

UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À  
L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE  
DE LA MAITRISE EN MATHÉMATIQUES ET INFORMATIQUE  
APPLIQUÉES

PAR  
BOIRO THIerno MOUSSA

RECONNAISSANCE DES ACTIVITES HUMAINES A TRAVERS LES TELEPHONES  
INTELLIGENTS

OCTOBRE 2022

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire, de cette thèse ou de cet essai a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire, de sa thèse ou de son essai.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire, cette thèse ou cet essai. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire, de cette thèse et de son essai requiert son autorisation.



# Abstract

Human Activity Recognition (HAR) is a method for identifying a person's activity using sensitive sensors to detect movements. Smartphone users and capabilities (sensors) are growing, and users are increasingly carrying their phones with them. These facts increase the significance and popularity of HAR. This research focuses on applying several machine learning classification algorithms to recognize human activity using smartphone sensors. This study also is a comprehensive look at the architecture of the human activity recognition system and examines a variety of design issues. It also emphasizes how feature selection methods reduce computational costs and achieving significant accuracy. It also introduces the use of recurrent neural networks to learn features from long sequences of time-series data, which can help to enhance accuracy and lessen dependency on domain knowledge for feature engineering and extraction.

# Remerciement

Merci à mon superviseur, **Monsieur Nouboud Fathallah**, pour votre patience, vos conseils et votre soutien. J'ai grandement bénéficié de votre richesse de connaissances et de votre édition méticuleuse. Je suis extrêmement reconnaissant que vous m'ayez accepté comme étudiant et que vous ayez continué à me faire confiance au fil des ans.

Je suis reconnaissant envers mes parents dont l'amour et le soutien constants me gardent motivée et confiant. Mes réalisations et mon succès sont dus au fait qu'ils ont cru en moi. Un grand merci à mes frères, sœurs et à mon meilleur ami **Aglaou Souleymane**, qui me gardent les pieds sur terre, me rappellent ce qui est important dans la vie et me soutiennent toujours dans mes aventures.

# Résumé

Actuellement, il est facile de collecter des données de séquence brutes car les téléphones portables sont si courants et intègrent de nombreux capteurs. L'écart entre l'activité humaine et divers capteurs est comblé par ces données. Dans de nombreuses facettes de notre vie quotidienne, telles que la sécurité médicale, la sécurité personnelle, le soutien à la vie et d'autres domaines, la reconnaissance de l'activité humaine est désormais largement utilisée.

J'ai décrit diverses technologies importantes pour les problèmes essentiels de détection de l'activité humaine, telles que la catégorisation des activités, l'ingénierie des fonctionnalités et les algorithmes communs qui ont été développés récemment, afin de donner un aperçu. J'ai d'abord exploré la nature des capteurs embarqués et leurs caractéristiques dans cette étude. Après le processus d'identification de l'activité humaine, j'ai continué à couvrir les moyens de prétraiter les données brutes et d'extraire les caractéristiques, ainsi que quelques algorithmes bien connus pour l'identification des modèles et la reconnaissance des caractéristiques. Troisièmement, j'ai présenté des scénarios d'application de reconnaissance d'activité humaine intrigants et proposé des ensembles de données accessibles pour vérifier les techniques suggérées.

J'ai esquissé les principales difficultés et suggéré quelques directions possibles pour une étude plus approfondie.

# Tableau des matière

1.1 Introduction.....	2
1.2 Problématique.....	3
1.4 Questions de recherche.....	5
2.1 Architecture générale.....	7
2.2 Modalité de capteur.....	8
2.2.1 Capteurs portés sur le corps.....	8
2.2.2 Capteurs d'objets.....	8
2.2.3 Capteurs ambiants.....	9
2.3 Les problèmes de conception.....	9
2.3.1 Sélection des attributs de capteur.....	9
2.3.1.1 Signal d'accélération.....	9
2.3.1.2 Signaux de localisation.....	10
2.3.1.3 Attributs environnementaux et signaux physiologiques.....	10
2.3.2 Consommation d'énergie.....	10
2.3.3 Techniques de traitement.....	11
2.3.4 Interférence.....	11
2.3.5 La flexibilité.....	11
2.4 Travaux Connexes.....	12
2.4.1 Systèmes en ligne.....	12
2.4.2 Systèmes hors ligne.....	13
3.1 Les données de la reconnaissance d'activités humaines.....	18
3.2 Prétraitement des données.....	20
3.2.2 Méthodes embarquées (Embedded Methods).....	22
3.2.4 Sélection de fonctionnalités basée sur l'arborescence.....	24
3.2.5 Sélection de fonctionnalités basées sur la régression lasso (L1).....	25
3.2.7 Algorithmes d'apprentissage automatique.....	26
3.2.7.1 Classificateur d'arbre de décision.....	26
3.2.7.2 Classificateur aléatoire de Forrest.....	27
3.2.7.3 Réseau neuronal artificiel.....	28
3.2.7.4 Réseau de neurones récurrents.....	31
4.1 Technologies et outils de développement.....	34
5.1 L'analyse des données.....	40

<b>5.2 Évaluation du modèle d'apprentissage automatique</b> .....	47
<b>5.2.1 Sélection des fonctionnalités et évaluation du modèle de classification</b> .....	47
<b>5.2.2 Réseau de mémoire longue à court terme (LSTM)</b> .....	51
<b>6.1 Conclusion</b> .....	57
<b>6.2 Travaux Futurs</b> .....	57
<b>6.3 References</b> .....	59

# La table des tableaux

<b>Tableau 1. Répartition en pourcentage des activités</b> .....	41
<b>Tableau 2 Méthode de sélection des caractéristiques et dimensions résultantes</b> .....	47
<b>Tableau 3 Résumé du modèle et de la précision</b> .....	48
<b>Tableau 4 Récapitulatif du temps de génération et de prédiction</b> .....	49
<b>Tableau 5 Évaluation du modèle RNN-LSTM sur 10 exécutions.</b> .....	53
<b>Tableau 6 Tableau de comparaison de précision</b> .....	54





# La table des figures

Figure 1 Structure d'acquisition de données pour le système HAR	8
Figure 2 Orientation de l'axe d'un appareil smartphone .....	19
Figure 3 Structure générale de l'arbre de décision pour la classification .....	27
Figure 4 Structure générale des forêts aléatoires pour la classification .....	28
Figure 5 Structure générale du réseau de neurones artificiels.....	29
Figure 6 Courbe sigmoïde.....	30
Figure 7 Courbe ReLU.....	30
Figure 8 Structure générale du réseau de neurones récurrents.....	31
Figure 9 Fréquence des activités dans l'ensemble de données d'entraînement .....	42
Figure 10 Fréquence des activités dans l'ensemble de données de test.....	43
Figure 11 Accélération moyenne du corps - Diagramme de dispersion de l'axe X .....	44
Figure 12 Accélération moyenne du corps - Diagramme de dispersion de l'axe Y .....	44
Figure 13 Accélération maximale du corps - Diagramme de dispersion de l'axe X.....	45
Figure 14 Diagramme de dispersion de l'angle (X, gravité moyenne).....	46
Figure 15 Précision par modèles et processus de sélection .....	49
Figure 16 Temps par modèles et processus de sélection .....	51

# **Chapitre I : Introduction**

## 1.1 Introduction

Les appareils mobiles sont devenus une partie intégrante de notre vie quotidienne. Cela est dû à une augmentation du développement sophistiqué et à l'intégration de capteurs de qualité, à une puissance de calcul élevée, à une grande capacité de stockage et à une connectivité continue dans les appareils mobiles. Les gens interagissent constamment avec leurs smartphones à faible coût et de petite taille dans leurs activités quotidiennes, ce qui a conduit à l'essor de la recherche visant à extraire des connaissances à partir de données acquises par des capteurs omniprésents dans les appareils mobiles. L'attention portée à la création d'un journal de vie, qui fait référence à l'utilisation de la technologie pour capturer et documenter de grandes quantités de la vie d'un utilisateur via des appareils mobiles, a considérablement augmenté. Un bon exemple du journal de vie est la capture du nombre de pas parcourus chaque jour à l'aide d'un smartphone. Le journal de vie peut être utilisé pour documenter des activités physiques simples telles que marcher, courir, s'asseoir, etc. ou des activités complexes telles que manger, travailler, faire de l'exercice, etc. Cela a une grande variété d'applications dans divers domaines de recherche tels que la médecine, la réalité augmentée, interaction homme-ordinateur, sécurité et publicité ciblée. Un journal de vie peut être utilisé pour approfondir les connaissances et donner des informations sur le mode de vie d'un utilisateur et aider à améliorer la qualité de vie en fournissant des recommandations et des services personnalisés. La création d'applications et de services contextuels avec du matériel grand public à faible coût sera une étape importante vers la résolution de problèmes plus complexes.

Le plus gros problème rencontré lors de la création d'un journal de vie détaillé est la collecte de données d'activité via divers capteurs portables et la classification précise de l'activité humaine sur la base des données collectées. Les appareils mobiles sont intégrés à des capteurs puissants, tels que des accéléromètres, des gyroscopes, des GPS, des magnétomètres, des détecteurs de proximité et de lumière ambiante. Avec l'utilisation de smartphones, les chercheurs ont la possibilité de collecter facilement des données de capteurs avec l'utilisation d'une infrastructure minimale. Les techniques modernes d'apprentissage automatique peuvent être utilisées pour distinguer et reconnaître les activités humaines en fonction des données collectées. Un simple smartphone pourrait aider à résoudre le problème de la documentation d'un historique détaillé de l'activité quotidienne d'un utilisateur. Les progrès de l'apprentissage en profondeur et des méthodes de sélection de

caractéristiques ainsi que l'inclusion d'une variété de capteurs peuvent repousser les limites de la reconnaissance de l'activité humaine à des niveaux ontologiques plus profonds.

Ce rapport est organisé de la manière suivante :

Le *chapitre 1* parle de la problématique, des objectifs et des questions de recherches liés à la reconnaissance des activités humaine. *Le chapitre 2* décrit la technique des traitements de données et aussi discute des travaux connexes. *Le chapitre 3* discute des méthodes de classification. De plus, le chapitre explique pourquoi il est important d'améliorer les performances globales de la classification. *Le chapitre 4* décrit les technologies et outils de développement *utilisé pour l'implémentation de l'application mobile*. *Le chapitre 5* discute des résultats obtenus et le *chapitre final* parle de la conclusion ainsi que les travaux futurs et la liste des références.

## 1.2 Problématique

La Reconnaissance d'Activité humaine est le problème d'identification d'une activité physique réalisée par un individu dépendant d'une trace de mouvement dans un certain environnement. Les activités telles que marcher, s'allonger, s'asseoir, se tenir debout et monter des escaliers sont classées comme des mouvements physiques réguliers et forment notre classe d'activité qui doit être reconnue. Pour enregistrer un mouvement ou un changement de mouvement, des capteurs tels que des accéléromètres triaxiaux et des gyroscopes capturent des données pendant que l'activité est en cours. Les données d'un accéléromètre triaxial détectent l'accélération ou le mouvement le long des trois axes et un gyroscope mesure la rotation le long des trois axes pour déterminer la direction. Les données enregistrées sont le long des trois dimensions des axes X, Y et Z à la fréquence spécifiée. Par exemple, une fréquence de 20 Hz indiquerait que 20 points de données sont enregistrés chaque seconde de l'action. Divers autres signaux physiologiques tels que le rythme cardiaque, la respiration, etc. et les signaux environnementaux tels que la température, l'heure, l'humidité, etc. peuvent augmenter davantage le processus de reconnaissance. La reconnaissance d'activité peut être réalisée en exploitant les informations récupérées à partir de ces capteurs. Le défi se pose car il n'y a pas d'approche explicite pour déduire les actions humaines des informations des capteurs de manière générale. Le grand volume de données produites par les capteurs et l'utilisation de ces fonctionnalités pour développer des heuristiques introduisent le défi technique. Le stockage, la communication, le calcul, l'efficacité énergétique et la flexibilité du système sont quelques-uns des aspects qui doivent

être analysés en détail pour construire un système de reconnaissance d'activité robuste. Les méthodes conventionnelles de reconnaissance de formes ont fait d'énormes progrès dans la découverte d'informations importantes à partir de scores de lectures de bas niveau. Mais de tels modèles de reconnaissance sont efficaces pour les données collectées dans des environnements contrôlés, et pour quelques activités seulement. Les tâches de la reconnaissance d'activités complexes sont entravées en raison des techniques d'extraction de caractéristiques naïves et de la limitation de la connaissance du domaine.

Les caractéristiques superficielles extraites dégradent les performances des algorithmes d'apprentissage non supervisés et des activités connectées. Les modèles d'apprentissage en profondeur ont la capacité d'apprendre des fonctionnalités d'ordre supérieur. L'avancement de ces modèles permet d'apprendre et d'améliorer les performances des modèles prédictifs et d'approfondir les connaissances sur les activités humaines.

### **1.3 Objectifs**

L'objectif principal de ce travail est d'effectuer la reconnaissance de l'activité physique humaine de manière discrète et sans avoir besoin d'une infrastructure fixe. L'accent est mis sur l'utilisation de méthodes d'apprentissage automatique simples, car la mise en œuvre est destinée aux petits appareils aux ressources limitées. De cet objectif général, les objectifs particuliers suivants sont dérivés :

- Effectuez la reconnaissance de l'activité physique en temps réel à l'aide d'un smartphone.
- Rechercher quelles fonctionnalités sont simples à calculer mais efficaces pour caractériser les activités.
- Évaluer différents types de classificateurs de niveau de base afin de déterminer celui qui convient le mieux pour se conformer à l'objectif général.
- Concevoir un moyen de contextualiser les activités physiques afin d'avoir une meilleure compréhension de l'état de l'utilisateur.
- Validez les résultats afin de savoir si la reconnaissance convient ou non à une utilisation réelle.

## 1.4 Questions de recherche

Pour atteindre les objectifs proposés et vérifier les hypothèses, les questions de recherche suivantes sont abordées :

- Pouvons-nous utiliser de simples capteurs de smartphone pour effectuer une reconnaissance d'activité en temps réel ?
- Peut-on utiliser de simples méthodes d'apprentissage supervisé pour classer les activités physiques ?
- Quelles caractéristiques conviennent à la tâche de classification ?
- Comment représenter des activités complexes pour effectuer une classification ?

## 1.5 Conclusion

Les journaux de vie, qui utilisent la technologie pour capturer et documenter la vie d'un utilisateur via des appareils mobiles, ont considérablement augmenté ces dernières années. Un exemple de journal de vie est le nombre de pas parcourus chaque jour. Les journaux de vie peuvent être utilisés pour documenter diverses activités, telles que marcher, courir, s'asseoir, etc., et peuvent même être utilisés pour approfondir les connaissances et donner un aperçu de la vie d'un utilisateur. Pour améliorer la qualité de vie d'un utilisateur, des recommandations et des services personnalisés peuvent être donnés après la création des journaux de vie. La création d'applications et de services contextuels avec du matériel grand public à faible coût constituera une étape importante vers la résolution de problèmes plus complexes. Le plus gros problème rencontré lors de la création d'un journal de vie détaillé est la collecte de données d'activité via divers capteurs portables et la classification précise de l'activité humaine. Leur solution proposée consiste à utiliser des appareils mobiles dotés de capteurs puissants, notamment des accéléromètres, des gyroscopes, des GPS et des magnétomètres, entre autres, pour collecter facilement des données. Les techniques d'apprentissage automatique peuvent être utilisées pour distinguer et reconnaître les activités humaines sur la base des données collectées. Ils pensent qu'un simple smartphone pourrait aider à résoudre ce problème. Les progrès de l'apprentissage en profondeur et des méthodes de sélection des fonctionnalités, ainsi que l'inclusion d'une variété de capteurs, peuvent repousser les limites de la reconnaissance de l'activité humaine à des niveaux ontologiques plus profonds.

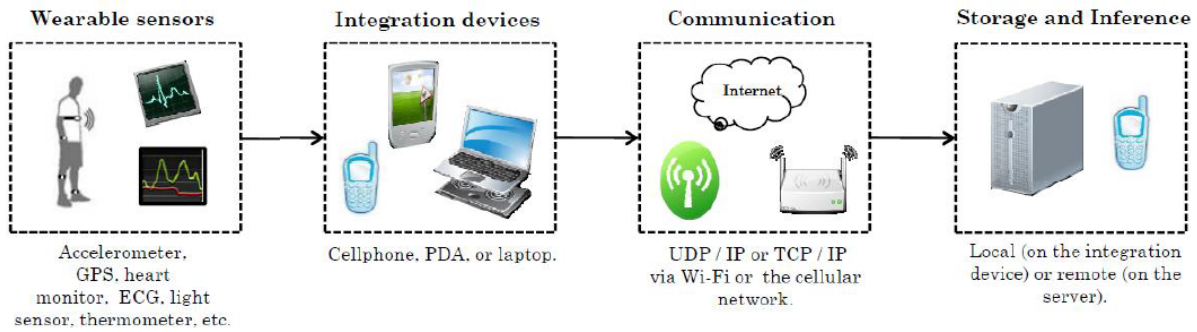
Avoir la possibilité de reconnaître l'activité humaine à l'aide d'un smartphone peut être bénéfique pour rendre notre monde meilleur. À la fin de ce projet, nous aurons un logiciel efficace et efficient qui peut reconnaître l'activité humaine. Dans le prochain chapitre du rapport nous allons décrire la technique des traitements de données et aussi discuter des travaux connexes

# **Chapitre II : Revue de Littérature**



## 2.1 Architecture générale

Le système de reconnaissance de l'activité humaine a une architecture généralisée composée de deux phases principales - (1) Acquisition de données, (2) Reconnaissance d'activité. L'acquisition de données concerne principalement la collecte et le stockage des données des capteurs, tandis que la reconnaissance d'activité traite des modèles d'apprentissage automatique utilisés pour l'analyse prédictive. Le système d'acquisition de données a une structure standard comme le montre la **figure 1**. Le composant principal de la phase d'acquisition de données sont les capteurs qui mesurent les différents attributs tels que l'accélération, la localisation, l'audio, la température, etc. Les autres composants sont le dispositif d'intégration, le réseau de communication et le serveur d'application distant. Le dispositif d'intégration est principalement utilisé pour collecter et prétraiter le signal brut du capteur. Les données peuvent également être envoyées à un serveur d'applications distant à l'aide de protocoles réseau tels que TCP/IP ou UDP, pour une analyse et une visualisation en temps réel. Tous les composants ne sont pas nécessaires et mis en œuvre dans chaque système d'acquisition de données. Les capteurs sont intégrés dans un appareil lui-même et effectuent le traitement analytique sur celui-ci. D'autres systèmes nécessitent un dispositif portable externe qui communique avec un dispositif d'intégration tel qu'un ordinateur portable, un téléphone portable, etc. Différentes applications et exigences reflètent ces différences dans l'architecture générale. Le composant de reconnaissance d'activité s'appuie fortement sur des modèles d'apprentissage automatique et repose sur les étapes de formation et de test. Comme décrit dans [11], l'étape de formation nécessite un grand ensemble de données des caractéristiques collectées pour former le modèle. Des processus variés tels que le nettoyage des données, l'extraction de caractéristiques, la réduction de la dimensionnalité et la sélection de caractéristiques ont lieu au stade de la formation. De même, les auteurs de [11] décrivent la phase de test. Cette étape a un ensemble de données plus petit et subit le même traitement de données. Il est ensuite utilisé pour tester les prédictions de la machine et évaluer la formation.



**Figure 1 Structure d'acquisition de données pour le système HAR**

## 2.2 Modalité de capteur

Selon [27], les modalités des capteurs sont classées en trois types de base :

- Capteurs portés sur le corps
- Capteurs d'objets
- Capteurs ambiants

### 2.2.1 Capteurs portés sur le corps

Les capteurs tels qu'un accéléromètre, un gyroscope et un GPS qui sont intégrés dans des appareils tels que les smartphones, les montres, les lunettes, les casquettes, etc. relèvent de la catégorie des capteurs portés sur le corps. Ces capteurs sont fixés à diverses positions sur le corps humain pour tracer et reconnaître différentes activités. Parmi les travaux étudiés, l'accéléromètre est le capteur le plus adopté, en raison de ses capacités d'enregistrement des changements d'accélération du corps humain. Un gyroscope ou un magnétomètre est couplé à l'accéléromètre pour reconnaître les activités physiques quotidiennes ou les activités sportives.

### 2.2.2 Capteurs d'objets

Les capteurs d'objets sont des capteurs qui sont placés dans un environnement ou sur certains objets pour détecter le mouvement d'un objet spécifique. RFID, WIFI, Bluetooth peuvent enregistrer des informations à un niveau plus profond pour reconnaître des activités complexes. De tels capteurs ne sont pas portés sur le corps de la personne mais sont placés dans l'environnement pour détecter les mouvements sous un angle différent. L'utilisation de tels capteurs est très rare car ils sont coûteux et difficiles à installer. L'association de capteurs d'objets avec des capteurs portés sur le corps dans

des environnements contrôlés peut aider à faire avancer la recherche d'une reconnaissance plus profonde de l'activité humaine au niveau ontologique.

### **2.2.3 Capteurs ambiants**

Les capteurs ambiants capturent les variables de l'environnement telles que la température, le son, la pression, l'humidité, etc. Ils ne capturent pas spécifiquement le changement de mouvement d'un être humain mais capturent des données liées au changement dans l'environnement. Ils fournissent des connaissances holistiques sur l'environnement d'une personne et l'environnement dans lequel l'action est menée. De tels capteurs se trouvent dans les environnements de maison intelligente.

## **2.3 Les problèmes de conception**

Plusieurs problèmes de conception doivent être pris en compte pour développer un système de reconnaissance d'activités humaine efficace. Les problèmes de conception ont un impact considérable sur l'utilisation du système et la précision de la prédiction.

### **2.3.1 Sélection des attributs de capteur**

Une variété de capteurs sont présents et peuvent être globalement classés dans les trois groupes suivants :

#### **2.3.1.1 Signal d'accélération**

Les accéléromètres triaxiaux sont des capteurs peu énergivores et bon marché qui sont largement utilisés pour reconnaître des activités telles que marcher, dormir, s'asseoir, etc. La plupart des activités ambulatories peuvent être reconnues à l'aide d'un accéléromètre triaxial [7] – [9]. La position et le placement de l'accéléromètre jouent un rôle important dans la prédiction de l'activité et de la précision atteinte. Selon He et al. [9], le meilleur endroit pour garder l'accéléromètre est dans la poche du pantalon, mais cette conclusion varie selon le type d'activité à reconnaître. Les accéléromètres ne sont pas non plus utiles pour déduire des activités plus significatives telles que travailler au bureau ou manger à table. Le même mouvement du corps pour plusieurs activités est déroutant à reconnaître du point de vue de l'accéléromètre [15].

### **2.3.1.2 Signaux de localisation**

Les systèmes de positionnement global fournissent des données de localisation à l'aide de satellites. Tous les smartphones actuels sont équipés de capteurs GPS et les données peuvent être utilisées pour fournir une reconnaissance contextuelle ou déduire une activité basée sur un raisonnement ontologique [5]. Cependant, Reddy et al. [19] rapportent que les plus gros problèmes rencontrés par les capteurs GPS sont qu'ils fonctionnent mal à l'intérieur, qu'ils consomment beaucoup d'énergie et qu'ils sont associés à des problèmes de confidentialité. Cela limite l'utilisation des données GPS pour les applications en temps réel. Riboni et al. [5] suggère que pour surmonter les mauvaises performances à l'intérieur, les capteurs GPS devraient être associés à des accéléromètres.

### **2.3.1.3 Attributs environnementaux et signaux physiologiques**

Pour de meilleures informations contextuelles, des attributs tels que l'audio, la température, l'intensité lumineuse, l'heure, les microphones, etc. sont utilisés. Ils fournissent des informations sur le cadre environnemental d'un individu pour en déduire des activités. Les auteurs de [15] et [17] analysent qu'individuellement, les capteurs environnementaux ne contribuent pas suffisamment à la reconnaissance et sont également facilement affectés en raison des conditions météorologiques, de l'éclairage artificiel externe et des niveaux de bruit élevés. Les signaux physiologiques tels que la fréquence cardiaque ou respiratoire, l'ECG, la température corporelle peuvent être considérés comme des signes vitaux et ont été utilisés dans quelques systèmes de reconnaissance d'activités humain. Tapia et al. [16] ont combiné les données recueillies à partir d'un moniteur cardiaque et d'un accéléromètre pour la reconnaissance d'activité et ont conclu que le signal de fréquence cardiaque n'est pas utile. Il a mal classé les activités tout en augmentant les coûts et l'énergie avec l'utilisation de capteurs intrusifs supplémentaires.

### **2.3.2 Consommation d'énergie**

Les capteurs individuels et les capteurs intégrés dans les appareils mobiles sont fortement limités par la durée de vie de la batterie. De nombreuses applications nécessitent que des données critiques soient fournies par les capteurs, ce qui rend les capteurs énergivores efficaces hautement souhaitables. La durée de vie de la batterie peut être prolongée en limitant la communication avec les capteurs car il s'agit d'une opération très coûteuse. Les techniques de communication à courte portée, de filtrage des données et de compression doivent également être utilisées pour économiser

de l'énergie. Riboni et al. dans [5] expliquent comment l'analyse et la classification doivent être effectuées sur le dispositif intégré lui-même pour réduire la communication avec un serveur externe.

### **2.3.3 Techniques de traitement**

Le traitement des données sur le serveur ou dans l'appareil intégré lui-même est une décision de conception importante. Lara et al. Dans [11] en déduire que l'aspect de la conception dépend de l'application, que des résultats rapides en temps réel ou des résultats passifs soient requis. Un système reconnaissance d'activités humain déployé sur un appareil mobile est une conception plus évolutive car il réduit la communication et allège la charge du serveur en calculant localement la classification. Cela aiderait également à surmonter les systèmes de communication peu fiables et à devenir une application très réactive. Cependant, les appareils mobiles ne disposent pas d'une puissance de traitement élevée et de besoins de stockage importants. Dans [12] - [14], les chercheurs discutent des lacunes d'un système de reconnaissance d'activités humain mobile en termes de stockage, de consommation d'énergie et de puissance de calcul. Dans [13], Williams et Matthew décrivent un scénario où l'application nécessite des données d'un groupe d'utilisateurs, et comment il est avantageux de calculer la classification sur un serveur central plutôt que sur des appareils mobiles individuels. Ce compromis doit être soigneusement analysé avec les besoins de l'application.

### **2.3.4 Interférence**

De nombreux systèmes HAR nécessitent la collecte de données à partir de nombreux capteurs pour une inférence plus approfondie. Cela oblige les individus à porter ou à transporter plus de capteurs, ce qui peut devenir inconfortable, coûteux et également invasif. Les systèmes discrets sont hautement souhaitables et les auteurs de [4], [15] and [19] proposent des systèmes qui collectent des données à partir de capteurs embarqués dans un smartphone ou une montre.

### **2.3.5 La flexibilité**

La conception d'un modèle de reconnaissance d'activité fait l'objet d'un examen minutieux et de débats, car certaines études suggèrent que le modèle de reconnaissance devrait être spécifique à un individu [23] et certains soulignent que le modèle devrait être suffisamment flexible pour un groupe généralisé [20]. L'analyse montre qu'il n'est ni approprié ni efficace d'entraîner le même modèle pour différents utilisateurs s'il y a trop d'activités pour lesquelles entraîner le modèle ou si l'individu est incapable d'effectuer certaines activités (par exemple, la natation). Cependant,

reconnaître les activités sans tenir compte des caractéristiques individuelles entraînerait une diminution de la précision et de l'efficacité de l'entraînement.

## 2.4 Travaux Connexes

Dans cette section, nous analysons les systèmes de reconnaissance d'activités humaines précédentes qui reposent sur l'apprentissage supervisé et les données de capteurs et les séparons en systèmes en ligne et hors ligne. L'apprentissage supervisé nécessitait des données étiquetées pour apprendre et donner un résultat. Les systèmes en ligne fournissent un retour immédiat tandis que les systèmes hors ligne ont besoin de plus de temps pour reconnaître les activités en raison des exigences de calcul élevées.

### 2.4.1 Systèmes en ligne

**Vigilante** est une application mobile conçue par les auteurs de [24] pour la reconnaissance en temps réel de l'activité humaine. Les auteurs ont mesuré l'accélération et les signaux physiologiques tels que la fréquence cardiaque et respiratoire, la forme d'onde respiratoire et la température de la peau à l'aide de capteurs externes. À l'aide de la bibliothèque **d'évaluation mobile de l'algorithme de classification**, les auteurs ont implémenté le classificateur d'arbre de décision C4.5 pour reconnaître trois activités de déambulation. Le classificateur a atteint une précision globale de 92,6 %. L'application a un temps de réponse rapide et est formée avec différents utilisateurs aux caractéristiques diverses pour garantir la construction d'un système plus flexible. Cela a réduit le besoin de recycler le modèle pour les nouveaux utilisateurs et a été considéré comme un système économe en énergie modéré. Berchtold et al. dans [4] a proposé la plate-forme **Actiserv** qui utilisait un téléphone portable pour capturer le signal d'accélération. Les auteurs ont développé un système d'inférence floue efficace et portable pour classer les activités de déambulation. La précision atteinte variait entre 71 % et 97 %. Si l'algorithme est censé répondre à une réponse en temps réel, la précision tombe à 71 %, et si l'algorithme est autorisé à s'entraîner à sa pleine capacité, ce qui prend quelques jours, il atteint une précision améliorée de 97 %. Une analyse dépendante du sujet a augmenté la précision à 90 %. Riboni et al. [5] ont présenté **COSAR**, un framework pour la reconnaissance d'activités contextuelle. Grâce à COSAR, il était possible de reconnaître des activités telles que le brossage, l'écriture sur un tableau noir, la promenade, etc. Les auteurs ont utilisé la combinaison de deux accéléromètres et d'un capteur GPS dans un téléphone portable pour collecter des données. Les auteurs ont introduit les concepts de matrice d'activité potentielle et de classification historique

statistique pour filtrer la prédiction d'activité qui n'était pas appropriée. La précision globale atteinte était de 93 %.

### 2.4.2 Systèmes hors ligne

Parkka et al. [17] ont utilisé vingt-deux signaux qui incluent l'accélération, les signes vitaux et les variables environnementales pour classer sept activités. Les capteurs étaient placés à différents endroits du corps et la personne devait transporter un ordinateur compact dans un sac pour collecter les données des capteurs. Ce système était considéré comme très intrusif et posait également de graves problèmes de confidentialité. Les auteurs ont construit trois modèles de classification comme l'arbre de décisions générées automatiquement, l'arbre de décision personnalisé et un réseau de neurones artificiels. Les chercheurs ont atteint la précision la plus élevée de 86 % à partir du premier classificateur. Ces modèles nécessitaient des capacités de calcul élevées et ont donc été classés comme des systèmes hors ligne. Zhu et Sheng [25] ont proposé un système qui utilisait la combinaison de deux classificateurs pour reconnaître les activités. L'architecture du système était très intrusive car les données des capteurs étaient collectées sur un appareil PDA, puis les signaux étaient transférés vers un ordinateur. La classification utilisait les signaux d'accélération et fonctionnait en 2 phases. La première phase nécessitait un réseau de neurones artificiels pour classer l'activité comme stationnaire ou non stationnaire. L'output est ensuite input dans un modèle de modèle de Markov caché (HMM) pour une prédiction d'activité spécifique. Lara et al. [20] ont utilisé un seul smartphone et un capteur pour collecter des informations sur l'accélération et les signes vitaux et créer une plate-forme en temps réel portable et discrète, nommée **Centinela**. Les signaux acquis par les capteurs ont été traités pour extraire les caractéristiques temporelles et fréquentielles. Centinela a été programmé pour reconnaître cinq activités ambulatoires et a atteint une précision de 95,7% après avoir été évaluée sur huit classificateurs différents. Certaines activités ont atteint une précision de 100 %. Comme le système proposé dépendait d'un ensemble de classificateurs, il comportait un coût de calcul élevé et était donc considéré comme un système hors ligne. Pour surmonter les problèmes rencontrés dans ce système, les auteurs ont proposé **Vigilante**.

## 2.5 Conclusion

Des capteurs portés sur le corps sont utilisés pour enregistrer les activités. L'un des capteurs les plus largement utilisés est l'accéléromètre, qui sert à détecter les changements d'accélération du

corps humain. Ce capteur est couplé à un gyroscope ou un magnétomètre pour reconnaître les activités. Des capteurs d'objets sont placés dans un environnement pour détecter un mouvement sous un angle différent. Ils sont utilisés pour reconnaître des activités complexes et ne sont pas portés sur le corps d'une personne, mais placés dans un environnement. Ils peuvent être associés à des capteurs portés sur le corps dans des environnements contrôlés pour faire avancer la recherche d'une reconnaissance plus profonde de l'activité humaine au niveau ontologique. Les capteurs ambiants capturent les variables de l'environnement telles que la température, le son, la pression, l'humidité, etc. Ils fournissent une connaissance holistique de l'environnement d'une personne, tandis que les capteurs d'objets fournissent les changements de mouvement d'une personne. Les problèmes de conception ont un impact sur la précision de la prédiction, la manière dont le système est utilisé et l'utilisation du système. Le principal problème avec ce capteur est qu'il n'est pas capable de différencier avec précision les activités comme s'asseoir et dormir. Le meilleur endroit pour le capteur est dans la poche du pantalon, mais cela varie selon l'activité. L'analyse de l'activité actuelle à l'aide de l'accéléromètre prête à confusion en raison du même mouvement pour plusieurs activités. Les capteurs GPS sont très populaires, mais ils présentent de nombreux défauts, notamment des performances médiocres à l'intérieur et des capteurs à forte consommation d'énergie. Cela limite l'utilisation des données GPS pour les applications en temps réel. Bien que l'idée d'associer des capteurs GPS à des accéléromètres soit séduisante, ces capteurs ont leurs propres inconvénients, en particulier leurs faibles performances à l'intérieur. Afin d'être plus précis, les chercheurs utilisent des données telles que l'audio, la température, l'intensité lumineuse, l'heure, les microphones, etc. de l'environnement pour déduire les activités. Les auteurs de [15] et [17] soutiennent que les capteurs environnementaux ne fournissent pas suffisamment d'informations pour être considérés comme utiles, et sont également faciles à affecter en raison des conditions météorologiques ou des niveaux de bruit élevés. Des signaux physiologiques, comme la fréquence cardiaque ou la fréquence respiratoire, ont été utilisés dans quelques systèmes HAR, mais l'étude Tapia et al. [16] a révélé que le signal de fréquence cardiaque n'était pas efficace. Cela réduisait la précision, augmentait les coûts et l'énergie et utilisait des capteurs gênants supplémentaires. Les capteurs consomment beaucoup d'énergie, mais doivent souvent pouvoir communiquer avec un serveur externe. La solution consiste à intégrer l'analyse et la classification sur l'appareil lui-même pour réduire le besoin de communication. Les auteurs de [5] expliquent comment effectuer des analyses et une classification sur l'appareil lui-même. La conception d'un système HAR sur un



appareil mobile est une conception plus évolutive. Cependant, les appareils mobiles sont moins puissants qu'un serveur centralisé, il est donc important d'analyser attentivement les options de conception et les besoins de l'application. Lara et al. dans [11] proposent que cela dépende de ce qu'est l'application, tandis que Williams et Matthew dans [14] décrivent un scénario où l'application doit calculer la classification sur un serveur central. Certains systèmes HAR exigent que les individus portent ou transportent plus de capteurs, ce qui peut devenir inconfortable et coûteux. Les auteurs dans [4], [15] et [19] proposent des systèmes discrets qui collectent des données à partir des capteurs embarqués. La conception d'un modèle de reconnaissance d'activité fait l'objet d'un examen et d'un débat approfondis, car certaines études suggèrent que le modèle devrait être spécifique à un individu, tandis que d'autres soulignent que le modèle devrait être suffisamment flexible pour un groupe généralisé. Les différents modèles s'avèrent inefficaces s'ils sont formés pour trop d'activités ou s'ils sont incapables d'effectuer certaines activités. Cependant, il n'est pas possible de reconnaître les activités sans tenir compte des caractéristiques individuelles. Ce texte présente deux types de systèmes HAR : l'apprentissage supervisé sur les capteurs et l'apprentissage en ligne et l'apprentissage hors ligne. L'apprentissage en ligne nécessite une rétroaction immédiate tandis que l'apprentissage hors ligne nécessite plus de temps en raison des exigences de calcul élevées. Les auteurs ont créé une application qui peut reconnaître trois activités ambulatoires en utilisant la bibliothèque Mobile Evaluation of Classification Algorithm (MECLA). Ils ont utilisé le classificateur d'arbre de décision C4.5 pour atteindre une précision globale de 92,6 %. Il a également un temps de réponse rapide et est formé avec divers utilisateurs pour garantir la construction d'un système flexible. Les auteurs proposent la plateforme Actiserv qui peut être utilisée avec un téléphone portable pour capter le signal d'accélération. Il est efficace et portable et utilise un système d'inférence floue pour classer les activités ambulatoires. La précision de l'IA variait entre 71% et 97%. Si l'algorithme est censé répondre à une réponse en temps réel, la précision tombe à 71 %. Une analyse dépendante du sujet augmente la précision à 90 %. Riboni et al. [5] ont introduit COSAR, un cadre pour la reconnaissance d'activité sensible au contexte. Grâce au COSAR, il était possible de reconnaître des activités comme le brossage et l'écriture sur un tableau noir. Les auteurs ont utilisé la combinaison de deux accéléromètres et d'un capteur GPS avec un téléphone portable. Les auteurs ont introduit les concepts de matrice d'activité potentielle et de classification historique statistique pour filtrer la prédiction d'activité qui n'était pas appropriée. La précision globale obtenue était de 93 %.

Les chercheurs ont créé trois classificateurs pour reconnaître différentes activités : l'arbre de décision généré automatiquement, l'arbre de décision personnalisé et un réseau neuronal artificiel.

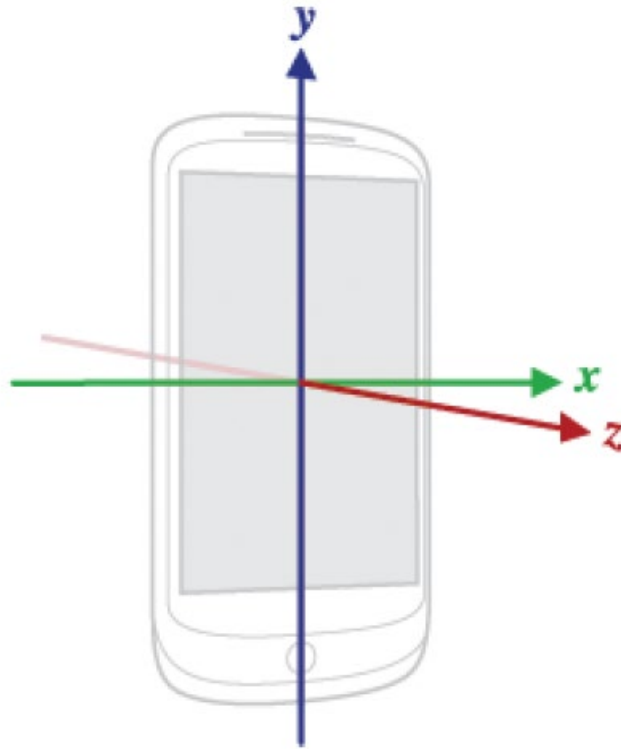
Ils ont obtenu la meilleure précision de 86 % en utilisant le premier classificateur. Le système nécessitait des capacités de calcul élevées et était donc classé comme hors ligne. Ils ont proposé un système qui avait deux classificateurs. L'architecture du système était envahissante car les données des capteurs étaient collectées sur un PDA, puis les signaux étaient transférés vers un ordinateur. La classification utilisait des signaux d'accélération et fonctionnait en 2 phases. La première phase a nécessité un réseau de neurones artificiels pour classer l'activité comme stationnaire ou non stationnaire. La sortie est ensuite entrée dans un modèle de Markov caché (HMM) pour une prédiction d'activité spécifique. Ce travail a utilisé un seul smartphone et un capteur pour collecter des informations sur l'accélération et les signes vitaux et créer une plate-forme en temps réel portable et discrète, nommée Centinela. Les signaux collectés ont été traités pour extraire les caractéristiques de temps et de fréquence et le système a été programmé pour reconnaître cinq activités ambulatoires et a atteint une précision de 95,7 % après avoir été évalué sur huit classificateurs différents. Certaines activités ont atteint une précision de 100 %. Pour surmonter les problèmes rencontrés dans ce système les auteurs ont proposé Vigilante.

En ce qui concerne la croissance de l'industrie de la technologie portable, la reconnaissance d'activité est devenue l'une des études les plus populaires avec un nombre croissant d'applications pratiques et de questions qu'elle peut résoudre. Dans le prochain chapitre du rapport nous allons discuter des méthodes de classification.

# **Chapitre III : *Méthodologie***

### 3.1 Les données de la reconnaissance d'activités humaines

L'ensemble des données de la reconnaissance d'activités humaines à travers les smartphones a été préparé et rendu public par Davide Anguita et al. [26] et peuvent être téléchargés gratuitement à partir du repository d'apprentissage automatique UCI. Les données brutes ne sont pas disponibles, mais la version prétraitée de l'ensemble de données est rendue publique pour effectuer des expériences. Les données du capteur du smartphone ont été collectées à partir des expériences menées sur un groupe de 30 volontaires âgés de 19 à 48 ans. L'ensemble des activités physiques ciblées par les auteurs sont la marche, la position assise, la position debout, allongé, la montée et la descente des escaliers. Les auteurs ont attaché un smartphone Samsung Galaxie SII à chaque sujet pour capturer les données du capteur. Chaque sujet a été chargé d'effectuer chacune des six activités deux fois. Lors du premier essai, le smartphone était fermement attaché sur le côté gauche de la taille du sujet, mais lors du deuxième essai, le sujet a eu la possibilité de placer le smartphone à sa guise. Cela assure qu'il existe une variation des données en fonction de la position du téléphone pour la même activité. La vidéo des auteurs a enregistré l'action effectuée par chaque sujet, ce qui les a aidés à étiqueter manuellement les signaux capturés par les capteurs. Signaux produits par l'accéléromètre et le gyroscope, intégrés dans le Samsung Galaxie SII, sont capturés via une application pour smartphone. Un accéléromètre, comme son nom l'indique, est utilisé pour mesurer l'accélération de l'appareil. Les valeurs le long des axes X, Y et Z sont utilisées pour détecter des mouvements tels que le balancement, l'inclinaison, les vibrations, etc. La figure 2 montre l'orientation de l'axe d'un accéléromètre triaxial par rapport au dispositif. Les valeurs fournies sur les trois axes incluent également l'accélération gravitationnelle de la Terre ( $g = 9,81 \text{ m/s}^2$ ). Si l'appareil mobile est au repos, il n'affichera que l'accélération gravitationnelle sur l'un des axes en fonction de l'orientation. Un gyroscope, quant à lui, utilise la vitesse angulaire pour calculer la rotation ou la torsion d'un smartphone. La vitesse de rotation est mesurée en rad/s le long des trois axes. Alors qu'un accéléromètre détecte un mouvement directionnel, un gyroscope détecte l'orientation latérale de l'appareil. Les deux capteurs sont utilisés pour mesurer le taux de changement, mais pour des choses différentes.



**Figure 2 Orientation de l'axe d'un appareil smartphone**

Anguita et al. A capturé les signaux du capteur à une fréquence constante de 50 Hz et ont ensuite été prétraités pour réduire le bruit. Les signaux ont été prétraités pour la réduction du bruit avec un filtre médian et un filtre Butterworth passe-bas de 3e ordre avec une fréquence de coupure de 20 Hz. Le filtre de Butterworth a été utilisé pour séparer le signal d'accélération en accélération corporelle et accélération gravitationnelle. Les signaux traités ont été échantillonnés dans une fenêtre fixe d'une durée de 2,56 secondes avec un chevauchement de 50 %. Chaque fenêtre comportait 128 points de données pour chacune des caractéristiques d'origine enregistrées, à savoir l'accélération du corps, le gyroscope du corps et l'accélération de la gravité sur les axes X, Y et Z. Les signaux inertiels fenêtrés ont été conçus et plusieurs caractéristiques, temporelles et fréquentielles, ont été extraites de chaque fenêtre. L'ingénierie des caractéristiques a abouti à un vecteur de caractéristiques de 561 éléments. Les auteurs ont divisé au hasard l'ensemble de données dans un rapport 70 :30 qui a créé une distribution de 21 sujets pour la formation et 9 sujets pour les tests. L'ensemble de données d'apprentissage a un total de 7352 fenêtres de données tandis que l'ensemble de données de test a 2947 fenêtres de données. Le processus de reconnaissance d'activités humains devient clair

lorsqu'une fenêtre d'activité, en particulier 2,56 secondes d'activité, est prédite pour un nouvel utilisateur par un modèle formé sur une trace d'activité de sujets connus.

## 3.2 Prétraitement des données

Commençons par définir ce qu'est une fonctionnalité. Une fonctionnalité est une variable  $X$  dans votre ensemble de données, le plus souvent définie par une colonne. De nos jours, de nombreux ensembles de données peuvent avoir plus de 100 fonctionnalités qu'un analyste de données doit trier ! C'est une quantité insensée à traiter normalement, et c'est là que les méthodes de sélection de fonctionnalités sont utiles. Ils vous permettent de réduire le nombre de fonctionnalités incluses dans un modèle sans sacrifier la puissance prédictive. Les fonctionnalités redondantes ou non pertinentes peuvent en fait avoir un impact négatif sur les performances de votre modèle, il est donc nécessaire (et utile) de les supprimer.

Le principal avantage de la sélection des fonctionnalités est qu'elle réduit le sur ajustement. En supprimant les données superflues, cela permet au modèle de se concentrer uniquement sur les caractéristiques importantes des données et de ne pas s'attarder sur les caractéristiques qui n'ont pas d'importance. Un autre avantage de la suppression des informations non pertinentes est qu'elle améliore la précision des prédictions du modèle. Cela réduit également le temps de calcul nécessaire pour obtenir le modèle. Avoir un plus petit nombre de fonctionnalités rend votre modèle plus interprétable et facile à comprendre. Dans l'ensemble, la sélection des fonctionnalités est essentielle pour pouvoir prédire les valeurs avec n'importe quelle précision. Les techniques de réduction de la dimensionnalité combinent souvent des caractéristiques pour réduire les dimensions, tandis que les techniques de sélection de caractéristiques éliminent les attributs sans affecter le reste. Nous allons entrer dans une explication des méthodes de prétraitement des données et les techniques de sélection de caractéristiques.

Le prétraitement des données est une étape du processus d'exploration et d'analyse de données qui prend des données brutes et les transforme en un format pouvant être compris et analysé par les ordinateurs et l'apprentissage automatique.

Les données brutes du monde réel sous forme de texte, d'images, de vidéos, etc., sont désordonnées. Non seulement il peut contenir des erreurs et des incohérences, mais il est souvent incomplet et n'a pas une conception régulière et uniforme.

Les machines aiment traiter des informations claires et ordonnées, elles lisent les données sous forme de 1 et de 0. Il est donc facile de calculer des données structurées, comme des nombres entiers et des pourcentages. Cependant, les données non structurées, sous forme de texte et d'images doivent d'abord être nettoyées et formatées avant l'analyse.

## 3.2.1 Étapes de prétraitement des données

### 3.2.1.1 Évaluation de la qualité des données

Examinez attentivement vos données et faites-vous une idée de leur qualité globale, de leur pertinence par rapport à votre projet et de leur cohérence. Il existe un certain nombre d'anomalies de données et de problèmes inhérents à surveiller dans presque tous les ensembles de données, par exemple :

- **Types de données incompatibles** : lorsque vous collectez des données à partir de nombreuses sources différentes, elles peuvent vous parvenir sous différents formats. Bien que l'objectif ultime de tout ce processus soit de reformater vos données pour les machines, vous devez toujours commencer avec des données formatées de la même manière.
- **Valeurs de données mixtes** : peut-être que différentes sources utilisent des descripteurs différents pour les fonctionnalités. Ces descripteurs de valeur doivent tous être uniformisés.
- **Données aberrantes** : les valeurs aberrantes peuvent avoir un impact énorme sur les résultats de l'analyse des données.
- **Données manquantes** : recherchez les champs de données manquants, les espaces vides dans le texte ou les questions d'enquête sans réponse. Cela peut être dû à une erreur humaine ou à des données incomplètes. Pour prendre soin des données manquantes, vous devrez effectuer un nettoyage des données.

### 3.2.1.2 Nettoyage des données

Le nettoyage des données consiste à ajouter des données manquantes et à corriger, réparer ou supprimer des données incorrectes ou non pertinentes d'un ensemble de données. Le nettoyage des données est l'étape la plus importante du prétraitement, car il garantit que vos données sont prêtes à répondre à vos besoins.

Le nettoyage des données corrigera toutes les données incohérentes que vous avez découvertes lors de votre évaluation de la qualité des données. Selon le type de données avec lesquelles vous travaillez, il existe un certain nombre de nettoyeurs possibles dont vous aurez besoin pour traiter vos données.

- **Données manquantes** : Il existe plusieurs façons de corriger les données manquantes, mais les deux plus courantes sont : **ignorer les tuples** et **Remplir manuellement les données manquantes**.
- **Données bruyantes** : le nettoyage des données comprend également la correction des données « bruyantes ». Il s'agit de données qui incluent des points de données inutiles, des données non pertinentes et des données plus difficiles à regrouper.

### 3.2.1.3 Transformation des données

Avec le nettoyage des données, nous avons déjà commencé à modifier nos données, mais la transformation des données commencera le processus de transformation des données dans le(s) format(s) approprié(s) dont vous aurez besoin pour l'analyse et d'autres processus en aval.

Cela se produit généralement dans un ou plusieurs des cas ci-dessous :

- **Agrégation** : l'agrégation de données combine toutes vos données dans un format uniforme.
- **Normalisation** : la normalisation met vos données à l'échelle dans une plage régularisée afin que vous puissiez les comparer plus précisément.
- **Sélection de fonctionnalité**: la sélection des fonctionnalités est le processus consistant à décider quelles variables (fonctionnalités, caractéristiques, catégories, etc.) sont les plus importantes pour votre analyse. Ces fonctionnalités seront utilisées pour former des modèles ML. Il est important de se rappeler que plus vous choisissez d'utiliser de fonctionnalités, plus le processus de formation est long et, parfois, moins vos résultats sont précis, car certaines caractéristiques peuvent se chevaucher ou être moins présentes dans les données.
- **Discréditation** : la discréditation regroupe les données dans des intervalles plus petits. C'est un peu similaire au binning, mais cela se produit généralement après que les données ont été nettoyées.
- **Génération de la hiérarchie des concepts** : la génération de hiérarchie de concepts peut ajouter une hiérarchie dans et entre vos entités qui n'était pas présente dans les données d'origine.

#### 3.2.1.4 Réduction de donnée

Plus vous travaillez avec des données, plus il sera difficile de les analyser, même après les avoir nettoyées et transformées. En fonction de votre tâche à accomplir, vous pouvez en fait disposer de plus de données que nécessaire. Surtout lorsque l'on travaille avec l'analyse de texte, une grande partie de la parole humaine ordinaire est superflue ou sans rapport avec les besoins du chercheur. La réduction des données rend non seulement l'analyse plus facile et plus précise, mais réduit également le stockage des données.

Cela aidera également à identifier les caractéristiques les plus importantes du processus en cours.

- **Sélection d'attribut** : semblable à la discréditation, la sélection d'attributs peut intégrer vos données dans des pools plus petits.
- **Réduction de la numérosité** : cela facilitera le stockage et la transmission des données. Vous pouvez utiliser un modèle de régression, par exemple, pour n'utiliser que les données et les variables pertinentes pour votre analyse.
- **Réduction de la dimensionnalité** : Ceci, encore une fois, réduit la quantité de données utilisées pour faciliter l'analyse et les processus en aval.

### 3.2.2 Méthodes embarquées (Embedded Methods)

Les méthodes embarquées effectuent la sélection des fonctionnalités dans le cadre du processus de création du modèle. La régression Lasso et Ridge sont les deux méthodes de sélection de caractéristiques les plus courantes de ce type, et l'arbre de décision crée également un modèle utilisant différents types de sélection de caractéristiques.



Parfois, vous souhaitez peut-être conserver toutes les caractéristiques de votre modèle final, mais vous ne voulez pas que le modèle se concentre trop sur un seul coefficient. **La régression Ridge** peut le faire en pénalisant les coefficients bêta d'un modèle pour être trop grands. Fondamentalement, il réduit la force de la corrélation avec des variables qui peuvent ne pas être aussi importantes que d'autres. La régression de Ridge est effectuée en ajoutant un terme de pénalité à la fonction de coût de la régression. Le terme de pénalité prend tous les bêtas et les met à l'échelle par un terme lambda ( $\lambda$ ) qui doit être ajusté (généralement avec validation croisée : compare le même modèle mais avec différentes valeurs de lambda). Lambda est une valeur comprise entre 0 et l'infini, bien qu'il soit bon de commencer avec des valeurs comprises entre 0 et 1. Plus la valeur de lambda est élevée, plus les coefficients sont rétrécis. Lorsque lambda est égal à 0, le résultat sera un modèle régulier des moindres carrés ordinaires sans pénalité.

**La régression Lasso** est un autre moyen de pénaliser les coefficients bêta dans un modèle et est très similaire à la régression Ridge. Il ajoute également un terme de pénalité à la fonction de coût d'un modèle, avec une valeur lambda qui doit être ajustée. La distinction la plus importante par rapport à la régression Ridge est que la régression Lasso peut forcer le coefficient bêta à zéro, ce qui supprimera cette caractéristique du modèle. C'est pourquoi Lasso est parfois préféré, en particulier lorsque vous cherchez à réduire la complexité du modèle. Plus le nombre de fonctionnalités d'un modèle est petit, plus sa complexité est faible. Afin de forcer les coefficients à zéro, le terme de pénalité ajouté à la fonction de coût prend la valeur absolue des termes bêta au lieu de la mettre au carré, ce qui, en essayant de minimiser le coût, peut annuler le reste de la fonction, conduisant à un bêta égal à zéro.

Une remarque importante pour la régression Ridge et Lasso est que toutes vos fonctionnalités doivent être standardisées. De nombreuses fonctions dans Python et R le font automatiquement, car le lambda doit être appliqué de manière égale à chaque fonctionnalité. Avoir une caractéristique avec des valeurs en milliers et une autre avec des valeurs décimales ne permettra pas que cela se produise, d'où l'exigence de normalisation.

Un autre moyen courant de modéliser des données avec une sélection de caractéristiques est appelé arbre de décision, qui peut être soit un arbre de régression, soit un arbre de classification selon que la variable de réponse est continue ou discrète, respectivement. Cette méthode crée des fractionnements dans l'arbre en fonction de certaines caractéristiques pour créer un algorithme permettant de trouver la variable de réponse correcte. La façon dont l'arborescence est construite utilise une **méthode Wrapper** à l'intérieur d'une méthode embarquée. Ce que nous entendons par là, c'est que lors de la création du modèle d'arbre, la fonction intègre plusieurs méthodes de sélection de fonctionnalités. A chaque découpage, la fonction utilisée pour créer l'arbre essaie tous les découpages possibles pour toutes les caractéristiques et choisit celui qui découpe les données dans les groupes les plus homogènes. En termes clairs, il choisit la caractéristique qui peut le mieux prédire ce que sera la variable de réponse à chaque point de l'arbre. Il s'agit d'une **méthode Wrapper** car elle essaie toutes les combinaisons possibles de fonctionnalités, puis choisit la meilleure.

Les caractéristiques les plus importantes dans la prédiction de la variable de réponse sont utilisées pour faire des divisions près de la racine (début) de l'arbre, et les caractéristiques les plus

non pertinentes ne sont pas utilisées pour faire des divisions jusqu'à proximité des nœuds de l'arbre (extrémités). De cette façon, l'arbre de décision pénalise les caractéristiques qui ne sont pas utiles pour prédire la variable de réponse. Une fois qu'un arbre a été créé, il est possible de revenir en arrière et de « tailler » certains des nœuds qui ne fournissent aucune information supplémentaire au modèle. Cela empêche le sur ajustement et se fait généralement par validation croisée avec un ensemble de tests d'exclusion.

### 3.2.3 Méthode Wrapper (Wrapper Methods)

Les méthodes d'encapsulation calculent des modèles avec un certain sous-ensemble de fonctionnalités et évaluent l'importance de chaque fonctionnalité. Ensuite, ils itèrent et essaient un sous-ensemble différent de fonctionnalités jusqu'à ce que le sous-ensemble optimal soit atteint. Deux inconvénients de cette méthode sont le temps de calcul important pour les données avec de nombreuses caractéristiques et le fait qu'elle a tendance à sur ajuster le modèle lorsqu'il n'y a pas une grande quantité de points de données. Les méthodes Wrapper les plus notables de sélection de fonctionnalités sont *la sélection avant, la sélection arrière et la sélection pas à pas*.

*La sélection avant* commence avec zéro entité, puis, pour chaque entité individuelle, exécute un modèle et détermine la valeur de **p** associée au **test t** ou au **test F** effectué. Il sélectionne ensuite la fonctionnalité avec la valeur de **p** la plus faible et l'ajoute au modèle de travail. Ensuite, il prend la première fonctionnalité sélectionnée et exécute des modèles avec une deuxième fonctionnalité ajoutée et sélectionne la deuxième fonctionnalité avec la valeur de **p** la plus faible. Ensuite, il prend les deux fonctionnalités précédemment sélectionnées et exécute des modèles avec une troisième fonctionnalité et ainsi de suite, jusqu'à ce que toutes les fonctionnalités qui ont des valeurs **p** significatives soient ajoutées au modèle. Toutes les fonctionnalités qui n'ont jamais eu de valeur **p** significative lorsqu'elles ont été essayées dans les itérations seront exclues du modèle final.

*La sélection arrière* commence par toutes les entités contenues dans le jeu de données. Il exécute ensuite un modèle et calcule une **valeur de p** associée au **test t** ou au **test F** du modèle pour chaque caractéristique. La caractéristique avec la plus grande valeur de **p** non significative sera alors supprimée du modèle et le processus recommencera. Cela continue jusqu'à ce que toutes les entités avec des valeurs de **p** non significatives soient supprimées du modèle.

*La sélection pas à pas* est un hybride de sélection avant et arrière. Il commence par zéro entité et ajoute l'entité avec la valeur **p** significative la plus faible, comme décrit ci-dessus. Ensuite, il parcourt et trouve la deuxième caractéristique avec la valeur **p** significative la plus faible. À la troisième itération, il recherchera la fonctionnalité suivante avec la valeur **p** significative la plus faible, et il supprimera également toutes les fonctionnalités ajoutées précédemment qui ont maintenant une valeur **p** insignifiante. Cela permet au modèle final d'avoir toutes les fonctionnalités incluses significatives.

### 3.2.4 Sélection de fonctionnalités basée sur l'arborescence

L'importance des caractéristiques est une propriété importante qui accompagne les classificateurs arborescents tels que les arbres de décision ou les forêts aléatoires. La propriété du

modèle vous donne l'importance de chaque caractéristique avec le jeu de données et nous permet de déterminer l'influence de la caractéristique sur la prédiction. Plus le score de la caractéristique est élevé, plus l'influence sur la sortie positive est importante. Les caractéristiques sans importance sont rejetées en fonction d'une valeur seuil.

### **3.2.5 Sélection de fonctionnalités basées sur la régression lasso (L1)**

La sélection de caractéristiques basée sur **L1** utilise les coefficients des modèles de régression pour la sélection et l'interprétation des caractéristiques. L'idée derrière cette technique est que les caractéristiques qui ne sont pas corrélées à la variable de prédiction auront leurs coefficients proches ou égaux à zéro et les caractéristiques importantes auraient des coefficients élevés. Les modèles linéaires pénalisés par la norme **L1** produisent des solutions creuses. Comme chaque coefficient non nul contribue à la pénalité, la régularisation **L1** force les caractéristiques faibles à zéro et sont rejetées. Pour la classification, des modèles de régression logistique ou de LinearSVC sont utilisés, tandis que pour la régression, le modèle Lasso est utilisé. Les paramètres C ou alpha contrôlent la rareté ou le nombre d'entités sélectionnées.

### **3.2.6 Apprentissage en profondeur**

L'apprentissage automatique consiste à ce que les ordinateurs soient capables d'effectuer des tâches sans être explicitement programmés... mais les ordinateurs pensent et agissent toujours comme des machines. Leur capacité à effectuer certaines tâches complexes en recueillant des données à partir d'une image ou d'une vidéo, par exemple, est encore bien en deçà de ce dont les humains sont capables.

Les modèles d'apprentissage en profondeur introduisent une approche extrêmement sophistiquée de l'apprentissage automatique et sont prêts à relever ces défis car ils ont été spécifiquement modélisés d'après le cerveau humain. Des « réseaux de neurones profonds » complexes et multicouches sont construits pour permettre aux données d'être transmises entre les nœuds (comme les neurones) de manière hautement connectée. Le résultat est une transformation non linéaire des données qui est de plus en plus abstraite.

Bien qu'il faille d'énormes volumes de données pour « alimenter et construire » un tel système, il peut commencer à générer des résultats immédiats, et il y a relativement peu besoin d'intervention humaine une fois que les programmes sont en place.

### 3.2.6.1 Types d'algorithmes d'apprentissage en profondeur

#### ❖ Réseau de neurones convolutifs (ou CNN)

Les réseaux de neurones convolutifs sont des algorithmes spécialement conçus pour fonctionner avec des images. La «convolution» dans le titre est le processus qui applique un filtre basé sur le poids à chaque élément d'une image, aidant l'ordinateur à comprendre et à réagir aux éléments de l'image elle-même.

Cela peut être utile lorsque vous avez besoin de numériser un volume élevé d'images pour un élément ou une fonctionnalité spécifique ; par exemple, des images du fond de l'océan pour les signes d'un naufrage ou une photo d'une foule pour le visage d'une seule personne.

#### ❖ Réseau de neurones récurrents

Les réseaux de neurones récurrents, quant à eux, introduisent un élément clé dans l'apprentissage automatique qui est absent des algorithmes plus simples : la mémoire. L'ordinateur est capable de garder à l'esprit les points de données et les décisions passées et de les prendre en compte lors de l'examen des données actuelles, ce qui introduit la puissance du contexte.

Cela a fait des réseaux de neurones récurrents un axe majeur pour le travail de traitement du langage naturel. Comme avec un humain, l'ordinateur fera un meilleur travail pour comprendre une section de texte s'il a accès au ton et au contenu qui l'ont précédé. De même, les indications routières peuvent être plus précises si l'ordinateur "se souvient" que toute personne suivant un itinéraire recommandé un samedi soir prend deux fois plus de temps pour se rendre à destination.

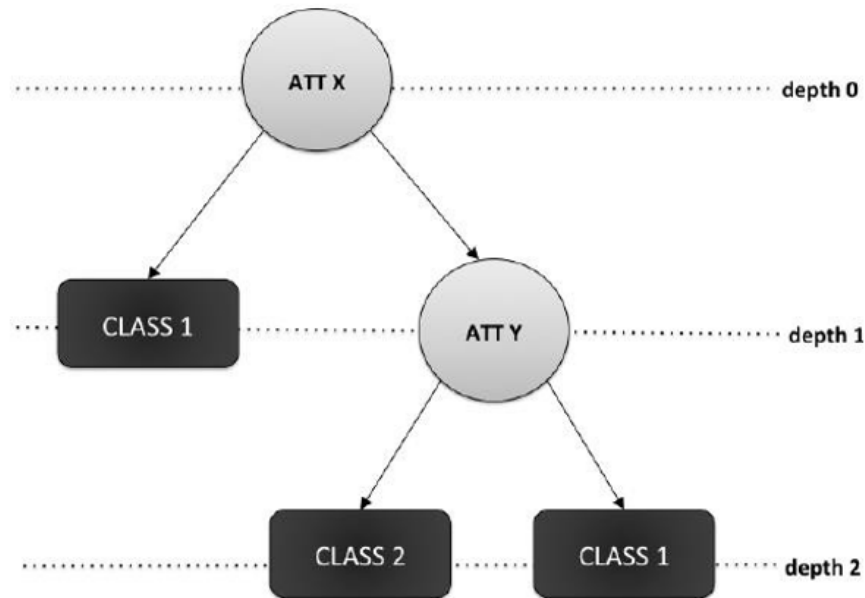
### 3.2.7 Algorithmes d'apprentissage automatique

La section suivante traite des divers apprentissages automatiques utilisés dans les expériences.

#### 3.2.7.1 Classificateur d'arbre de décision

Dans un modèle d'apprentissage automatique populaire, les arbres de décision utilisent une structure arborescente pour représenter les décisions. Ils sont construits dans une structure descendante avec l'utilisation de mesures telles que l'impureté et l'information de Gini. Il calcule l'importance de chaque entité et l'utilise pour diviser les éléments en sous-ensembles homogènes. Les nœuds représentent la condition du fractionnement et les nœuds terminaux représentent la

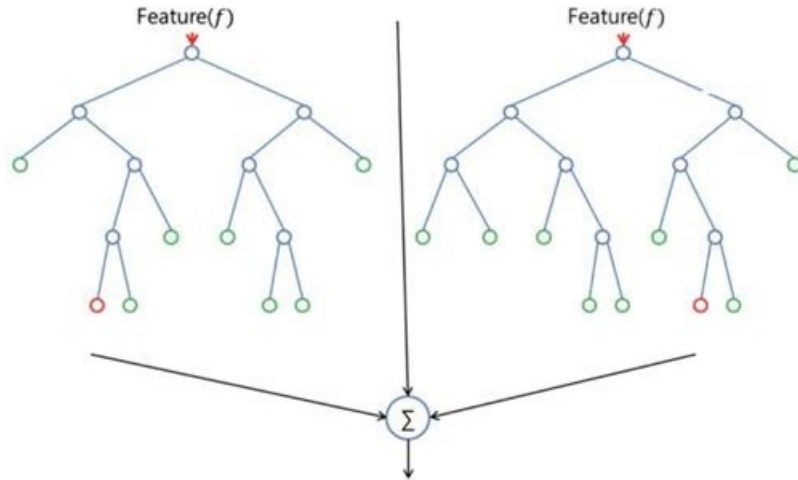
décision ou la sortie prévue. Les branches ou les bords de l'arborescence dirigent vers l'une des variables en sortie. Les arbres de décision sont modélisés à la fois pour les problèmes de classification et de régression. Bien que l'arbre de décision soit facile à comprendre, il a tendance à surdimensionner car il continue de se diviser en attributs et s'entraîne de manière critique sur les données d'entraînement. Pour éviter un surajustement, l'arbre de décision est généralement élagué pour l'empêcher de pousser trop profondément.



**Figure 3 Structure générale de l'arbre de décision pour la classification**

### 3.2.7.2 Classificateur aléatoire de Forrest

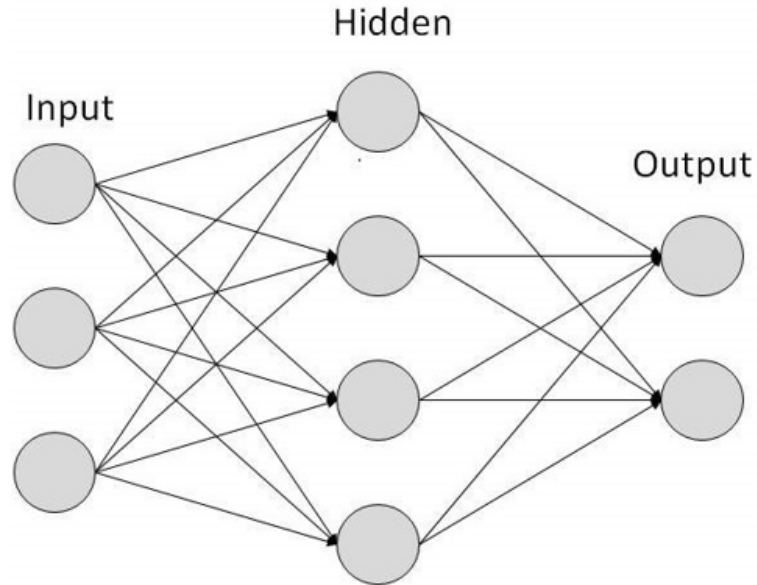
Random Forest Classifier construit une forêt qui est un ensemble d'arbres de décision. Il crée un ensemble d'arbres de décision à partir d'un sous-ensemble sélectionné au hasard des données d'apprentissage et agrège la décision de tous les arbres pour décider de la sortie finale. Cette technique est robuste car elle empêche la sortie bruitée de certains arbres d'affecter la décision finale et évite le surapprentissage. Il annule le biais en faisant la moyenne de toutes les prédictions. La forêt aléatoire est différenciée des arbres de décision car elle ne recherche pas la meilleure caractéristique lors de la division du nœud. Il recherche à la place la fonctionnalité la plus appropriée à partir d'un sous-ensemble de fonctionnalités. Cela fournit de la diversité et du caractère aléatoire à l'algorithme. L'algorithme peut être facilement modélisé à la fois pour des problèmes de classification et de régression.



**Figure 4 Structure générale des forêts aléatoires pour la classification**

### 3.2.7.3 Réseau neuronal artificiel

Le réseau de neurones artificiels est un système inspiré de la structure biologique des neurones du cerveau. La structure peut être définie comme un ensemble de neurones connectés organisés en couches consécutives. La couche d'entrée agit comme la première couche qui introduit les données dans le réseau. La couche cachée est constituée de neurones artificiels qui reçoivent un ensemble d'entrées pondérées et appliquent une fonction d'activation pour produire une sortie. Il peut y avoir plusieurs couches cachées dans un réseau, ce qui le rend capable de résoudre des problèmes complexes. La sortie d'une couche de neurones est transmise en tant qu'entrée à la couche suivante et est appelée réseau d'anticipation. La couche de sortie fournit la sortie prédictive finale.



**Figure 5 Structure générale du réseau de neurones artificiels**

L'activation fonctionne au cœur de la logique des réseaux de neurones et définit la sortie du neurone à partir d'une entrée ou d'un ensemble d'entrées. Voici les différentes fonctions d'activation:

### Sigmoide

La fonction sigmoïde a une courbe caractéristique en forme de «S» et est largement utilisée dans la classification binaire. La fonction génère une sortie de probabilité comprise entre 0 et 1 pour un ensemble d'entrées.

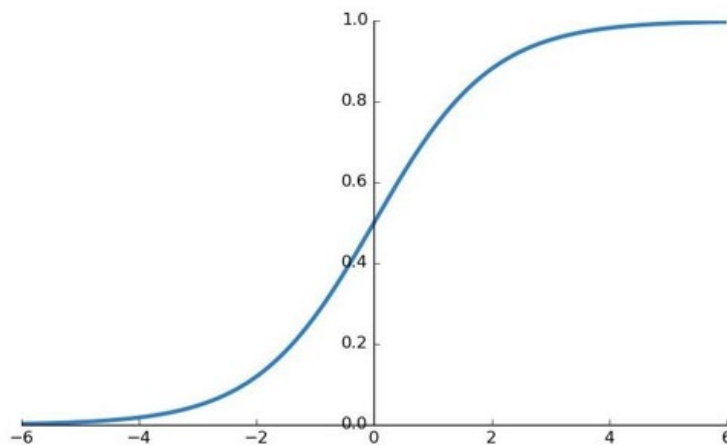


Figure 6 Courbe sigmoïde

### Unités linéaires rectifiées (ReLU)

Les unités linéaires rectifiées sont le plus souvent utilisées dans les couches cachées des réseaux de neurones artificiels. La fonction est telle que si l'entrée est inférieure à zéro, la sortie est 0 et si l'entrée est supérieure à zéro, elle donne l'entrée elle-même comme sortie.

$$ReLU(x) = \max(0, x)$$

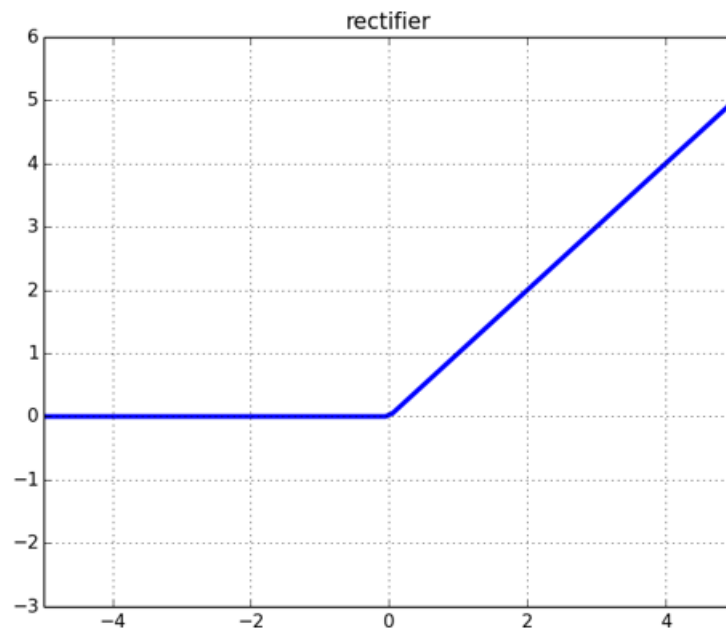


Figure 7 Courbe ReLU

### Fonction Softmax

La fonction d'activation softmax est utilisée pour la classification multi classe. Il calcule la distribution de probabilité de chaque classe sur toutes les classes cibles possibles. Sur la base des probabilités calculées, il détermine la sortie pour un ensemble donné d'entrées.

$$Softmax(x) = \frac{e^j}{\sum_i e^i}$$



### 3.2.7.4 Réseau de neurones récurrents

Les réseaux de neurones récurrents sont les seuls réseaux dotés d'une mémoire interne, ce qui les rend robustes et puissants. Le réseau peut être précis sur la prochaine prédiction car il a la capacité de mémoriser des bits significatifs de l'entrée grâce à la mémoire interne. Cela rend le modèle hautement préférable pour former des données séquentielles telles que du texte, de l'audio, de la vidéo et des séries temporelles. Un réseau feed-forward n'a aucune mémoire de l'entrée précédente et ne fonctionne que sur l'entrée actuelle. Dans un réseau de neurones récurrents, le courant est pris en compte avec les apprentissages passés. Les pondérations sont appliquées à la fois à l'entrée de courant et à la sortie de bouclage et sont ajustées par descente de gradient ou propagation en arrière.

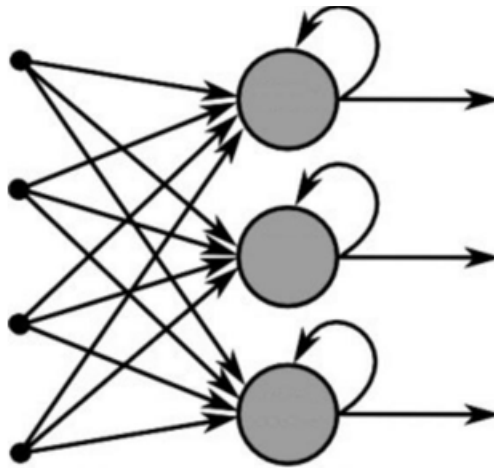


Figure 8 Structure générale du réseau de neurones récurrents

- **Mémoire à long court terme**

Les réseaux de neurones récurrents ont généralement une mémoire courte et sont étendus par des unités LSTM pour étendre la mémoire du réseau. Il permet au réseau de mémoriser les entrées sur une plus longue période de temps, ce qui en fait une unité essentielle dans les couches du réseau neuronal récurrent. Il permet d'absorber plus d'informations à partir de séquences de données encore plus longues. Cela aide à augmenter la précision de la prédiction en prenant en compte plus de données.

### 3.3 Conclusion

Anguita et al a utilisé un capteur pour capturer les signaux à un taux constant à 50 Hz, puis filtré les signaux pour la réduction du bruit. Ils ont prétraité les signaux en appliquant un filtre médian et un filtre Butterworth passe-bas du troisième ordre avec une fréquence de coupure de 20 Hz. Ils ont divisé le signal d'accélération en accélération corporelle et en accélération gravitationnelle. Ils ont ensuite échantillonné les signaux sur une fenêtre fixe d'une durée de 2,56 secondes avec un chevauchement de 50 %. Ils ont ensuite calculé un total de 561 caractéristiques à partir des signaux inertiels fenêtrés. Ils ont divisé au hasard l'ensemble de données dans un ratio de 70:30. L'ensemble de données de test a 2947 fenêtres de données et l'ensemble de données formé a 7352 fenêtres de données. L'histogramme des fenêtres d'activité prédites par HAR devient clair au fur et à mesure que les séries chronologiques sont construites.

En somme il est très important de faire une étude approfondie sur les méthodes que nous comptons choisir et voir celle qui s'adapte au mieux à notre système afin d'avoir de meilleurs résultats. Dans le prochain chapitre, nous allons décrire les technologies et outils de développement utilisé pour l'implémentation de notre application mobile.

# **Chapitre IV :**

# **Implémentation**

## 4.1 Technologies et outils de développement

- **Pandas**

Pandas, une bibliothèque Python open source, fournit un framework pour construire des données sous forme de tableau et effectuer des transformations de ligne, de colonne et de cellule. C'est un outil utile pour charger, analyser et extraire des données pour obtenir des informations et les structurer sous une forme à utiliser par les algorithmes d'apprentissage automatique.

- **Numpy**

NumPy, une bibliothèque Python open source, est utilisée avec la bibliothèque pandas pour gérer des données multidimensionnelles et effectuer des opérations scientifiques et mathématiques complexes sur les données.

- **Scikit-learn**

Scikit - learn est la bibliothèque d'apprentissage automatique open source de Python. Il offre la possibilité de créer facilement divers algorithmes de régression, de classification et de regroupement. Il permet de créer des pipelines et de valider la sortie avec une variété de métriques d'évaluation. La librairie permet aussi toujours de personnaliser les algorithmes en hyper ajustant les paramètres des modèles. Il contient également des algorithmes qui sont utilisés pour prétraiter les données, extraire des caractéristiques et réduire les dimensions.

- **Keras**

Keras est un API de réseau neuronal open source en Python qui peut s'exécuter sur Theano, CNTK et Tensorflow. Il fournit une couche d'abstraction aux complexités de la création d'un réseau neuronal et qui augmente le processus d'expérimentation rapide. Il est bien adapté pour créer des réseaux récurrents et des réseaux convolutifs.

## 4.2 Les images d'implémentation de l'application mobile

# HAR Project

## Functionalities



Recognition



Data Acquisition

## Settings



Sound

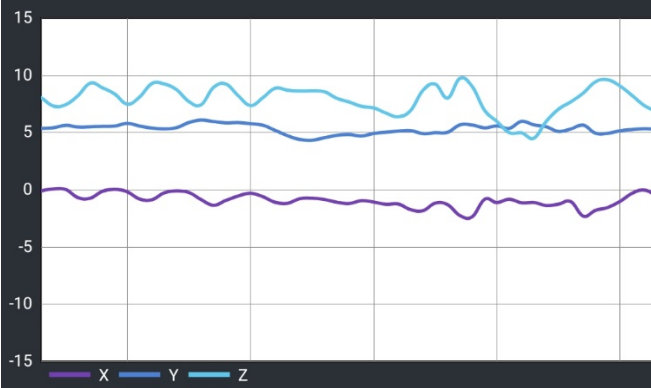


Contact to notify

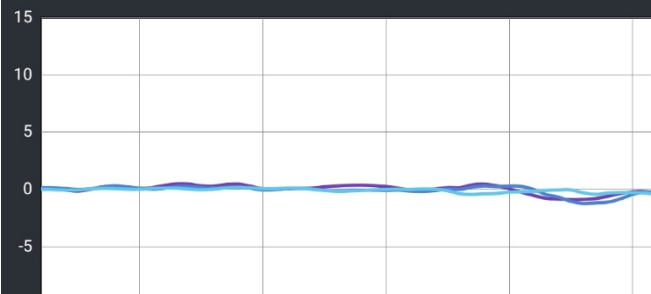


# REALTIME PLOTS

Acceleration



Gyroscope



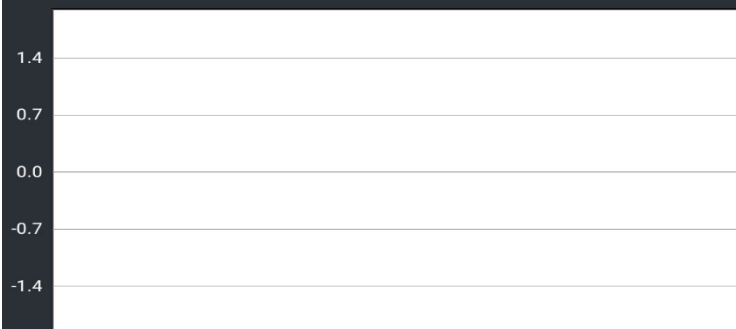


## Human Activity Recognition

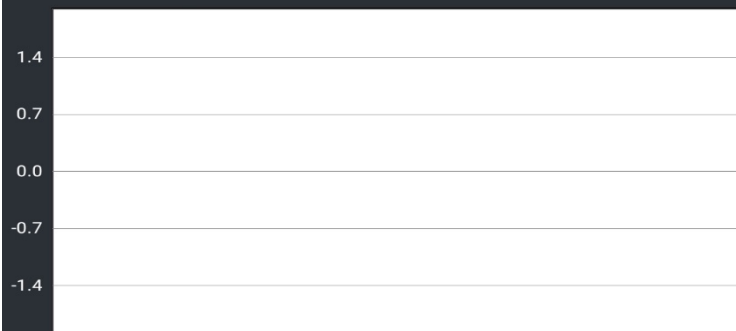


# PREPROCESSED PLOTS

Acceleration



Gyroscope



## **4.2 Conclusion**

Sur ce chapitre nous avons discuté des technologies et outils de développement et nous avons montré les images de l'implémentation de l'application Android de la reconnaissance des activités humaines à travers nos téléphones intelligents.

Utiliser les technologies qu'il faut nous permet d'avoir d'excellents résultats. Dans la prochaine section nous discuterons des résultats obtenus.



# **Chapitre V :**

## ***Expérimentation et résultats***

## 5.1 L'analyse des données

L'ensemble de données était converti dans un fichier CSV et importé dans une trame de données pandas. Le seul prétraitement requis consistait à combiner les sujets et les colonnes d'activité à leurs fonctions fenêtrées respectives. Lors de l'exploration, il a été constaté que l'ensemble de données ne comportait aucune valeur manquante. L'ensemble de données a été exploré plus avant pour comprendre les diverses caractéristiques et leurs effets sur les activités. L'ensemble de données d'apprentissage contient un total de 7352 observations ou fenêtres de données. Il a un total de 561 caractéristiques de temps et de fréquence où chaque observation correspond à l'une des 6 activités de la classe ambulatoire. Les 6 activités sont les suivantes :

- Classe 1 – marche
- Classe 2 - monté à l'étage
- Classe 3 - descendre les escaliers
- Classe 4 - s'asseoir
- Classe 5 - debout
- Classe 6 - allongé

L'ensemble de données a été étudié pour vérifier l'équilibre entre les observations de l'activité effectuées par les 30 sujets. Le comptage et l'analyse graphique ont montré que la distribution des classes variait de 13 % à 19 % pour les données d'entraînement et de test. Bien que la distribution des données ne soit pas la même pour toutes les activités, elles sont étroitement équilibrées. Le tableau 1 nous donne la distribution de chaque activité de classe pour les données d'entraînement et de test. La **figure 9** trace le nombre d'activités pour l'ensemble de données d'apprentissage et la figure 10 affiche le nombre d'activités enregistrées pour les données de test. Les graphiques indiquent clairement que l'activité « allonger » a le nombre maximum d'observations enregistrées et « descendre les escaliers » a le minimum. Cette observation peut également être observée dans l'ensemble de données d'entraînement.

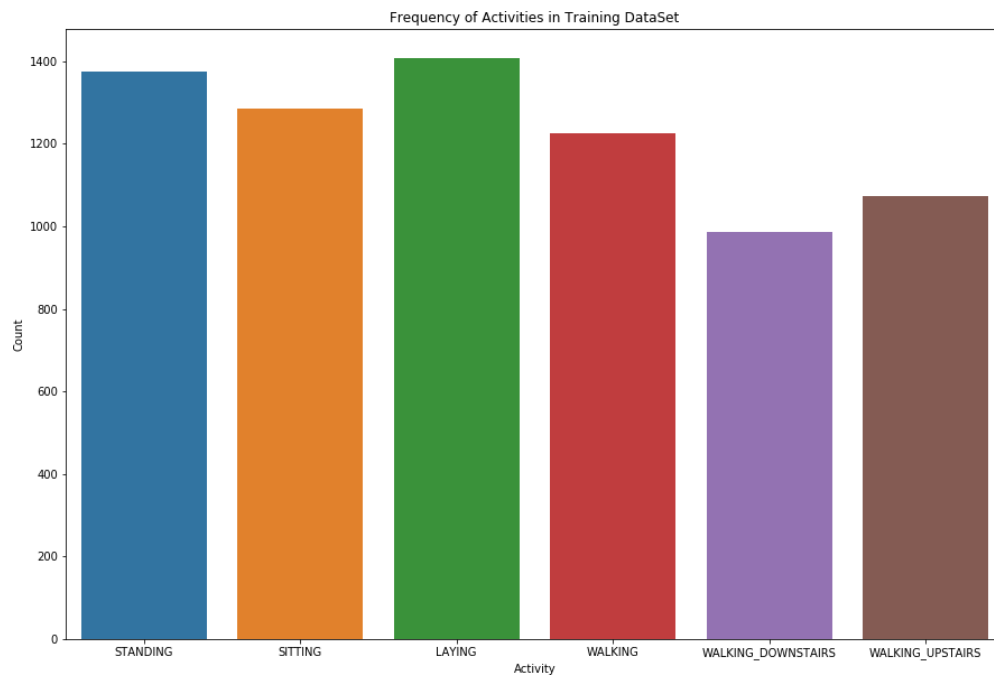
<b>Données d'entraînement</b>		
<b>Classe</b>	<b>Décompte</b>	<b>Pourcentage</b>
Marche	1226	16.676
Monté à l'étage	1073	14.595
Descendre les escaliers	986	13.411
S'asseoir	1286	17.492
Debout	1374	18.689
Allonger	1407	19.138

<b>Données de test</b>		
<b>Classe</b>	<b>Décompte</b>	<b>Pourcentage</b>
Marche	496	16.831
Monté à l'étage	471	15.982
Descendre les escaliers	420	14.252
S'asseoir	491	16.661
Debout	532	18.052
Allonger	537	18.222

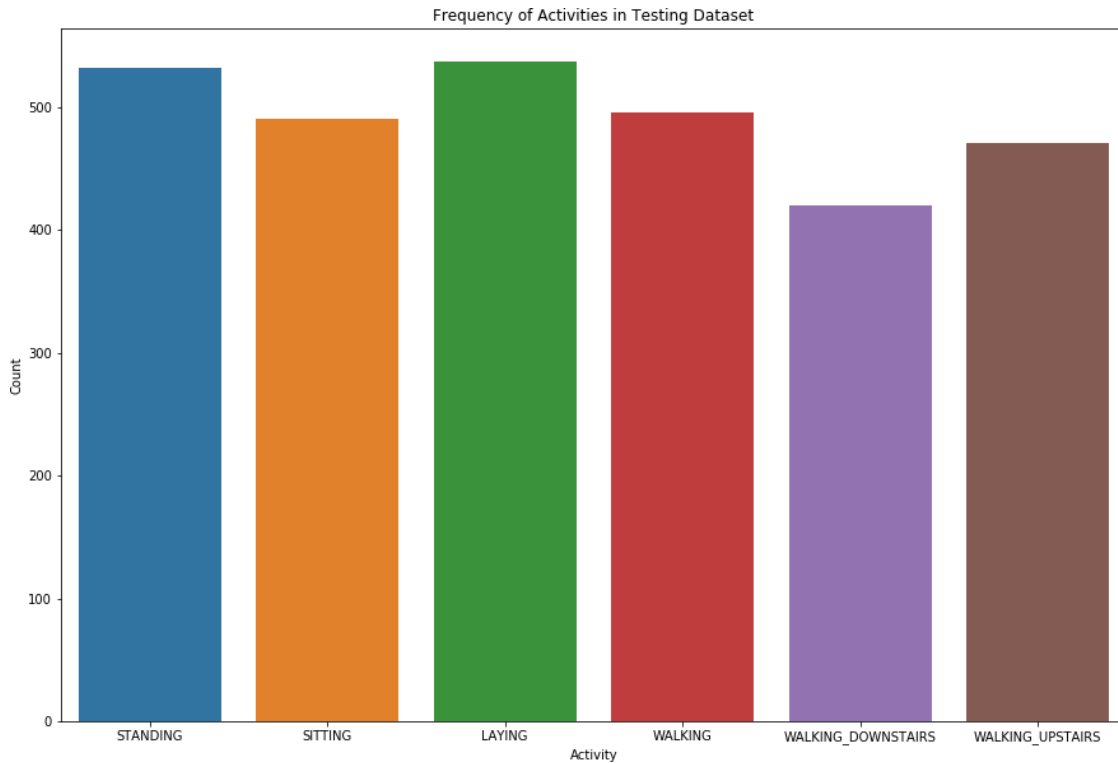
<b>Données d'entraînement et de test</b>		
<b>Classe</b>	<b>Décompte</b>	<b>Pourcentage</b>
Marche	1722	16.72
Monté à l'étage	1544	14.992
Descendre les escaliers	1406	13.652
S'asseoir	1777	17.254
Debout	1906	18.507
Allonger	1944	18.876

**Tableau 1. Répartition en pourcentage des activités**

Bien que les observations pour chaque activité ne soient pas exactement égales, l'ensemble de données fournit globalement une distribution bien équilibrée des observations d'activité. Même après la séparation des données dans l'ensemble de données d'entraînement et de test, l'équilibre dans l'observation reste vrai. Le décompte des observations pour certaines activités telles que monter les escaliers ou descendre les escaliers permet de comprendre que les sujets n'ont pas réalisé la tâche pendant un temps précis mais plutôt sur une distance fixe. Différents styles de marche ont conduit à enregistrer un nombre différent d'observations pour chaque sujet.

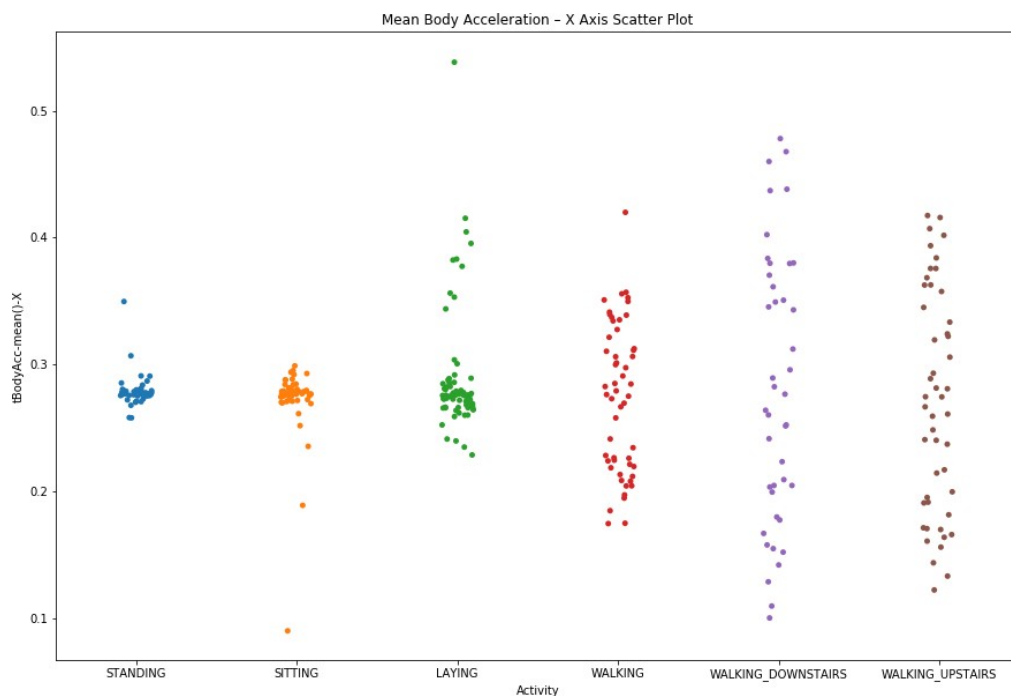


**Figure 9** Fréquence des activités dans l'ensemble de données d'entraînement

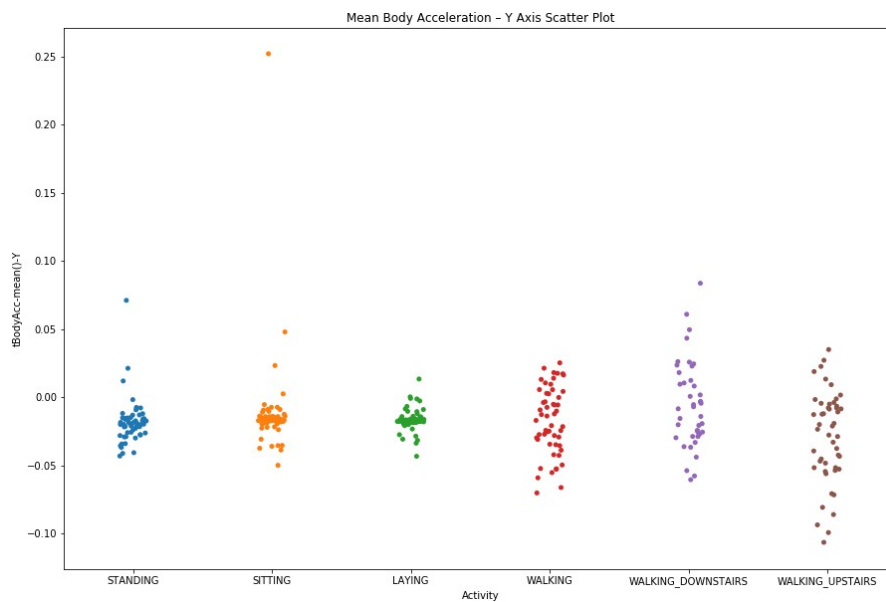


**Figure 10 Fréquence des activités dans l'ensemble de données de test**

La prochaine série d'analyses est effectuée pour comprendre la variation des données pour chaque activité. Les expériences ont été réalisées sur les données enregistrées uniquement pour le sujet 15. La caractéristique de « l'accélération moyenne du corps » le long de l'axe X, Y, Z a été documentée sous forme de graphique en nuage de points. Le graphique de la Figure 11 et de la Figure 12 indique que la valeur moyenne de l'accélération du corps est plus variable pour les activités de marche, monter les escaliers et descendre les escaliers que les activités passives de s'asseoir, de se tenir debout et de s'allonger.



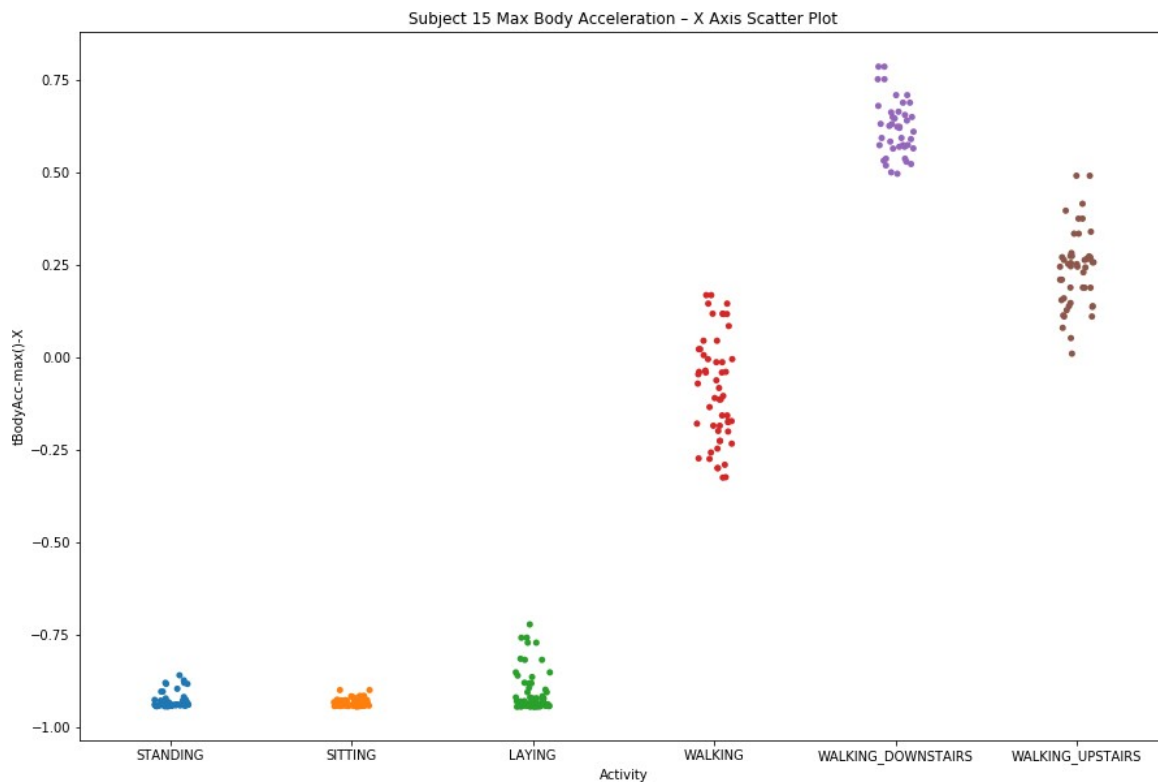
**Figure 11 Accélération moyenne du corps - Diagramme de dispersion de l'axe X**



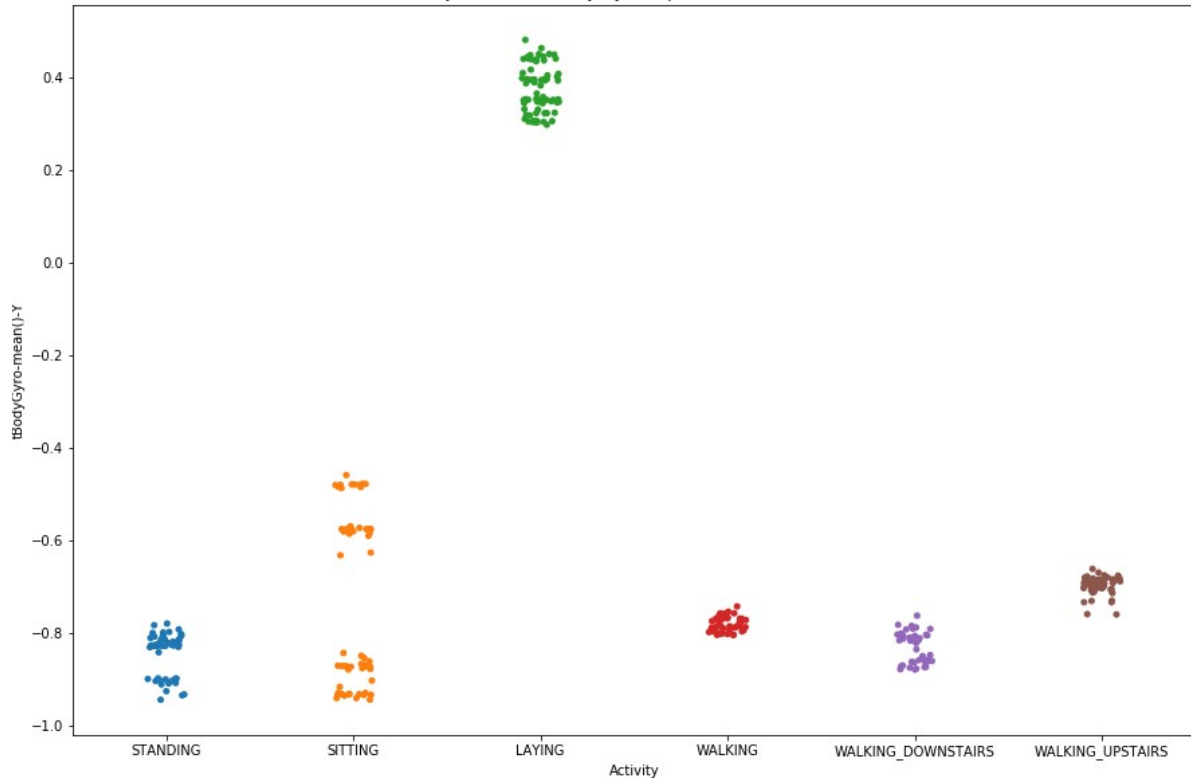
**Figure 12 Accélération moyenne du corps - Diagramme de dispersion de l'axe Y**

La caractéristique d'accélération maximale le long des axes X, Y et Z pour le sujet 15 est tracée et analysée ensuite. L'analyse indique qu'il existe une distinction claire dans les valeurs maximales entre les activités passives et actives, car toutes les activités passives sont inférieures aux activités actives. L'analyse de la figure 13 révèle que les valeurs le long de l'axe X peuvent

nous aider à différencier la marche, monter les escaliers et descendre les escaliers, mais ne fournit aucun aperçu des activités passives. Cela fournit une certaine indication que l'accélération seule n'est pas suffisante pour la reconnaissance de l'activité ambulatoire, mais les données d'un capteur différent tel qu'un gyroscope aideraient à différencier les activités passives. La figure 14 est le graphique tracé pour la caractéristique « Angle (X, gravité moyenne) ». L'intrigue montre une distinction claire pour l'activité "Allonger" des autres classes. De la même manière, d'autres caractéristiques conçues fournissent des informations importantes sur la reconnaissance des activités humaines.



**Figure 13 Accélération maximale du corps - Diagramme de dispersion de l'axe X**



**Figure 14 Diagramme de dispersion de l'angle (X, gravité moyenne)**



## 5.2 Évaluation du modèle d'apprentissage automatique

### 5.2.1 Sélection des fonctionnalités et évaluation du modèle de classification

Une analyse comparative a été effectuée sur quatre classificateurs pour comprendre leur amélioration dans le modèle après la sélection des caractéristiques. Les quatre algorithmes de classification expérimentés sont le classificateur d'arbre de décision, le classificateur de forêt aléatoire, le classificateur d'amplification de gradient et le réseau de neurones artificiels. Les résultats ont été comparés à la métrique du temps nécessaire pour construire et former le modèle et la précision du modèle. Nous utilisons les méthodes de sélection d'entités basées sur l'arborescence et de sélection d'entités basées sur L1 pour réduire les dimensions de l'ensemble de données d'entités. Parmi les 561 entités du jeu de données d'origine, 91 entités ont été sélectionnées par la méthode de sélection d'entités basée sur l'arborescence et 108 entités ont été sélectionnées par la méthode de sélection d'entités basée sur L1. Le tableau 2 répertorie la méthode de sélection des fonctions et les dimensions des fonctions résultantes.

Méthode de sélection des fonctionnalités	Fonctionnalités sélectionnées
Sélection de fonctionnalités basée sur l'arborescence	Forme du jeu de données (7352, 91)
Sélection de fonctionnalités basée sur L1	Forme du jeu de données (7352, 108)

**Tableau 2 Méthode de sélection des caractéristiques et dimensions résultantes**

La sélection de fonctionnalités basée sur L1 conserve plus de fonctionnalités que la méthode basée sur l'arborescence et le temps d'exécution est bien inférieur à la méthode de sélection de fonctionnalités basée sur l'arborescence. Pour la sélection de caractéristiques basée sur L1, la variable C avait une valeur de 0,01. Un réglage plus poussé de ce paramètre aurait pu permettre de sélectionner des caractéristiques plus distinctes et importantes. Pour construire les modèles du classificateur de forêt aléatoire et du classificateur d'amplification de gradient, une valeur de 200 a été définie pour la variable « n\_estimateurs ». Lors de l'expérimentation avec des valeurs de 100, 200, 300, 400 et 500 pour la variable 'n\_estimateurs', il a été remarqué que la précision n'augmentait pas après la valeur de 200 et a donc été considérée comme le cas de base pour un ensemble d'expériences. Le réseau de neurones artificiels a été construit à l'aide de la bibliothèque Keras en utilisant Tensorflow comme backend. Le réseau a été configuré pour avoir deux couches cachées avec 40 unités cachées dans chacune avec la fonction d'activation "ReLU"

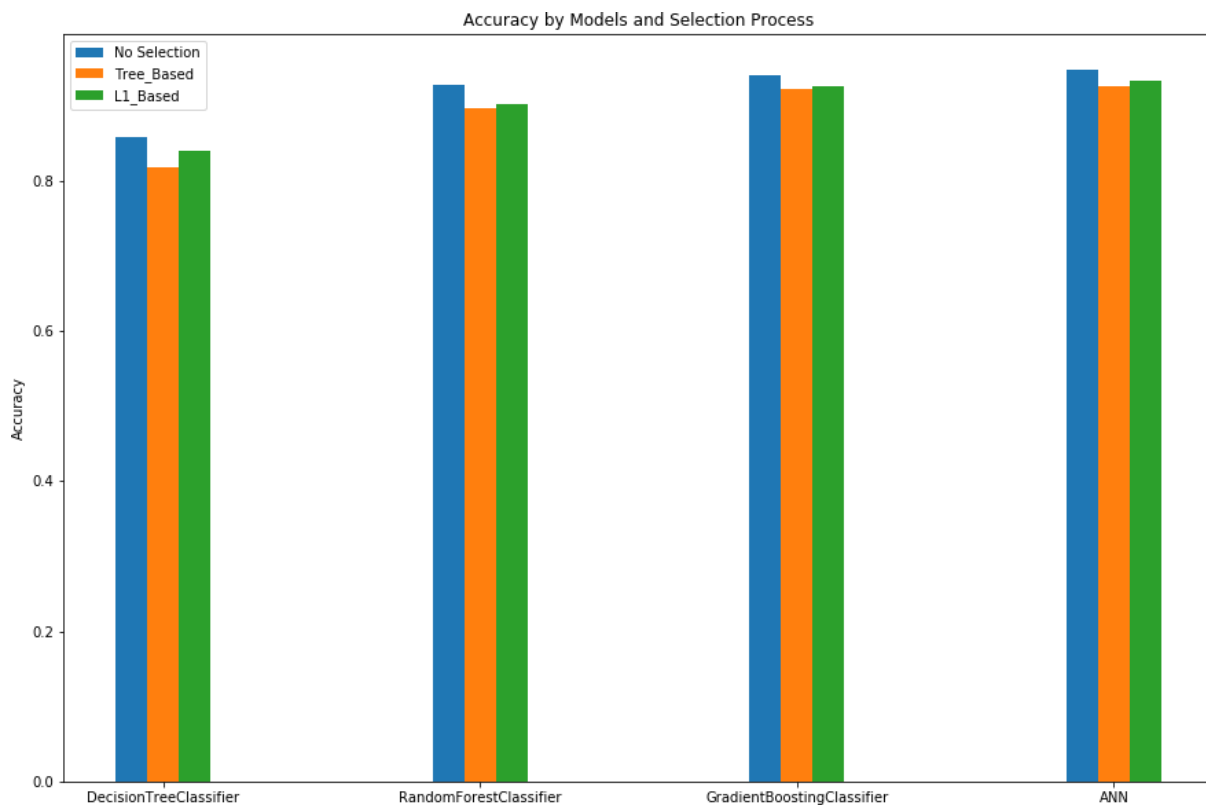
pour chaque neurone. La couche de sortie a été configurée pour utiliser une activation "Softmax" et l'optimiseur "Adam" a été utilisé pour améliorer la précision. Le modèle a été compilé pour exécuter plus de 500 époques avec une taille de lot de 20 en utilisant "l'entropie croisée catégorielle" comme fonction de perte. Le tableau 3 résume la précision de chaque modèle et le tableau 4 résume le temps de construction et de prédiction pour chaque modèle de classification.

<b>Résumé de la précision du modèle</b>			
	<b>Aucune sélection de fonctionnalités</b>	<b>Sélection de fonctionnalités basée sur l'arborescence</b>	<b>Sélection de fonctionnalités basée sur L1</b>
<b>Classificateur d'arbre de décision</b>	85.78%	82.91%	84.08%
<b>Classificateur de forêt aléatoire</b>	92.80%	89.71%	90.22%
<b>Classificateur d'amplification de gradient</b>	94.06%	92.29%	92.60%
<b>Réseaux de neurones artificiels</b>	94.77%	92.63%	93.41%

**Tableau 3 Résumé du modèle et de la précision**

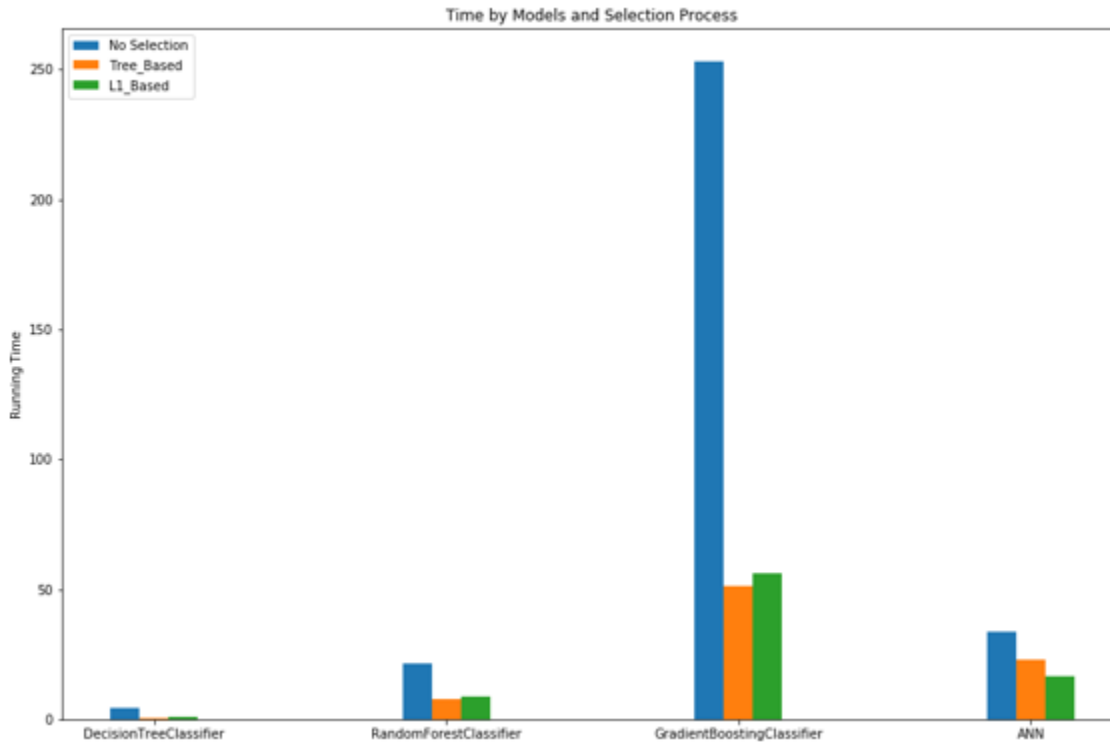
Récapitulatif du temps de construction du modèle			
	Aucune sélection de fonctionnalités	Sélection de fonctionnalités basée sur l'arborescence	Sélection de fonctionnalités basée sur L1
<b>Classificateur d'arbre de décision</b>	4.279s	0.724s	0.759s
<b>Classificateur de forêt aléatoire</b>	21.44s	8.019s	8.733s
<b>Classificateur d'amplification de gradient</b>	252.89s	51.39s	56.10s
<b>Réseaux de neurones artificiels</b>	33.74s	22.79s	16.65s

**Tableau 4 Récapitulatif du temps de génération et de prédiction**



**Figure 15 Précision par modèles et processus de sélection**

Le graphique de la Figure 15 indique qu'avec la sélection des caractéristiques, la précision ne diminue pas considérablement. Cela indique que la précision globale du modèle n'est pas compromise par la réduction de la taille de l'ensemble de données. Les méthodes de sélection de caractéristiques sont des techniques solides pour sélectionner des caractéristiques percutantes et également empêcher le surajustement. L'amélioration et le réglage des paramètres des techniques de sélection de caractéristiques devraient également fournir les capacités d'améliorer les performances. Le graphique de la Figure 16 montre une diminution du temps nécessaire à la construction du modèle et à la prédiction des valeurs avec l'utilisation de la sélection de caractéristiques. Cela indique une augmentation de l'efficacité, sans mettre en danger la précision du modèle. Pour les classificateurs tels que les arbres de décision, il n'y a pas de changement drastique du temps d'exécution. Mais pour les modèles plus complexes tels que les classificateurs d'amplification de gradient et les réseaux de neurones artificiels, il existe une différence de temps significative avec et sans sélection de fonctionnalités. Le classificateur d'amplification de gradient fonctionne de manière additive, les fonctionnalités étant permutées et divisées de manière aléatoire à chaque étape. Il continue le processus jusqu'à ce qu'il améliore la fonction de perte. Le temps de construction du classificateur est considérablement réduit lorsqu'un plus petit ensemble de données contenant les caractéristiques importantes est entré dans le modèle. Le modèle se construit plus rapidement car il nécessite moins d'itérations pour améliorer la précision. De même, pour le réseau de neurones artificiels, des données moins trompeuses aideraient à améliorer plus rapidement la fonction de perte et un vecteur d'entrée plus petit nécessiterait un plus petit nombre de calculs à chaque neurone. Nous ne voyons pas de différence de temps critique pour les arbres de décision car le modèle calcule d'abord le gain d'informations de chaque caractéristique et divise l'arbre sur les fonctionnalités où le gain d'informations est le maximum. Il exécute le même processus pour l'ensemble de données sélectionné et, par conséquent, nous ne constatons pas d'amélioration substantielle du temps de construction.



**Figure 16 Temps par modèles et processus de sélection**

### 5.2.2 Réseau de mémoire longue à court terme (LSTM)

Un modèle LSTM est développé pour reconnaître l'activité humaine à partir des signaux inertiels bruts au lieu d'utiliser des données techniques. Cette expérience a été menée pour tester la capacité du réseau à apprendre des caractéristiques à partir d'une séquence de données de séries chronologiques et valider s'il peut reconnaître les activités humaines à partir des caractéristiques qu'il extrait. Les signaux inertiels bruts sont constitués de trois signaux principaux, l'accélération du corps, l'accélération totale et le gyroscope du corps, chaque attribut enregistrant des données sur les trois axes (X, Y et Z). Pour une fenêtre de 2,56 secondes ou 128 pas de temps, 9 variables sont enregistrées pour donner un total de 1152 éléments (9\*128) pour chaque ligne de données. Les RNN-LSTM conviennent aux données de séries chronologiques car ils ont la capacité d'apprendre et de se souvenir sur de longues séquences de données. L'ensemble de données « Reconnaissance des activités humaine » convient parfaitement à un LSTM car il est destiné à être utilisé avec des séquences de données, jusqu'à 400 pas de temps. Au lieu d'ingénierie manuelle des caractéristiques, le RNN-LSTM apprend directement des signaux de la série chronologique et peut obtenir des résultats comparables aux modèles qui sont construits sur des données d'ingénierie des

caractéristiques. Chaque axe de chaque signal est stocké dans un fichier différent, avec un total de 9 fichiers chacun pour les données d'apprentissage et de test. Les données d'entraînement et de teste sont chargées dans un seul tableau NumPy tridimensionnel séparé (échantillons, pas de temps, caractéristiques). La fonctionnalité de la bibliothèque NumPy permet d'empiler des fonctionnalités et de créer un seul tableau 3D. La variable de sortie, qui est un nombre entier représentant l'une des 6 activités, est codée à chaud pour la rendre adaptée à un modèle de réseau neuronal multi-classes. Le modèle de RNN-LSTM est construit à l'aide de la bibliothèque Keras et est défini comme un modèle Keras séquentiel. Le modèle a d'abord une seule couche de LSTM cachée qui est utilisée pour extraire les caractéristiques de la séquence de données d'entrée. Une couche d'abandon est ajoutée au modèle pour réduire le surajustement sur les données d'apprentissage. Ensuite, une couche dense entièrement connectée est ajoutée au modèle qui interprète les caractéristiques, suivie d'une couche de sortie qui donne les prédictions finales de l'activité humaine. L'« entropie croisée catégorielle » est utilisée comme fonction de perte avec un optimiseur « Adam » pour augmenter la précision et optimiser le réseau. L'« entropie croisée catégorielle » est une fonction de perte couramment utilisée pour les tâches de classification, tandis que l'optimiseur « Adam » est bien adapté aux grands ensembles de données car ils sont efficaces en termes de calcul et nécessitent moins de mémoire. La fonction d'activation "ReLU" est utilisée dans la couche dense avec une fonction d'activation "Softmax" pour la sortie. La fonction d'activation ReLU surmonte le problème des gradients de fuite et est donc préférée aux fonctions d'activation Sigmoid et Tanh. La fonction d'activation "Softmax" fournit une distribution de probabilité sur toutes les classes et réagit bien aux simulations basses et hautes. Le modèle est exécuté pour un total de 15 époques (epochs) avec une taille de lot de 64 échantillons. Généralement, les données de séquence ne sont pas mélangées pour un LSTM, mais pour cette expérience particulière, les fenêtres de données de séries chronologiques sont mélangées. Ceci est effectué car l'objectif est d'apprendre des fonctionnalités à travers des pas de temps dans une fenêtre et non à travers plusieurs fenêtres.

Comme les réseaux de neurones sont stochastiques, il est difficile de juger l'évaluation à partir d'une seule exécution et, par conséquent, le LSTM est évalué 10 fois au total. Les résultats sont résumés dans le tableau 5. Le modèle fonctionne bien, atteignant une précision globale de 90,19 % avec un écart type de 0,994. La précision nous montre que le modèle de LSTM à une seule couche évaluée sur des entrées de signal brut est au même niveau que d'autres modèles de

classification qui sont construits sur des données d'ingénierie de caractéristiques. Le tableau 6 fournit un tableau de comparaison de la précision obtenue par différents modèles construits sur l'ensemble de données d'ingénierie et de la précision obtenue par le LSTM sur les données brutes du signal inertiel. Le modèle de LSTM atteint une plus grande précision que l'article original [26] qui a publié une précision de 86 % en utilisant un algorithme de classification de machine à vecteurs de support.

<b>RNN - Évaluation du modèle LSTM</b>	
<b>Cycle d'évaluation</b>	<b>Précision</b>
1	90.13%
2	88.70%
3	90.09%
4	91.42%
5	91.65%
6	89.41%
7	89.79%
8	89.72%
9	91.69%
10	89.31%
<b>Agrégat</b>	<b>90.19%</b>

**Tableau 5 Évaluation du modèle des RNN- LSTM sur 10 exécutions.**

Tableau de comparaison de précision				
Jeu de données d'ingénierie des fonctionnalités				Raw Inertial Signals Dataset
Arbre de décision	Forêt aléatoire	Classificateur d'amplification de gradient	Réseau de neurones artificiels	Les RNN-LSTM
85.78%	92.80%	94.06%	94.77%	90.19%

**Tableau 6 Tableau de comparaison de précision**

A travers les résultats obtenus en évaluant le LSTM 10 fois et en calculant la moyenne, nous pouvons dire que le LSTM fonctionne bien car il a une précision globale de 90.19 % ce qui montre qu'il est très efficace.

- L'Arbre de décision a une précision de 85.78% ;
- La Forêt aléatoire a une précision de 92.80% ;
- Le Classificateur d'amplification de gradient a une précision de 94.06% ;
- Le Réseau de neurones artificiels a une précision de 94.77% ;

### 5.3 Conclusion

L'ensemble de données a été étudié pour vérifier l'équilibre entre les observations de l'activité effectuées par les 30 sujets. Sans surprise, la distribution des données n'est pas la même pour toutes les activités. L'analyse graphique a montré que la répartition des classes variait de 13 % à 19 % pour les données d'entraînement et de test. Bien que la distribution des données ne soit pas la même pour toutes les activités, elles sont toujours équilibrées. L'ensemble de données fournit globalement une distribution bien équilibrée des observations d'activité. Même après avoir séparé les données dans l'ensemble d'entraînement et de test, l'équilibre dans l'observation est toujours là. Le décompte des observations pour certaines activités, comme monter ou descendre les escaliers, nous aide à comprendre que les sujets n'ont pas effectué la tâche pendant une période de temps spécifique, mais plutôt pendant une distance fixe. Une caractéristique d'accélération maximale le long des axes X, Y et Z est analysée pour faire la distinction entre marcher, monter les escaliers et



descendre les escaliers. La reconnaissance d'activité ne semble pas être en mesure d'identifier les activités passives. Il semble que les données d'un capteur différent comme un gyroscope aideraient à différencier les activités passives. De la même manière, d'autres caractéristiques conçues fournissent des informations importantes sur la reconnaissance des activités humaines. Après la sélection des caractéristiques, une analyse comparative a été effectuée sur quatre classificateurs - le classificateur d'arbre de décision, le classificateur de forêt aléatoire, le classificateur de renforcement de gradient et le réseau de neurones artificiels. Les résultats ont été comparés au temps nécessaire pour construire et former le modèle et à la précision du modèle.

Le modèle LSTM s'évalue bien, atteignant une précision globale de 90,19 % avec un écart type de 0,994. La précision montre que le réseau LSTM est similaire en termes de performances à d'autres modèles qui ont été construits sur des données d'ingénierie de fonctionnalités. Le modèle LSTM surpasse l'article original qui a publié une précision de 86 % en utilisant un algorithme de classification SVM modifié. Le tableau 6 compare la précision obtenue par l'article original et le réseau LSTM.

Le but de cette expérimentation est d'obtenir des résultats précis afin de mieux comprendre mon travail et son effet sur ceux qui m'intéressent. Dans le prochain chapitre, nous allons parler de la conclusion ainsi que les travaux futurs et la liste des références

# **Chapitre VI**

## ***Conclusion, Travaux futurs et Références***

## 6.1 Conclusion

Dans ce document de recherche, nous avons présenté l'architecture générale utilisée pour construire des systèmes de reconnaissance de l'activité humaine et souligné les problèmes de conception tels que la sélection des capteurs, l'obscurcissement, la flexibilité, etc. qui sont évalués indépendamment en fonction du type de système en cours de développement. Le document se concentre en outre sur l'importance de sélectionner des caractéristiques importantes à partir des données et fournit une analyse quantitative des mesures de temps d'exécution et de précision. Des méthodes de sélection de caractéristiques basées sur des arbres et sur L1 ont été utilisées pour sélectionner des caractéristiques importantes et ont été évaluées sur quatre modèles de classification. Les résultats indiquent que sans compromis sur la précision, le temps d'exécution et le coût de calcul sont considérablement réduits avec l'utilisation de méthodes de sélection de caractéristiques. De meilleures méthodes de sélection de caractéristiques et une amélioration du réglage des paramètres peuvent aider à améliorer davantage la précision et à réduire les coûts de calcul. Le document de recherche fournit également une solution pour réduire et éliminer la dépendance de l'exigence de connaissance du domaine pour créer des fonctionnalités artisanales à partir des signaux bruts obtenus à partir des données du capteur. Nous avons montré avec succès qu'avec l'utilisation du modèle de réseau LSTM, conçu pour s'entraîner sur les séquences de signaux inertiels bruts, les caractéristiques peuvent être apprises automatiquement par le réseau et une précision importante est obtenue. La précision obtenue grâce à l'utilisation d'un réseau neuronal récurrent sur les données de signal brutes est comparable à d'autres modèles de classification qui sont construits sur des fonctionnalités artisanales. L'ajout de couches supplémentaires au réseau ou l'augmentation de la complexité augmenterait encore la précision de la reconnaissance de l'algorithme d'apprentissage en profondeur.

## 6.2 Travaux Futurs

Pour que les systèmes de reconnaissance d'activités humains atteignent leur plein potentiel, des recherches supplémentaires sont nécessaires. La comparaison entre les systèmes de reconnaissance d'activités humains est entravée et devient non quantifiable car chaque chercheur utilise un ensemble de données différent pour la reconnaissance des activités. Un ensemble de

données public commun aiderait les chercheurs à comparer leurs systèmes et à faire évoluer le système dans son ensemble. Les activités reconnues dans les systèmes existants sont simples et atomiques, ce qui pourrait faire partie de comportements composites plus complexes. La reconnaissance d'activités composites peut enrichir la connaissance du contexte. Il existe également une excellente opportunité de recherche pour reconnaître les activités qui se chevauchent et qui sont concurrentes. En étendant les travaux menés sur les algorithmes d'apprentissage en profondeur, les réseaux de neurones convolutifs unidimensionnels et bidimensionnels, les hybrides de réseaux convolutifs et les LSTM devraient être étudiés plus avant pour déterminer leur aptitude à résoudre le problème de la reconnaissance de l'activité humaine à partir de données de signal brutes. Les systèmes de reconnaissance d'activités humains existants sont principalement axés sur les activités individuelles, mais pourraient être étendus davantage à la reconnaissance des modèles et des tendances d'activité pour un groupe de personnes utilisant les réseaux sociaux. Enfin, les systèmes de reconnaissance qui pourraient prédire les actions avant qu'elles n'aient lieu par l'utilisateur pourraient constituer une évolution révolutionnaire dans certaines applications.

## 6.3 References

- [1] A. Perez, M. Labrador, and S. Barbeau, "G-Sense: A Scalable Architecture for Global Sensing and Monitoring," *IEEE Network*, vol. 24, no. 4, pp. 57–64, 2010.
- [2] L. C. Jatoba, U. Grossmann, C. Kunze, J. Ottenbacher, and W. Stork, "Context-aware mobile health monitoring: Evaluation of different pattern recognition methods for classification of physical activity," in *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5250–5253, 2008.
- [3] G. Tanaka, M. Okada, and H. Minemo, "GPS-Based Daily Context Recognition for Lifelog Generation Using Smartphone," in *(IJACSA) Int. Journal of Advanced Computer Science and Applications*, vol. 6, no. 2, 2015.
- [4] M. Berchtold, M. Budde, D. Gordon, H. Schmidtke, and M. Beigl, "Actiserv: Activity recognition service for mobile phones," in *International Symposium on Wearable Computers*, pp. 1–8, 2010.
- [5] D. Riboni and C. Bettini, "Cosar: Hybrid Reasoning for Context-Aware Activity Recognition," *Personal and Ubiquitous Computing*, vol. 15, pp. 271–289, 2011.
- [6] T. Brezmes, J. Gorricho, and J. Cotrina, "Activity Recognition from Accelerometer Data on a Mobile Phone," in *Distributed Computing, Artificial Intell., Bioinformatics, Soft Computing, and Ambient Assisted Living*, vol. 5518 of *Lecture Notes in Computer Science*, pp. 796–799, Springer Berlin / Heidelberg, 2009.
- [7] L. Bao and S. S. Intille, "Activity Recognition from User-Annotated Acceleration Data," in *Pervasive*, pp. 1–17, 2004.

[8] Y. Hanai, J. Nishimura, and T. Kuroda, "Haar-Like Filtering for Human Activity Recognition Using 3D Accelerometer," in *IEEE 13th Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop*, pp. 675–678, 2009.

[9] Z. He and L. Jin, "Activity Recognition from Acceleration Data Using AR Model Representation and SVM," in *Int. Conf. on Machine Learning and Cybernetics*, vol. 4, pp. 2245–2250, 2008.

[10] R. Olszewski, C. Faloutsos, and D. Dot, "Generalized Feature Extraction for Structural Pattern Recognition," in *Time-Series Data*. 2001.

[11] O. Lara, M. Labrador, "A survey on human activity recognition using wearable sensors", *IEEE Common. Surveys Tuts.* vol. 15, no. 3, pp. 1192-1209, 2013.

[12] J. Kwapisz, G. Weiss and S. Moore, "Activity Recognition using Cell Phone Accelerometers," in *ACM SIGKDD Explorations Newsletter*, vol. 12, issue 2, Dec 2010

[13] C. Williams and J. Mathew, "An Architecture for Mobile Context Services," in *Lect. Notes in Electr. Eng.*, vol 313, Springer, Cham, 2015, pp. 61-68.

[14] Z. Wei, R.H. Deng, J. Shen, J. Zhu, K. Ouyang, and Y. Wu, "Multidimensional Context-Awareness in Mobile Devices," *MultiMedia Modeling: 21st Int. Conf. MMM 2015*, pp. 38-49, Jan. 2015.

[15] U. Maurer, A. Smailagic, D. P. Siewiorek, and M. Deisher, "Activity recognition and monitoring using multiple sensors on different body positions," in *Int. Workshop on Wearable and Implantable Body Sensor Networks*, (Washington, DC, USA), IEEE Computer Society, 2006.

[16] E. Tapia, S. Intille, W. Haskell, K. Larson, J. Wright, A. King, and R. Friedman, "Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart monitor," in *Int. Symposium on Wearable Computers*, 2007

[17] J. Parkka, M. Ermes, P. Korpijaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, "Activity classification using realistic data from wearable sensors," in *IEEE Trans. on Inf. Technol. in Biomedicine*, vol. 10, no. 1, pp. 119–128, 2006.

[18] A. Khan, Y.-K. Lee, S. Lee, and T.-S. Kim, "A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer," in *IEEE Trans. on Inf. Technol. in Biomedicine*, vol. 14, no. 5, pp. 1166–1172, 2010.

[19] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava, "Using mobile phones to determine transportation modes," *ACM Trans. on Sensor Networks*, vol. 6, no. 2, pp. 1–27, 2010.

[20] Mustapha Badshah "Sensor-based Human Activity Recognition using smartphones", 2019.

[21] Gabriele Gamberi, "Human Activity recognition (HAR) based on convolutional Neural Network", 2021

[22] O. D. Lara, A. J. Perez, M. A. Labrador, and J. D. Posada, "Centinela: A human activity recognition system based on acceleration and vital sign data," *Journal on Pervasive and Mobile Computing*, 2011.

[23] K. L. Huang, S. S. Kanhere, and W. Hu, "Preserving privacy in participatory sensing systems," *Computer Communications*, vol. 33, no. 11, pp. 1266–1280, 2010.

[24] I. J. Vergara-Laurens and M. A. Labrador, "Preserving privacy while reducing power consumption and information loss in lbs and participatory sensing applications," in *IEEE GLOBECOM*, 2011.

[25] M. Berchtold, M. Budde, H. Schmidtke, and M. Beigl, “An extensible modular recognition concept that makes activity recognition practical,” in *Advances in Artificial Intelligence, Lecture Notes in Computer Science*, pp. 400–409, Springer Berlin / Heidelberg, 2010.

[26] O. D. Lara and M. A. Labrador, “A mobile platform for real-time human activity recognition,” in *IEEE Conference on Consumer Communications and Networks*, 2012.

[27] C. Zhu and W. Sheng, “Human daily activity recognition in robot-assisted living using multi-sensor fusion,” in *IEEE International Conference on Robotics and Automation*, pp. 2154–2159, 2009.

[28] D. Anguita, A. Ghio, L. Oneto, X. Parra, J. L. Reyes-Ortiz, “Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine,” in *Ambient Assisted Living and Home Care, Lecture Notes in Computer Science*, vol 7657. Springer, Berlin, Heidelberg, 2012

[29] J. Wang, Y. Chen, S. Hao, X. Peng, L. Hu, “Deep learning for sensor-based activity recognition: A survey,” in *Pattern Recognition Letters*, vol. 119, pp. 3-11, Mar 2019. 4

[30] Ryan Farmar, Ning Han, Madeline McCombe, “Intro to Feature Selection Methods for Data Science”, 2019

[31] Jason Brownlee, “How to choose a Feature Selection Method for Machine Learning”, 2020

[32] Jason Brownlee, “A Gentle Introduction to long Short-Term Memory Networks by Experts”, 2021

[33] Michael Middleton, “Deep Learning vs. Machine Learning-what’s the difference”, 2021

[34] Tobias Geisler Mesevage, “What is Data Preprocessing and what are the steps involved?”, 2021