052007>
- Benatti, S., Casamassima, F., Milosevic, B., Farella, E., Schönle, P., Fateh, S., Burger, T., Huang, Q., & Benini, L. (2015). A Versatile Embedded Platform for EMG Acquisition and Gesture Recognition. *IEEE Transactions on Biomedical Circuits and Systems*, 9 (5), pp. 620-630. <https://doi.org/10.1109/TBCAS.2015.2476555>
- Bhobhriya, R., Boora, R., Jangra, M., & Dalal, P. (2023). W-NLM: a proficient EMG denoising technique. *International Journal of Information Technology*, 15 (5), pp. 2517–2527. <https://doi.org/10.1007/s41870-023-01324-5>
- Biga, L. M., Dawson, S., Harwell, A., Hopkins, R., Kaufmann, J., LeMaster, M., Matern, P., Morrison-Graham, K., Quick, D., & Runyeon, J. (2020). *Anatomy & Physiology*. OpenStax/Oregon State University.
- Boujelben, O., & Bahoura, M. (2016). FPGA implementation of an automatic wheezes detector based on MFCC and SVM. *2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, Monastir, Tunisia, 2016, pp. 647-650. <https://doi.org/10.1109/ATSIP.2016.7523173>
- Boujelben, O., & Bahoura, M. (2018). Efficient FPGA-based architecture of an automatic wheeze detector using a combination of MFCC and SVM algorithms. *Journal of Systems Architecture*, 88, pp. 54-64. <https://doi.org/10.1016/j.sysarc.2018.05.010>
- Brosset, C. (2018). *Needle EMG-Detection*. vietnamnews.vn. <https://vietnamnews.vn/life-style/424926/electromyogram-explained-in-detail.html>. Visité le 02 mai 2023.
- Cao, L., Zhang, W., Kan, X., & Yao, W. (2021). A Novel Adaptive Mutation PSO Optimized SVM Algorithm for sEMG-Based Gesture Recognition. *Scientific Programming*, 2021, pp. 1-13. <https://doi.org/10.1155/2021/9988823>
- Chithra, P. L., & Aparna, R. (2015). Performance Analysis of Windowing Techniques in Automatic Speech Signal Segmentation. *Indian Journal of Science and Technology*, 8 (29), 1. <https://doi.org/10.17485/ijst/2015/v8i29/83616>
- Colli ro, J. G., & Trejos, A. L. (2022). User-independent hand gesture recognition classification models using sensor fusion. *Sensors (Basel, Switzerland)*, 22 (4), 1321. <https://doi.org/10.3390/s22041321>
- Colligan, L. H. (2010). *The Amazong Human Body. Muscles*.

- Du, Y., Jin, W., Wei, W., Hu, Y., & Geng, W. (2017). Surface EMG-based inter-session gesture recognition enhanced by deep domain adaptation. *Sensors (Switzerland)*, 17 (3), 458. <https://doi.org/10.3390/s17030458>
- Evans, O. G. (2023). *Motor Neuron: Function, Types, And Structure*. simply psychology. <https://www.simplypsychology.org/motor-neuron.html>. Visité le 02 mai 2023.
- Fajardo, J. M., Gomez, O., & Prieto, F. (2021). EMG hand gesture classification using handcrafted and deep features. *Biomedical Signal Processing and Control*, 63, 102210. <https://doi.org/10.1016/j.bspc.2020.102210>
- Ferreira, A., Celeste, W. C., Cheein, F. A., Bastos-Filho, T. F., Sarcinelli-Filho, M., & Carelli, R. (2008). Human-machine interfaces based on EMG and EEG applied to robotic systems. *Journal of NeuroEngineering and Rehabilitation*, 5 (1), pp. 1-15. <https://doi.org/10.1186/1743-0003-5-10>
- Gomez-Gil, J., San-Jose-Gonzalez, I., Nicolas-Alonso, L. F., & Alonso-Garcia, S. (2011). Steering a tractor by means of an EMG-based human-machine interface. *Sensors*, 11 (7), pp. 7110-7126. <https://doi.org/10.3390/s110707110>
- Graupe, D. (2013). *Principles of artificial neural networks* (3rd Edition). World Scientific Publishing Co. Pte. Ltd.
- Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN model-based approach in classification. *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003*, pp. 986-996. [https://doi.org/10.1007/978-3-540-39964-3\\_62](https://doi.org/10.1007/978-3-540-39964-3_62)
- Hassan, H. F., Abou-Loukh, S. J., & Ibraheem, I. K. (2020). Teleoperated robotic arm movement using electromyography signal with wearable Myo armband. *Journal of King Saud University - Engineering Sciences*, 32 (6), pp. 378-387. <https://doi.org/10.1016/j.jksues.2019.05.001>
- Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13 (2), pp. 415-425. <https://doi.org/10.1109/72.991427>
- Hussain, J. S., Al-Khazzar, A., & Raheema, M. N. (2020). Recognition of new gestures using myo armband for myoelectric prosthetic applications. *International Journal of Electrical and Computer Engineering (IJECE)*, 10 (6), 5694. <https://doi.org/10.11591/ijece.v10i6.pp5694-5702>
- Jarmey, C. (2018). *The concise book of muscles*. North Atlantic Books.
- Jarque-Bou, N. J., Sancho-Bru, J. L., & Vergara, M. (2021). A systematic review of EMG applications for the characterization of forearm and hand muscle activity during

activities of daily living: Results, challenges, and open issues. *Sensors*, 21 (9), 30335. <https://doi.org/10.3390/s21093035>

Jia, G., Lam, H., Liao, J., & Wang, R. (2020). Neurocomputing Classification of electromyographic hand gesture signals using machine learning techniques. *Neurocomputing*, 401, pp. 236-248. <https://doi.org/10.1016/j.neucom.2020.03.009>

Journaldesfemmes. (2022). *Muscle squelettique : définition, schéma de ce type de muscle*. Journal des femmes. <https://sante.journaldesfemmes.fr/fiches-anatomie-et-examens/2788161-muscle-squelettique-definition-schema/>. Visité le 02 mai 2023.

Karabulut, D., Ortes, F., Ziya, Y., & Arif, M. (2017). Comparative evaluation of EMG signal features for myoelectric controlled human arm prosthetics. *Biocybernetics and Biomedical Engineering*, 37 (2), pp. 326-335. <https://doi.org/10.1016/j.bbe.2017.03.001>.

Ke, Q., Liu, J., Bennamoun, M., An, S., Sohel, F., & Boussaid, F. (2018). Computer vision for human-machine interaction. *Computer Vision For Assistive Healthcare*. pp. 127–145. Academic Press. <https://doi.org/10.1016/B978-0-12-813445-0.00005-8>

Kim, J., Mastnik, S., & André, E. (2008). EMG-based hand gesture recognition for realtime biosignal interfacing. *The 13th international conference on Intelligent user interfaces*, pp. 30-39. <https://doi.org/10.1145/1378773.1378778>

Konard, P. (2005). *The ABC of EMG. A practical introduction to kinesiological electromyography, 1*.

Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. (2006). Machine learning: a review of classification and combining techniques. *Artificial Intelligence Review*, 26 (3), pp. 159-190. <https://doi.org/10.1007/s10462-007-9052-3>

Krishnan, K. S., Saha, A., Ramachandran, S., & Kumar, S. (2017). Recognition of human arm gestures using Myo armband for the game of hand cricket. *2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*, Ottawa, ON, Canada, 2017, pp. 389-394. <https://doi.org/10.1109/IRIS.2017.8250154>

Lee, K. H., Min, J. Y., & Byun, S. (2021). Electromyogram-Based Classification of Hand and Finger Gestures Using Artificial Neural Networks. *Sensors*, 22 (1), 225. <https://doi.org/10.3390/s22010225>

Lee, W., Seong, J. J., Ozlu, B., Shim, B. S., Marakhimov, A., & Lee, S. (2021). Biosignal sensors and deep learning-based speech recognition: A review. *Sensors (Switzerland)*, 21 (4). <https://doi.org/10.3390/s21041399>

Lin, Z., Zhang, G., Xiao, X., Au, C., Zhou, Y., Sun, C., Zhou, Z., Yan, R., Fan, E., Si, S., Weng, L., Mathur, S., Yang, J., & Chen, J. (2022). A Personalized Acoustic Interface

- for Wearable Human–Machine Interaction. *Advanced Functional Materials*, 32 (9), <https://doi.org/10.1002/adfm.202109430>
- Lobov, S., Krilova, N., Kastalskiy, I., Kazantsev, V., & Makarov, V. A. (2018). Latent factors limiting the performance of sEMG-interfaces. *Sensors (Switzerland)*, 18 (4), 1122. <https://doi.org/10.3390/s18041122>
- Ma, S., Lv, B., Lin, C., Sheng, X., & Zhu, X. (2020). EMG signal filtering based on variational mode decomposition and sub-band thresholding. *IEEE journal of biomedical and health informatics*, 14 (8), pp.47-58. <https://doi.org/10.1109/JBHI.2020.2987528>
- Marks, L. J., & Michael, J. W. (2001). Science, medicine, and the future: Artificial limbs. *British Medical Journal*, 323 (7315), pp. 732-735. <https://doi.org/10.1136/bmj.323.7315.732>
- Mora-Rubio, A., Alzate-Grisales, J. A., Arias-Garzón, D., Buriticá, J. I. P., Varón, C. F. J., Bravo-Ortiz, M. A., Arteaga-Arteaga, H. B., Hassaballah, M., Orozco-Arias, S., Isaza, G., & Tabares-Soto, R. (2022). Multi-subject Identification of Hand Movements Using Machine Learning. *Lecture Notes in Networks and Systems*, 253, pp. 117-128. [https://doi.org/10.1007/978-3-030-78901-5\\_11](https://doi.org/10.1007/978-3-030-78901-5_11)
- Murciago, L. P., Henrich, M. C., Spaich, E. G., & Dosen, S. (2022). Reducing the number of EMG electrodes during online hand gesture classification with changing wrist positions. *Journal of NeuroEngineering and Rehabilitation*, 19 (1), pp. 1-16. <https://doi.org/10.1186/s12984-022-01056-w>
- Noble, W. S. (2006). What is a support vector machine? *Nature Biotechnology*, 24 (12), pp. 1565-1567. <https://doi.org/10.1038/nbt1206-1565>
- Ortiz-Catalan, M., Brånemark, R., Håkansson, B., & Delbeke, J. (2012). On the viability of implantable electrodes for the natural control of artificial limbs: Review and discussion. *BioMedical Engineering Online*, 11 (1), pp. 1-24. <https://doi.org/10.1186/1475-925X-11-33>
- Ozdemir, M. A., Kisa, D. H., Guren, O., Onan, A., & Akan, A. (2020). EMG based Hand Gesture Recognition using Deep Learning. *2020 Medical Technologies Congress (TIPTEKNO)*, Antalya, Turkey, 2020, pp. 1-4. <https://doi.org/10.1109/TIPTEKNO50054.2020.9299264>
- Pal, M. (2008). Multiclass approaches for support vector machine-based land cover classification. *arXiv preprint arXiv:0802.2411*.
- Palermo, F., Cognolato, M., Gijsberts, A., Müller, H., Caputo, B., & Atzori, M. (2017). Repeatability of grasp recognition for robotic hand prosthesis control based on sEMG

data. *IEEE International Conference on Rehabilitation Robotics (ICORR)*, London, UK, 2017, pp.1154-1159. <https://doi.org/10.1109/ICORR.2017.8009405>

Patle, A., & Chouhan, D. S. (2013). SVM kernel functions for classification. *2013 International Conference on Advances in Technology and Engineering (ICATE)*, Mumbai, India, 2013, pp. 1-9. <https://doi.org/10.1109/ICAdTE.2013.6524743>

Phinyomark, A., Limsakul, C., & Phukpattaranont, P. (2011). *Application of Wavelet Analysis in EMG Feature Extraction for Pattern Classification. Measurement Science Review*, 11 (2), pp. 45-52. doi:10.2478/v10048-011-0009-y.

Phinyomark, A., Quaine, F., Charbonnier, S., Serviere, C., Tarpin-Bernard, F., & Laurillau, Y. (2013). EMG feature evaluation for improving myoelectric pattern recognition robustness. *Expert Systems with Applications*, 40 (12), pp. 4832-4840. <https://doi.org/10.1016/j.eswa.2013.02.023>

Platt, J., & others. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10 (3), pp. 61-74.

Rabbat, N. (2022). Gesture Classification from sEMG Signals using Machine Learning Approaches. <http://hdl.handle.net/1920/12856>.

Raurale, S. A., McAllister, J., & del Rincon, J. M. (2020). Real-Time Embedded EMG Signal Analysis for Wrist-Hand Pose Identification. *IEEE Transactions on Signal Processing*, 68, pp. 2713-2723. <https://doi.org/10.1109/TSP.2020.2985299>

Raurale, S. A., McAllister, J., & Rincon, J. M. Del. (2021). EMG Biometric Systems Based on Different Wrist-Hand Movements. *IEEE Access*, 9, pp. 12256-12266. <https://doi.org/10.1109/ACCESS.2021.3050704>

Rawat, S., Vats, S., & Kumar, P. (2017). Evaluating and exploring the MYO ARMBAND. *2016 International Conference System Modeling & Advancement in Research Trends (SMART)*, Moradabad, India, 2016, pp. 115-120. <https://doi.org/10.1109/SYSMART.2016.7894501>

Ren, X., Zhang, C., Li, X., Yang, G., Potter, T., & Zhang, Y. (2018). Intramuscular EMG decomposition basing on motor unit action potentials detection and superposition resolution. *Frontiers in Neurology*, 9, 2. <https://doi.org/10.3389/fneur.2018.00002>

Rosenstein, N. (2019). *myo-python*. GitHub. <https://github.com/NiklasRosenstein/myo-python/releases>. Visité le 02 mai 2023.

Saelim, W., Phukpattaranont, P., & Thongpull, K. (2019). Effect of electrode skin impedance on electromyography signal quality. *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*

- (ECTI-CON), Chiang Rai, Thailand, 2018, pp. 748-751. doi: 10.1109/ECTICon.2018.08619967.
- SEER Training Modules. (s. d.). *Structure of Skeletal Muscle*. U. S. National Institutes of Health, National Cancer Institute. <https://training.seer.cancer.gov/anatomy/muscular/structure.html>. Visité le 02 mai 2023.
- Semmad, A., & Bahoura, M. (2024). Comparative study of respiratory sounds classification methods based on cepstral analysis and artificial neural networks. *Computers in Biology and Medicine*, 108190. <https://doi.org/10.1016/j.combiomed.2024.108190>
- Sheffield, S. (2022). Motor Units in Skeletal Muscle. Dans *getbodysmart.com*. <https://www.getbodysmart.com/muscle-fiber-nerve-supply/motor-units/>. Visité le 02 mai 2023.
- Shi, W. T., Lyu, Z. J., Tang, S. T., Chia, T. L., & Yang, C. Y. (2018). A bionic hand controlled by hand gesture recognition based on surface EMG signals: A preliminary study. *Biocybernetics and Biomedical Engineering*, 38 (1), pp. 126-135. <https://doi.org/10.1016/j.bbe.2017.11.001>.
- Suthaharan, S. (2016). Machine learning models and algorithms for big data classification. *Integr. Ser. Inf. Syst*, 36, pp.1-12. <https://doi.org/10.1007/978-1-4899-7641-3>
- Tepe, C., & Demir, M. C. (2021). The effects of the number of channels and gyroscopic data on the classification performance in EMG data acquired by Myo armband. *Journal of Computational Science*, 51, 101348. <https://doi.org/10.1016/j.jocs.2021.101348>
- Tepe, C., & Demir, M. C. (2022). Real-Time Classification of EMG Myo Armband Data Using Support Vector Machine. *IRBM*, 43 (4), pp. 300-308. <https://doi.org/10.1016/j.irbm.2022.06.001>
- Tharwat, A. (2018). Classification assessment methods. *Applied Computing and Informatics*, 17 (1), pp. 168-192. <https://doi.org/10.1016/j.aci.2018.08.003>
- Tomaszewski, M. (2023). *Myo SDK MATLAB MEX Wrapper*. GitHub. <https://github.com/mark-toma/MyoMex>. Visité le 02 mai 2023.
- Too, J., Abdullah, A. R., & Saad, N. M. (2019). Classification of hand movements based on discrete wavelet transform and enhanced feature extraction. *International Journal of Advanced Computer Science and Applications*, 10 (6). <https://doi.org/10.14569/IJACSA.2019.0100612>
- Too, J., Abdullah, A. R., Saad, N. M., & Tee, W. (2019). EMG feature selection and classification using a Pbest-guide binary particle swarm optimization. *Computation*, 7 (1), 12. <https://doi.org/10.3390/computation7010012>

- Ukil, A. (2007). Support Vector Machine. *Image Processing*, pp. 161-226. [https://doi.org/10.1007/978-3-540-73170-2\\_4](https://doi.org/10.1007/978-3-540-73170-2_4)
- Venugopal, G., Navaneethakrishna, M., & Ramakrishnan, S. (2014). Extraction and analysis of multiple time window features associated with muscle fatigue conditions using sEMG signals. *Expert Systems with Applications*, 41 (6), pp. 2652-2659. <https://doi.org/10.1016/j.eswa.2013.11.009>
- Vural, V., & Dy, J. G. (2004). A hierarchical method for multi-class support vector machines. *Proceedings, Twenty-First International Conference on Machine Learning, ICML 2004*, Banff, Alberta, Canada, 2004. <https://doi.org/10.1145/1015330.1015427>
- Wahid, M. F., Tafreshi, R., Al-Sowaidi, M., & Langari, R. (2018). Subject-independent hand gesture recognition using normalization and machine learning algorithms. *Journal of Computational Science*, 27, pp. 69-76. <https://doi.org/10.1016/j.jocs.2018.04.019>
- Wahid, M. F., Tafreshi, R., & Langari, R. (2020). A Multi-Window Majority Voting Strategy to Improve Hand Gesture Recognition Accuracies Using Electromyography Signal. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28 (2), pp. 427-436. <https://doi.org/10.1109/TNSRE.2019.2961706>
- Wang, N., Chen, Y., & Zhang, X. (2013). The recognition of multi-finger prehensile postures using LDA. *Biomedical Signal Processing and Control*, 8 (6), pp. 706-712. <https://doi.org/10.1016/j.bspc.2013.06.006>
- Xiong, D., Zhang, D., Zhao, X., & Zhao, Y. (2021). Deep Learning for EMG-based Human-Machine Interaction: A Review. *IEEE/CAA Journal of Automatica Sinica*, 8 (3), pp. 512-533. <https://doi.org/10.1109/JAS.2021.1003865>
- Xue, J., & Lai, K. W. C. (2023). Dynamic gripping force estimation and reconstruction in EMG-based human-machine interaction. *Biomedical Signal Processing and Control*, 80, 104216. <https://doi.org/10.1016/j.bspc.2022.104216>
- Zhang, Z., Liu, S., Wang, Y., Song, W., & Zhang, Y. (2024). Online cross session electromyographic hand gesture recognition using deep learning and transfer learning. *Engineering Applications of Artificial Intelligence*, 127, 107251. <https://doi.org/10.1016/j.engappai.2023.107251>
- Zheng, N., Li, Y., Zhang, W., & Du, M. (2022). User-Independent EMG Gesture Recognition Method Based on Adaptive Learning. *Frontiers in Neuroscience*, 16, 847180. <https://doi.org/10.3389/fnins.2022.847180>
- Zhou, Z.-H. (2021a). Dimensionality Reduction and Metric Learning. *Machine Learning*, pp. 241-264. Springer Singapore. [https://doi.org/10.1007/978-981-15-1967-3\\_10](https://doi.org/10.1007/978-981-15-1967-3_10)
- Zhou, Z.-H. (2021b). Linear Models. *Machine Learning*, pp. 57-77. Springer Singapore. [https://doi.org/10.1007/978-981-15-1967-3\\_3](https://doi.org/10.1007/978-981-15-1967-3_3)

Zhou, Z.-H. (2021c). Machine Learning. Springer Singapore. <https://doi.org/10.1007/978-981-15-1967-3>

Zhou, Z.-H. (2021d). Support Vector Machine. *Machine Learning*, pp. 129-153. Springer Singapore. [https://doi.org/10.1007/978-981-15-1967-3\\_6](https://doi.org/10.1007/978-981-15-1967-3_6)

Zhu, F., Gao, J., Yang, J., & Ye, N. (2022). Neighborhood linear discriminant analysis. *Pattern Recognition*, 123, 108422. <https://doi.org/10.1016/j.patcog.2021.108422>