

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'INFORMATIQUE

Mémoire de Fin d'Etudes
De MASTER ACADEMIQUE
Domaine : Mathématiques et Informatique
Filière : Informatique

Spécialité : Réseaux, Mobilité et systèmes embarqués

Présenté par
OULD CHIKH Diana
MEDDANE Yasmina

Thème

**Classification du comportement des
vaches laitières à base de Réseaux de
Neurones Artificiels**

Mémoire soutenu 26/09/2018 devant le jury composé de:

Président : Mr M.DAOUI

Promotrice : Mme M.BELKADI

Examinatrice : Mme R.AOUJDIT

Invité : Mr B.ACHOUR

Remerciements

Qu'il nous soit permis avant toute chose d'exprimer notre reconnaissance envers toutes les personnes qui de près ou de loin nous ont soutenu fidèlement dans nos efforts et ont contribué ainsi à la réalisation de ce mémoire.

Nous réserveront une place particulière à Madame BELKADI notre promotrice, dont les qualités humaines et le savoir pratique nous ont permis d'illustrer ce travail par des simulations pertinentes.

Nous tenions à exprimer notre profonde gratitude à notre Co-encadreur, Monsieur ACHOUR Brahim pour nous avoir confié ce travail, pour son suivi, ses orientations et ses remarques pertinentes et précieuses.

Que les membres du jury trouveront ici nos plus vifs remerciements pour l'honneur qu'ils nous font en acceptant de juger notre travail.

Dédicaces

*Avant tout je remercie **Dieu** tout puissant de m'avoir donné la santé, la force et le courage afin de mener à bien ce modeste travail.*

Je dédie ce travail :

*Aux deux êtres les plus chers « **mes parents** » aucun hommage ne pourrait être à la hauteur pour exprimer ma reconnaissance envers tout ce que vous avez fait pour me voir réussir.*

A mes frères et sœurs qui n'ont cessé d'être pour moi des exemples de persévérance, de courage et de générosité.

A tous les membres de ma famille : tantes, oncles, cousins et cousines.

A la mémoire de mon regretté oncle Md Ourabah, qui nous a quitté il y a quatre ans.

*A ma copine, ma chère amie, mon binôme **Yasmine** avec laquelle j'ai partagé ce travail et qui a su me reconforter, me supporter dans les moments de stress, chose qui n'est pas toujours évidente.*

A toutes les personnes qui me sont et me seront toujours chères.

A tous ceux qui ont participés de près ou de loin à la réalisation de ce modeste travail.

M^{lle} Diana OULD CHIKH

Dédicaces

*Au nom du dieu le clément et le miséricordieux louange à **ALLAH** le tout puissant.*

Je dédie ce modeste travail en signe de respect, reconnaissance et de remerciement A :

Ma mère, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mes frères et sœurs qui n'ont cessé d'être pour moi des exemples de persévérance, de courage et de générosité.

*Ma chère amie **Diana** que j'adore de tout mon cœur car elle est à la hauteur du mot «Amitié», et pour son soutien moral, sa patience et sa compréhension tout au long de ce projet.*

Toutes mes chères amies.

Tous mes enseignants, je leurs exprime ma profonde gratitude.

Tous les étudiants de département informatique.

Et à tous ceux que j'aime.

M^{lle} Yasmina MEDDANE

Sommaire

Introduction générale.....	9
I. Chapitre: Etat de l'art.....	12
I.1 Introduction.....	13
I.2 Le besoin d'outils automatiques	14
I.2.1 Économiser la main d'œuvres	14
I.2.2 La première détection	14
I.2.3 L'amélioration du bien-être des animaux	14
I.2.4 Minimiser les pertes financières et maximiser la production et le profit	14
I.3 Le Comportement des vaches laitières	14
I.3.1 Quelque Comportement des vaches laitières	15
I.3.1.1 Position couchée	15
I.3.1.2 Position debout	15
I.3.1.3 Position perchée.....	15
I.3.1.4 Mouvements de lever et de coucher	15
I.3.1.5 Léchage sous la patte arrière et nettoyage.....	16
I.3.1.6 La Marche	16
I.3.2 L'objectif de la classification du comportement des vaches litières.....	16
I.3.2.1 La santé animale.....	16
I.3.2.2 Événement Chaleur / Œstrus	17
I.3.3 Maladies des vaches détectables par le comportement.....	17
I.3.3.1 La Mammite.....	17
I.3.3.2 Boiterie	18
I.4 Classification automatique	20
I.4.1 Les Etapes d'un processus de classification	20
I.4.1.1 Le prétraitement	21
I.4.1.2 Construction du modèle.....	22
I.4.1.2.1 Phase d'initialisation	22
I.4.1.2.2 Phase d'apprentissage.....	22
I.4.1.2.3 Phase de validation	23
I.4.1.3 Phase de test (évaluation).....	23

I.5	Les méthodes de classification.....	24
I.5.1	Logique floue	24
I.5.1.1	Avantages de la logique floue	25
I.5.1.2	Inconvénients de la logique floue	25
I.5.2	Les arbres de décisions.....	25
	Les arbres de décisions sont aussi utilisés dans divers domaines, tel que:	25
I.5.2.1	Avantage d'un arbre de décision.....	27
I.5.2.2	Inconvénients des arbres de décision	27
I.5.3	Machine à vecteurs de support (Support Vector Machine SVM)	27
I.5.3.1	Avantage des SVM.....	29
I.5.3.2	Inconvénients des SVM	29
I.5.4	Les réseaux de neurones	29
I.5.4.1	Avantage des réseaux de neurones	30
I.5.4.2	Inconvénients des réseaux de neurones	30
I.6	Conclusion	31
II.	Chapitre : Les réseaux neuronaux artificiels	32
II.1	Introduction.....	33
II.2	Pourquoi les réseaux de neurone ?.....	33
II.3	Du Neurone biologique au neurone artificiel.....	34
II.3.1	Neurone biologique.....	34
II.3.2	Neurone formel (artificiel).....	35
II.3.2.1	Structure d'un neurone artificiel.....	36
II.3.2.2	Les Entées	36
II.3.2.3	Poids de connexion.....	36
II.3.2.4	Fonction d'agrégation ou de sommation	37
II.3.2.5	Fonction de transfert.....	37
II.4	Les réseaux de neurones artificiels	38
II.4.1	Structure du réseau.....	38
II.5	Topologie des réseaux de neurones.....	40
II.5.1	Réseau multicouche	40
II.5.2	Réseau à connexions locales	40
II.5.3	Réseau à connexions récurrentes	41
II.5.4	Réseau à connexions complètes	41
II.6	Types de réseaux de neurone	41

II.6.1	Les réseaux non-boucles (FEED FORWARD).....	42
II.6.1.1	Réseau de neurones monocouche le perceptron	42
II.6.1.2	Le perceptron multicouche (PMC)	43
II.6.2	Les réseaux récurrents (FEED BACK)	44
II.6.2.1	Réseau de Hopfield(HNN).....	45
II.6.2.2	Les réseaux de neurones compétitifs.....	45
II.7	Apprentissage des réseaux de neurones.....	46
II.7.1	Type d'apprentissage	47
II.7.1.1	Apprentissage supervisé.....	47
II.7.1.2	Apprentissage non supervisé (ou auto-organisationnel)	47
II.7.1.3	Apprentissage hybride (mixte)	48
II.7.2	Règles d'apprentissage.....	48
II.7.2.1	Règle de correction d'erreurs.....	48
II.7.3	Algorithmes d'apprentissage.....	48
II.7.3.1	Algorithme de rétro propagation	48
II.8	Conclusion	50
 III. Chapitre: Système embarqué pour la détection du comportement des vaches laitières 51		
III.1	Introduction.....	52
III.2	Système embarqué.....	52
III.2.1	Architecture d'un système embarqué.....	53
III.2.1.1	Le fonctionnement du système.....	54
III.2.2	Contrôler à distance	54
III.2.3	Caractéristiques.....	55
III.2.4	Domaines d'application.....	55
III.2.4.1	Systèmes embarqués industriels.....	55
III.2.4.2	Électronique grand public.....	55
III.2.4.3	Objets connectés IoT(Internet of Things).....	55
III.3	Système réalisé pour la collecte des données des vaches	56
III.3.1	Description du système	56
III.3.1.1	Carte arduino NANO V 3.0.....	56
III.3.1.2	IMU (unité de mesure inertielle) 9DOF MPU9250	56
III.3.1.3	RTC ds1307	57
III.3.1.4	Interface I2C (Inter-Integrated Circuit), interface SPI (Serial Peripheral Interface): 58	

III.4	Fonctionnement du système.....	58
III.4.1	Le graphe de flux de données	58
III.4.2	Capture des données.....	59
III.4.2.1	Angles d'Euler	59
III.4.2.2	Quaternions.....	60
III.4.2.3	Algorithme de Madgwick	61
III.4.3	Graphe des Raffinements successives.....	61
III.5	Déploiement du système	62
III.6	Conclusion	63
IV.	Chapitre: Classification par Réseaux de Neurones Artificiel.....	64
IV.1	Introduction.....	65
IV.2	Développement d'un RNA pour la classification du comportement des vaches.....	65
IV.2.1	Outils de programmation	65
IV.2.2	Méthodologie et mise en œuvre.....	66
IV.2.3	Architecture générale du RNA.....	68
IV.2.3.1	La couche de sortie.....	68
IV.2.3.2	La couche d'entrée	69
IV.2.3.3	La couche cachée.....	69
IV.2.4	Fonctionnement	70
IV.2.5	Initialisation du RNA.....	70
IV.2.6	Apprentissage.....	70
IV.2.6.1	Paramétrage de la phase d'apprentissage	71
IV.2.6.2	Performances du réseau de neurones multicouche à rétropropagation.....	71
IV.2.6.2.1	Algorithme d'apprentissage.....	72
IV.2.6.3	Architecture des couches cachées :	73
IV.3	Résultats des tests effectués sur les données utilisant le model RNA construit :	74
IV.3.1	Test sur les données de deux vaches	75
IV.3.2	Résultat des Tests sur des données avec différentes fréquences.....	77
IV.3.2.1	Test sur le modèle RNA (MAGM)	77
IV.3.2.1.1	Matrice de confusion	78
IV.3.2.1.2	Performance du modèle	78
IV.3.2.2	Test sur le modèle RNA (MF)	79
IV.3.2.2.1	Matrice de confusion	80
IV.3.2.2.2	Performance du modèle	80

IV.4 Conclusion	82
Conclusion générale.....	84
Bibliographie	86

Liste des tableaux :

Tableau 1: Grille d'évaluation de la locomotion.....	19
Tableau 2: Comparaison entre les algorithmes d'apprentissage	72
Tableau 3: Résumé de l'effet du nombre de couches cachées sur la performance.....	73
Tableau 4: L'architecture du réseau de neurone à rétropropagation adapté à notre application.	74
Tableau 5: Performance du modèle RNA développé et testé sur les données de 2 vaches.....	76
Tableau 6: Matrice de confusion pour la classification des quatre comportements (debout, couché, transition debout-couché, transition couché-debout) avec des fréquences de données de (10, 20, 30, 40, 50, 60, 70, 80 Hz).Utilisant les données des trois capteurs.	78
Tableau 7: Performance du modèle RNA dans la classification des quatre modes du comportement..	79
Tableau 8: Matrice de confusion pour la classification des quatre comportements (debout, couché, transition debout-couché, transition couché-debout) avec des fréquences de données de (10, 20, 30, 40, 50, 60, 70, 80 Hz). utilisant les données du filtre.....	80
Tableau 9: Performance du modèle RNA dans la classification des quatre modes du comportement, le modèle à filtre.....	81

Liste des figures

Figure 1 : Frais directs liés aux mammites.....	18
Figure 2: Les pertes économiques liées aux boiteries	19
Figure 3: Différentes étapes du processus de classification	20
Figure 4:Algorithme de l'arbre de décision utilisé pour la classification des comportements chez les vaches laitières	26
Figure 5: L'arbre binaire de la Classification SVM.....	29
Figure 6: Mise en correspondance neurone biologique / neurone artificiel	35
Figure 7: Neurone artificiel	36
Figure 8: Un réseau de neurone artificiel	39
Figure 9: Réseau multicouches.....	40
Figure 10: Réseau a connexions locales.....	40
Figure 11: Réseau à connexions récurrentes	41
Figure 12: Réseau à connexion complète.....	41
Figure 13: Topologie des réseaux de neurones	42
Figure 14: Perceptron monocouche.....	43
Figure 15: Perceptron multicouche	44
Figure 16: Réseau de Hopfield à 5 neurones.....	45
Figure 17: Les réseaux à compétition.....	46
Figure 18: Apprentissage supervisé.....	47

Figure 19: Apprentissage non supervisé.....	47
Figure 20: Architecture d'un Système embarqué typique.	53
Figure 21: module RTC DS1307.....	57
Figure 22:Flux de données pour le système d'acquisition des données	59
Figure 23: Les angles d'Euler	60
Figure 24: Matrice de rotation.....	60
Figure 25: Rotation d'un objet	60
Figure 26: Calcul des Angles d'Euler à Partir du Quaternion.	61
Figure 27: Raffinement successives pour le système d'acquisition de données.	62
Figure 28 : ceinture placée sur le dos d'une vache laitière.....	63
Figure 29: Données d'une vache en position "debout"	66
Figure 30:Données d'une vache en position "couché"	67
Figure 31: Données d'une vache en transition debout-couché.....	67
Figure 32: Données d'une vache en transition couché debout	68
Figure 33: Architecture du RNA développée avec 3 couches cachées de [20 10 10] neurones.....	69
Figure 34: Matrice de confusion dont les lignes représentent les classes observées,	72
Figure 35: Les modèles RNA utilisés dans les tests.	75
Figure 36: Taux de classification pour toutes les fréquences	82

Résumé

Le bien-être des animaux est une question de grande importance dans les systèmes modernes de production alimentaire. Le suivi des troupeaux est essentiel car il permet aux agriculteurs d'avoir une vision globale de la santé du bétail et du temps de l'œstrus, ainsi un élevage réussi produira des génisses en meilleure santé, avec moins de risque d'être réformées jeunes. Ces animaux auront aussi plus de chance de vivre plus longtemps dans le troupeau, un bon élevage des génisses permettra aux éleveurs d'augmenter la productivité de leur troupeaux. Cependant, le problème avec ce suivi c'est que les éleveurs ne sont pas capables de surveiller les bovins pendant longtemps et d'élever beaucoup de ces derniers en particulier ; les troupeaux s'agrandissent mais la santé, les performances et le bien-être des animaux d'élevage nécessitent un suivi continu et de plus en plus précis. Parallèlement, les éleveurs ont besoin d'une autre organisation du travail. Par conséquent, la conception et le déploiement d'un système de suivi permettant de mesurer les paramètres comportementaux a fait l'objet de plusieurs études récentes.

Un système de collecte des paramètres comportementaux des vaches laitières a été réalisé par un doctorant au laboratoire LARI dont ces paramètres ont été mesurés à l'aide de différents types de capteurs. Dans ce travail nous proposons donc, de définir et d'utiliser les réseaux de neurones artificiels. L'objectif est de construire un modèle de classification performant en termes de précision et de taux de classification. Il sera utilisé pour classifier le comportement des vaches laitières en quatre modes de comportements (debout, couché, transition debout-couché, transition couché-debout).

Mots clés : Réseaux de neurones artificiels, vaches laitières, comportement, perceptron multicouches.

Introduction générale

Introduction générale

Introduction générale

Le bien-être des animaux est une question de grande importance dans les systèmes modernes de production alimentaire. Le suivi des troupeaux est essentiel car il permet aux agriculteurs d'avoir une vision globale de la santé du bétail et du temps de l'œstrus, ainsi un élevage réussi produira des génisses en meilleure santé, avec moins de risque d'être réformées jeunes. Ces animaux auront aussi plus de chance de vivre plus longtemps dans le troupeau, un bon élevage des génisses permettra aux éleveurs d'augmenter la productivité de leur troupeaux.

Cependant, le problème avec ce suivi c'est que les éleveurs ne sont pas capables de surveiller les bovins pendant longtemps et d'élever beaucoup de ces derniers en particulier ; les troupeaux s'agrandissent mais la santé, les performances et le bien-être des animaux d'élevage nécessitent un suivi continu et de plus en plus précis. Parallèlement, les éleveurs ont besoin d'une autre organisation du travail.

L'étude et l'observation de la faune ont toujours été un sujet de grand intérêt en étant donné que la mesure quantitative du comportement animal est un outil important pour la compréhension de leur reproduction, la survie, le bien-être et l'interaction avec d'autres animaux. Plusieurs études ont démontré que le comportement d'animaux d'élevage fournit des renseignements fiables sur l'état de santé et du bien-être des animaux.

Par conséquent, la conception et le déploiement d'un système de suivi permettant de mesurer les paramètres comportementaux a fait l'objet de plusieurs études récentes. Un système de collecte des paramètres comportementaux des vaches laitières a été réalisé par un doctorant au laboratoire LARI dont ces paramètres ont été mesurés à l'aide de différents types de capteurs.

Une fois les paramètres comportementaux sont mesurés et enregistrés sur une carte mémoire, les données ont besoin d'être traitées et transformées en comportements correspondants. Différentes méthodes de classification peuvent être utilisées telles que les arbres de décision, les supports vecteur machines (SVM), les réseaux de neurones (RNA) ...etc. Le but de ce travail est de développer un modèle RNA basé sur une architecture perceptron multicouches à rétropropagation qui a été initialisée et formée en utilisant les méthodes de Nguyen-Widrow et l'algorithme de rétropropagation de LevenbergMarquardt, respectivement, capable de classer le comportement des vaches laitières en quatre différents modes (Debout, Couché, Transition debout-couché, Transition couché-debout). En effet, Les réseaux de neurones artificiels (RNA) sont constitués de neurones artificiels communicants, qui peuvent être utilisés pour acquérir une compréhension des réseaux de neurones

Introduction générale

biologiques ou pour résoudre des problèmes d'intelligence artificielle sans avoir besoin de créer un modèle d'un système biologique réel. Similaire à son équivalent dans le système biologique, un réseau de neurones peut apprendre et par conséquent, peut être formé pour trouver des solutions, à reconnaître des régularités, classer les données. Le comportement d'un réseau de neurones est défini par la façon dont les neurones sont connectés et par la force de ces connexions (poids). Les poids sont ajustés automatiquement par le réseau de formation selon une règle d'apprentissage jusqu'à ce qu'il exécute la tâche désirée correctement.

Dans ce travail nous proposons donc, de définir et d'utiliser les réseaux de neurones artificiels. L'objectif est de construire un modèle de classification performant en termes de précision et de taux de classification. Il sera utilisé pour classifier le comportement des vaches laitières en quatre modes de comportements (debout, couché, transition debout-couché, transition couché-debout).

Ce mémoire s'organise en quatre chapitres, à travers lesquels nous décrivons le travail effectué pour la conception et la réalisation du modèle de classification du comportement des vaches laitières :

- ✓ **Chapitre I :** est un état de l'art sur la classification du comportement des vaches laitières. Une première partie a été consacrée à une description de l'intérêt d'utilisation des outils automatiques dans les troupeaux de vaches. Dans une deuxième partie, On s'intéresse à la description de quelques comportements, une présentation des maladies des vaches ainsi que les pertes économiques liées à ces derniers. Nous concluons par une analyse sur les différentes méthodes de classification qui ont été appliquées et les études réalisées dans ce domaine.
- ✓ **Chapitre II :** est dédié à la présentation de la théorie des réseaux de neurones artificiels. Après avoir exposé un aperçu historique sur les RNA, nous détaillons les différents éléments de base de cet outil mathématique. Nous limitons notre présentation à des aspects qui seront utiles pour la suite de ce travail.
- ✓ **Chapitre III :** est consacré à la partie de collection des données. La première partie a été consacrée à la présentation des systèmes embarqués et les domaines de leur utilisation. La deuxième partie est une description de fonctionnement du système de collecte des données ainsi que son déploiement dans une ferme laitière.
- ✓ **Chapitre IV :** est consacré à la modélisation par les réseaux de neurones. Pour se faire, un modèle RNA qui permet de classifier le comportement des vaches laitières en

Introduction générale

les quatre modes correspondant a été développé et testé sur les données collectées. La première partie est dédiée à la présentation du modèle RNA développé, la plateforme utilisée et l'architecture du modèle. La deuxième partie a été consacrée à la présentation des résultats des tests effectués sur l'ensemble des données.

I. Chapitre: Etat de l'art

I.1 Introduction

Les troupeaux de vaches laitières subissent dans le monde entier des pertes en raison des problèmes de santé. La maladie animale mène aux pertes financières qui peuvent transformer une ferme laitière rentable en une entreprise de perte.

C'est pour répondre à ces exigences qu'est né dans les années 2000 l'élevage de précision, qui est défini comme l'utilisation de technologies permettant de mesurer des indicateurs physiologiques, comportementaux ou de production sur les animaux ou les caractéristiques du milieu d'élevage (température, hygrométrie, ventilation...), de Technologies de l'Information et de la Communication pour échanger, stocker, transformer et restituer ces informations à l'éleveur afin de l'aider dans sa prise de décision en complément de ses observations. Plusieurs études ont montré que ces technologies répondent aux contraintes économiques, structurelles et sociales actuelles des exploitations agricoles. L'élevage de précision se développe dans les différentes filières animales pour faciliter la surveillance de troupeaux dans un contexte d'accroissement de la taille des élevages et de diminution de la disponibilité en main-d'œuvre. Du point de vue de l'éleveur, la possibilité de diminuer la pénibilité des tâches est également un élément déterminant pour investir dans l'élevage de précision.

De nombreux dispositifs ont été conçus pour contrôler l'atmosphère des bâtiments, distribuer les aliments, détecter les chaleurs, garder les animaux au pâturage. Les robots de traite par exemple sont devenus de véritables laboratoires, munis de capteurs qui mettent à profit le moment de la traite pour mesurer l'état général de la vache : rythme cardiaque, température corporelle et activité physique.

Les dispositifs combinent l'utilisation de biocapteurs et les technologies numériques de traitement des données. Certains appareils permettent de mesurer l'activité comportementale. Or, le niveau d'activité des animaux est un bon révélateur de leur état général. Les animaux ont un rythme d'activité journalier bien défini, avec un pic d'activité le matin et le soir, une activité modérée pendant la journée et faible la nuit. Ainsi, les animaux repérés individuellement et précocement peuvent être traités plus efficacement.

Ce chapitre donne une description de quelques comportements des vaches laitières, les maladies qui les affectent et des généralités sur les méthodes de classification basées sur l'intelligence artificielle telle que les réseaux de neurones (RNA), les arbres de décision, la logique floue, SVM (*Support Vector Machine*).

I.2 Le besoin d'outils automatiques

I.2.1 Économiser la main d'œuvres

La gestion d'une ferme laitière implique des tâches quotidiennes et des procédures qui exigent une grande habileté et précision. Par exemple, identifier les vaches qui ont besoin d'attention, approcher et traiter ces animaux et rapporter des traitements sont des tâches à forte intensité de main-d'œuvre.

I.2.2 La première détection

Pour que la ferme laitière reste rentable les fermiers doivent assurer la prévention, la détection précoce et le traitement de vaches malades.

I.2.3 L'amélioration du bien-être des animaux

Le bien-être des animaux est une question de grande importance dans les systèmes modernes de production alimentaire en raison de préoccupations des consommateurs à la sécurité et la qualité alimentaire. L'amélioration du bien-être des animaux d'élevage a une influence positive sur la résistance aux maladies et de pathologie animale.

I.2.4 Minimiser les pertes financières et maximiser la production et le profit

Les troupeaux de vaches laitières subissent dans le monde entier des pertes en raison des problèmes de santé qui réduisent la production et exigent des traitements vétérinaires coûteux. Les vaches doivent être en excellente santé pour produire le lait de grande qualité et une performance reproductrice supérieure.

I.3 Le Comportement des vaches laitières

Une vache laitière est une vache élevée pour produire du lait destiné à la consommation humaine. Les vaches laitières produisent des veaux, des génisses pour le renouvellement des troupeaux et de la viande. Il existe à l'heure actuelle dans le monde 225 millions de vaches laitières qui produisent annuellement plus de 500 millions de tonnes de lait. Une vache produit en moyenne 25 litres de lait par jour [ciwf 2018]. La sélection génétique a permis d'obtenir des vaches produisant entre 30 et 60 litres de lait par jour.

La production de vaches, la qualité du lait, la condition corporelle et le comportement fournissent les premières indications de changements concernant la santé. Les vaches

modifient leurs comportements pour leur permettre de faire face aux facteurs de stress tels que l'infection, la satiété ou les changements sociaux et environnementaux. Le comportement est souvent constant et prévisible mais ne peut pas être mesuré à l'échelle en raison de la main-d'œuvre requise pour surveiller physiquement un grand nombre d'animaux en continu.

I.3.1 Quelque Comportement des vaches laitières

I.3.1.1 Position couchée

En position couchée dans leur stalle les vaches adoptent l'une des quatre postures normales suivantes : longue, large, en boule ou étroite.

Lorsque leur stalle est confortable, les vaches y restent couchées 60 % du temps. Elles passent très peu de temps en position perchée ou en position debout.

I.3.1.2 Position debout

En position debout, une vache a les quatre pattes sur la plate-forme de sa logette. Le temps passé en station debout comprend également les tentatives ratées qu'elle fait pour se coucher.

I.3.1.3 Position perchée

En position « perchée », les vaches se tiennent debout avec les pattes avant sur la plate-forme de leur stalle et les pattes arrière dans le couloir. Ce comportement peut également décrire une vache couchée dont une partie du corps se trouve dans la stalle et l'autre dans le couloir. Les maladies de la corne des onglons sont plus fréquentes dans les étables où les conditions favorisent la position « perchée » chez les vaches [Philipot, 1994].

La position « perchée » des vaches couchées constitue un facteur de contamination de la mamelle, des trayons, des pattes et de la queue, et un facteur de risque lié à la mammite. Le temps passé en position « perchée » peut durer de plusieurs minutes à plus d'une heure.

I.3.1.4 Mouvements de lever et de coucher

Normalement, la vache se lève et se couche avec facilité et sans hésitation. Lorsqu'elle se lève, elle propulse d'abord la tête vers l'avant, pour la rétracter ensuite, puis

elle baisse et relève la tête. En fait, elle baisse la tête au point que son menton touche le sol. Par ce mouvement de bascule, elle fait passer son poids vers l'avant pour soulever son arrière-train. Elle commence ensuite à relever légèrement la partie avant de son corps et l'arrière-train suit, propulsé par les pattes arrière. Quand l'animal se soulève, les épaules se déplacent vers l'avant (longitudinalement avec le corps) d'environ 40 cm et la tête d'environ 60 cm.

I.3.1.5 Léchage sous la patte arrière et nettoyage

Le léchage sous la patte arrière désigne le mouvement qu'effectue la vache avec sa langue lorsqu'elle nettoie le pli entre sa mamelle et sa patte. Ce comportement ne devrait pas être ignoré, car c'est un indicateur de l'état du plancher (glissant ou non). Il peut également permettre à l'animal de prévenir l'échauffement entre la patte et la mamelle. Pour faire ce mouvement, l'animal se tient sur ses deux pattes avant et sur une patte arrière.

I.3.1.6 La Marche

La démarche d'une vache saine au pâturage se fait comme suit : elle dépose le pied arrière dans l'empreinte laissée par le pied avant. Sur un sol glissant ou dans l'obscurité, sa démarche est plus hésitante : elle place son pied arrière à l'extérieur de l'empreinte laissée par le pied avant, ce qui modifie son enjambée, la longueur de ses pas et sa vitesse de déplacement. Cette manière de marcher procure à l'animal une plus grande stabilité, mais exerce une plus grande pression sur l'extérieur de l'onglon.

On juge du confort d'une vache à son comportement et aux maladies qui l'affectent. La façon de se comporter d'un animal qui se repose, se nourrit, se déplace ou se tient dans sa stalle fournit d'autres renseignements sur son confort [Anderson, 2001].

I.3.2 L'objectif de la classification du comportement des vaches litières

Dans le secteur d'élevage, l'analyse et la classification du comportement peut fournir des informations sur:

I.3.2.1 La santé animale

Les changements des comportements des animaux peuvent être liés à la santé animale [cow intelligence 2016]- [cowAlert 2017], une détection précoce de la maladie est constatée lorsque par exemple la rumination et l'activité générale diminuent.

I.3.2.2 Événement Chaleur / Œstrus

Le cycle œstral se définit comme l'ensemble des modifications comportementales, anatomiques et physiologiques qui traduisent, chez la femelle, les modifications cycliques de l'ovaire. ([VAILES 1992] ; [NEBEL 2003]).

L'événement de chaleur a été montré pour être détectable par des changements dans l'agitation (activité) [M.S. Shahriar 2016]. La détection des périodes indique le moment approprié pour l'insémination artificielle. Cette période est très importante en élevage et doit être repérée par l'éleveur, devant inséminer l'animal 10 à 12 heures après l'apparition des premiers signes de chaleur.

I.3.3 Maladies des vaches détectables par le comportement

I.3.3.1 La Mammite

La mammite est la maladie la plus répandue chez les vaches laitières. Les vaches ont un rythme d'activité journalier bien défini, avec un pic d'activité le matin et le soir, une activité modérée pendant la journée et faible la nuit. Si l'on observe chez une vache une activité anormale la nuit, cela peut indiquer qu'elle a du mal à se coucher, ce peut être un signe avant-coureur de mammite. Cette maladie réduit la production de lait, sa qualité et nécessite la médication coûteuse et le traitement. La perte économique évaluée de mammite est 80 \$ américains à 250 \$ par lactation pour chaque vache dans un troupeau moyen. [AFIMILK]

Environ 40 % des vaches laitières en France souffrent de mammites. Le coût d'une mammite clinique est estimé à plus de 100 € et l'impact économique total atteint les 230 €/vache/an, soit environ 32 €/1.000 L (figure 1). [BV-2014].

Des recherches faites en Floride sur 2087 vaches ont démontré que les vaches affectées par la mammite clinique durant les 45 premiers jours de gestation avaient près de trois fois plus de risques d'avortement que les vaches non affectées par la mammite durant cette période [NJONG, 2006].

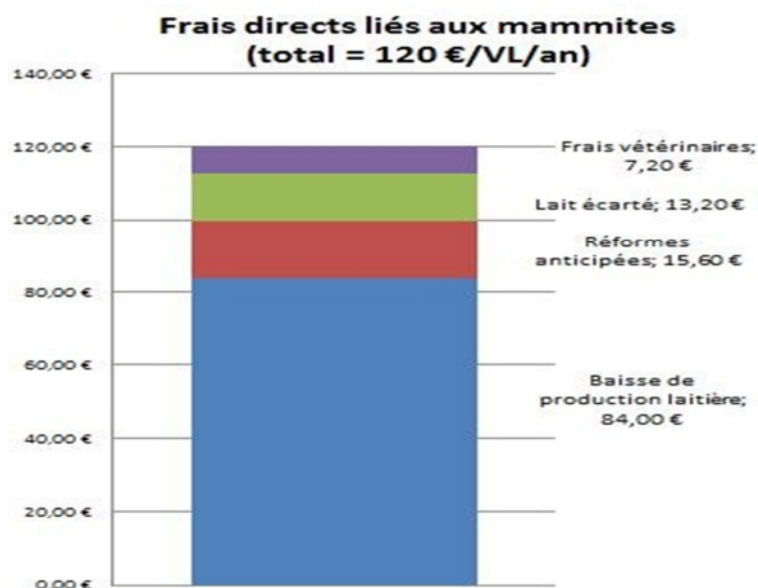


Figure 1 : Frais directs liés aux mammites

I.3.3.2 Boiterie

La boiterie est le deuxième problème de santé le plus répandu dans la ferme laitière moderne, avec une incidence d'entre 20 % et 50 %. Le terme de boiterie correspond littéralement au signe clinique présenté par les animaux qui manifestent une suppression ou un allègement de l'appui sur un membre douloureux générant une marche atypique lors du déplacement. Les vaches boiteuses passeront de plus en plus de temps couchées. Leur production de lait va baisser suite à une diminution de consommation d'aliments et d'eau. On estime la perte de production laitière attribuable à la boiterie à 360 Kg de lait sur 305 jours. [AFIMILK].

Une évaluation de la locomotion se faisait en attribuant à chacune des vaches un score qui varie de 1 jusqu'à 5 et ceci selon la grille d'évaluation de la locomotion montrée dans le tableau suivant [Sprecher 1997].

Tableau 1: Grille d'évaluation de la locomotion.

Score de boiterie	Description	Position du dos à l'arrêt	Position du dos en marche
1	Normal : se tient debout et marche normalement. Les jambes sont bien placées.	plat	plat
2	Légèrement boiteuse : se tient debout avec le dos plat, mais courbe le dos en marchant. La démarche est légèrement anormale.	plat	courbé
3	Modérément boiteuse : se tient debout et marche avec le dos courbé. Enjambées courtes avec une ou plusieurs pattes. Un léger affaissement des ergots du membre opposé au membre affecté peut être observé.	courbé	courbé
4	Boiteuse : se tient debout et marche avec le dos courbé. Favorise une ou plusieurs pattes mais peut encore mettre du poids dessus. Un affaissement des ergots du membre opposé au membre affecté est observé.	courbé	courbé
5	Dos courbé. refuse de mettre du poids sur une seule patte. Peut refuser ou a beaucoup de difficultés à se lever.	courbé	courbé

La figure suivante illustre les pertes économiques qui peuvent être engendrées de la boiterie.

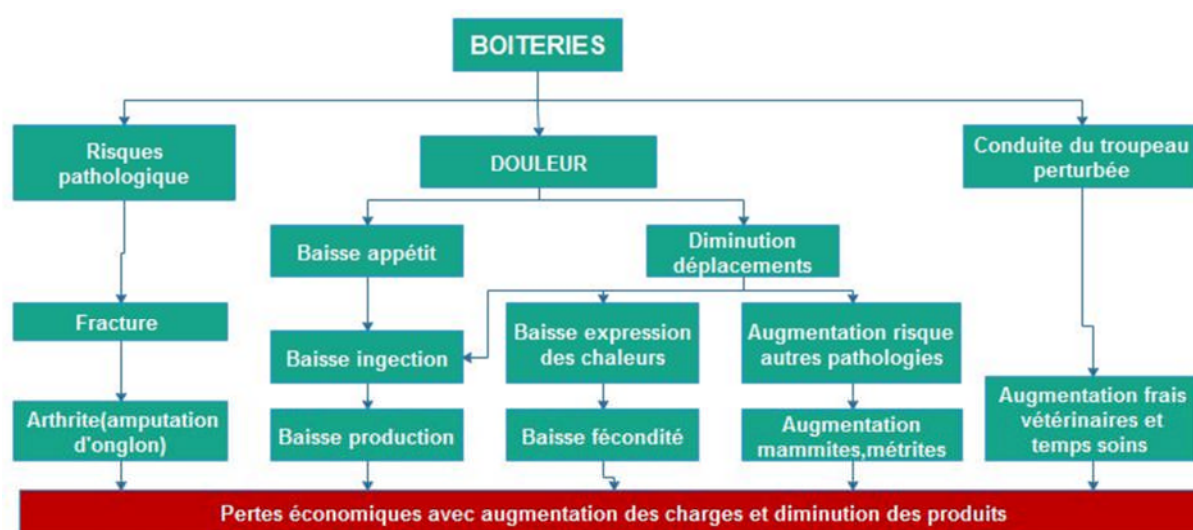


Figure 2: Les pertes économiques liées aux boiteries

I.4 Classification automatique

Le mot « Classification » (en français) désigne à la fois la classification supervisée (classification en Anglais) et la classification non supervisée (Clustering en Anglais) [MNN2005]. Elle est dite automatique lorsqu'elle est faite par une machine, c'est-à-dire qu'elle est réalisée par un programme informatique. La classification est supervisée si les données sont fournies sous forme de couple (x, y) où x est le vecteur d'attributs et chacun de ses composants a pris une valeur permettant de décrire un objet ou une situation, et y est l'étiquette de la classe de l'objet x . Le système, de manière automatique, construit un modèle capable d'associer la classe y au vecteur x pour chaque couple (x, y) . Quant à la classification non supervisée, les valeurs du composant y des vecteurs ne sont pas fournies, le système construit un modèle qui de manière automatique regroupe les objets similaires par catégorie. Dans ce travail, nous nous intéresserons à la classification automatique supervisée.

Nous pouvons aussi définir la classification comme étant le processus de recherche d'un ensemble de modèles (ou fonctions) qui décrivent et distinguent les classes d'objets [DSM2002, HK2001]. Ces modèles construits à partir d'un ensemble d'objets de classe connue seront capables de prédire la classe des autres objets.

I.4.1 Les Etapes d'un processus de classification

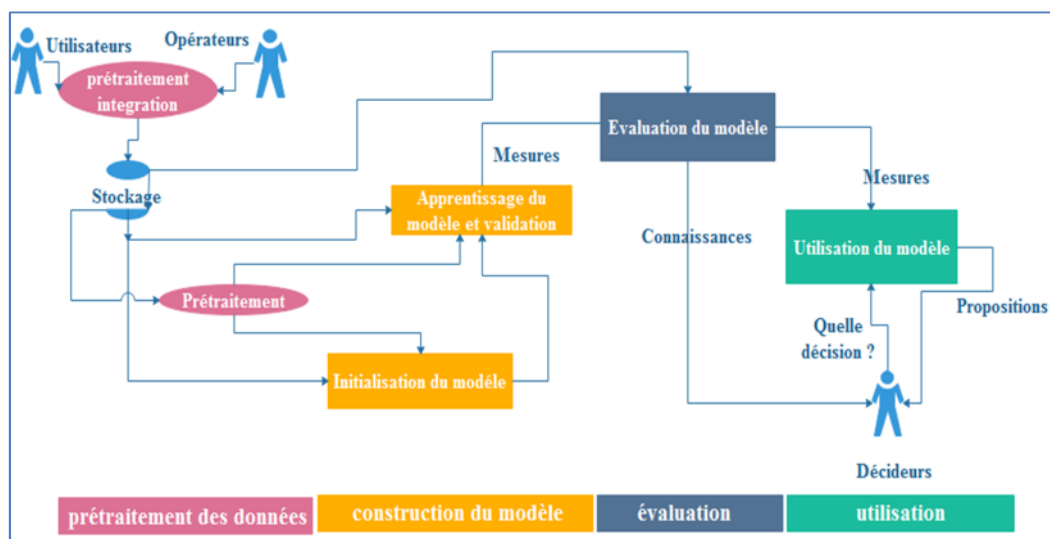


Figure 3: Différentes étapes du processus de classification

Un processus de classification comprend quatre étapes (figure ci-dessus): le prétraitement éventuel, la construction du modèle, l'évaluation du modèle construit et son

utilisation. Le modèle défini peut être représenté sous diverses formes : les réseaux de neurones, les arbres de décisions, les k-plus proches voisins, la logique floue, les séparateurs à vaste marge (SVM)... etc.

I.4.1.1 Le prétraitement

Les données ne se présentent pas toujours comme le souhaitent les utilisateurs. Le problème est surtout lié à : la présence de bruits (erreur introduite dans la mesure d'une variable) dans les données, la présence de valeur inconnue pour certains attributs (données manquantes), l'inconsistance des données, l'incohérence des données et le volume élevé des données. Le but du prétraitement [HK2001, KZ2002] est d'améliorer la qualité des données et par conséquent les résultats des algorithmes qui utilisent ces données, d'améliorer l'efficacité des algorithmes et la facilité de traitement par ces algorithmes. Le prétraitement regroupe [CM2010] :

- **L'intégration des données** : ce processus permet de regrouper les données provenant de plusieurs sources et d'uniformiser le format.
- **Le nettoyage** : il s'agit de supprimer les bruits et de corriger les inconsistances.
- **La transformation des données** : elle consiste à mettre les données sous un type plus approprié aux algorithmes de traitement. Elle intègre les opérations de :
 - **Normalisation** : elle consiste à transformer les valeurs d'un attribut afin que celles-ci puissent appartenir à un domaine.
 - **Binarisation** : elle consiste à transformer les données de type numérique (réel ou entier) ou symbolique (caractères ou énuméré) en données binaires.
 - **Discrétisation** : elle consiste à transformer les données continues d'un attribut en valeurs discrètes.
 - **Projection** : elle consiste à ajouter à chaque exemple un attribut supplémentaire (appelé projection) dont la valeur est calculée comme les sommes des carrés des autres attributs [PYH1995, PYH1997a]. Le but est d'éviter la dépendance du modèle à l'attribut dominant.
- **La réduction des données** : elle consiste à éliminer les redondances dans les données, regroupe ces données et fournit les moyens permettant de sélectionner les attributs pertinents et d'éliminer les attributs non pertinents.

- **Le traitement des valeurs manquantes** : elle consiste à remplacer les valeurs non renseignées dans les exemples.

Dans un processus de classification, après le prétraitement, l'ensemble de données est généralement séparé en trois sous-ensembles : un sous-ensemble d'apprentissage, un ensemble de validation et un sous-ensemble de test. Les deux premiers sous-ensembles sont utilisés pour la construction du modèle alors que le troisième est utilisé pour l'évaluer.

I.4.1.2 Construction du modèle

La construction du modèle de classification est la phase principale d'une tâche de classification. Pour la classification avec les RNA (Réseaux Neurones Artificiels), nous pouvons décomposer cette étape en trois phases successives : l'initialisation du modèle, l'apprentissage du modèle et la validation.

I.4.1.2.1 Phase d'initialisation

L'initialisation du modèle permet de lui donner une architecture initiale. Cette tâche varie suivant les techniques utilisées pour construire les classifieurs.

I.4.1.2.2 Phase d'apprentissage

En science cognitive, l'apprentissage est un processus qui doit faire de telle sorte qu'un individu réalise de manière autonome une tâche donnée [DSM2002]. Elle est intuitive et claire pour les hommes et les animaux. Cette définition correspond à l'apprentissage humain ou animal.

Au niveau machine, on parle d'apprentissage automatique ou artificiel, elle permet de rechercher les paramètres convenables d'un modèle décrivant une réalité à partir des exemples. L'apprentissage automatique est défini [CM2010] comme la recherche de l'hypothèse que vérifient les données avec une erreur minimale. Nous utiliserons l'algorithme d'apprentissage défini de la manière suivante :

L'algorithme d'apprentissage est un algorithme qui permet d'ajuster les paramètres d'une ou plusieurs fonctions pour décrire un phénomène avec une précision désirée ; cet ajustement utilise les observations décrivant certaines situations du phénomène [DSM2002].

Un problème d'apprentissage met en relation trois composants [HK2001, CM2010] :

- **L'environnement** : Il est représenté par les objets x et de leurs classes y . Ces objets sont des exemples à apprendre et sont tirés suivant une distribution de probabilité inconnue. Les exemples (x, y) décrivent un phénomène, ce dernier est une fonction inconnue ψ telle que $\psi(x) = y$, pour tous les couples (x, y) .
- **L'apprenant** : C'est le composant qui doit être capable d'établir à partir des différents exemples (x, y) une fonction ψ^* telle que $\psi^*(x) = y$ qui approxime au mieux ψ . On parle de l'apprentissage humain quand l'apprenant est un homme et d'apprentissage automatique ou artificiel quand il est un programme exécutant un algorithme.
- **Le superviseur**. Il compare pour chaque exemple décrit par x la réponse de l'apprenant à celle attendue y . Il permet alors d'évaluer le niveau de l'apprenant.

I.4.1.2.3 Phase de validation

Dans un problème de classification, évaluer le risque empirique se ramène à calculer une fonction de la différence entre ce que trouve le modèle et ce qu'il aurait dû trouver. Cette phase permet d'évaluer les performances du processus d'apprentissage, de vérifier s'il converge, d'arrêter l'apprentissage et de prévenir les risques de surapprentissage.

On dit que l'apprentissage converge lorsque l'erreur de validation décroît au cours du processus et tend vers une valeur minimale fixée.

I.4.1.3 Phase de test (évaluation)

Pendant cette phase, le modèle (ψ^*) appris est utilisé pour prédire des classes aux objets de l'ensemble de test. Le test permet d'évaluer le système sur les données inconnues du modèle.

Le tableau suivant donne des exemples de comportements classifiés pour un petit sous-ensemble d'espèces animales qui ont été récemment étudiées en utilisant des techniques de classification du comportement automatique.

I.5 Les méthodes de classification

Après le prétraitement, diverses méthodes peuvent être utilisées pour traiter ces données afin de proposer de l'aide aux décideurs. Les méthodes de classification diffèrent suivant la fonction construite pour classer les exemples et la manière de la construire.

I.5.1 Logique floue

La logique floue est une extension de la logique booléenne créée par Lotfi Zadeh en 1965 en se basant sur sa théorie mathématique des ensembles flous, qui est une généralisation de la théorie des ensembles classiques. En introduisant la notion de degré dans la vérification d'une condition, nous permettons à une condition d'être dans un autre état que vrai ou faux. La logique floue confère ainsi une flexibilité très appréciable aux raisonnements qui l'utilisent, ce qui rend possible la prise en compte des imprécisions et des incertitudes. Un des intérêts de la logique floue pour formaliser le raisonnement humain est que les règles sont énoncées en langage naturel.

La logique floue est utilisée dans plusieurs domaines :

- La robotique (reconnaissance de formes).
- La gestion de la circulation routière (feux rouges).
- La médecine (aide au diagnostic).

Dans le domaine de la classification du comportement des animaux plusieurs études ont été faites, allant de la classification des fausses alertes générées par un modèle statistique pour la détection de l'œstrus et la mammite [mol 2001], un problème pour l'application pratique de la détection est la génération d'alertes de fausses-positives. Une alerte est fausse positive si la vache dont l'alerte n'est pas en rut ou ne souffre pas de la mammite.

L'apparition de faux-positifs, les alertes générées par un modèle de détection, crée des problèmes dans la pratique. Le but de cette recherche était de développer et tester un modèle de logique floue pour la classification des alertes de la mammite et œstrus. L'objectif était de garder le même niveau de cas dépistés et de réduire considérablement le nombre des alertes fausse-positive.

I.5.1.1 Avantages de la logique floue

Toute la puissance de la logique floue est de rendre possible la mise en place de systèmes d'inférences dont les décisions sont sans discontinuités, flexibles et non linéaires, plus proches du comportement humain. De plus, les règles de la matrice des décisions sont exprimées en langage naturel.

I.5.1.2 Inconvénients de la logique floue

Le concepteur d'un système flou doit faire un nombre de choix important. Ces choix se basent essentiellement sur les conseils de l'expert ou sur l'analyse statistique des données passées, en particulier pour définir les fonctions d'appartenance et la matrice des décisions.

I.5.2 Les arbres de décisions

Un arbre de décision est un outil d'aide à la décision représentant un ensemble de choix sous la forme graphique d'un arbre. Les différentes décisions possibles sont situées aux extrémités des branches (les « feuilles » de l'arbre), et sont atteints en fonction de décisions prises à chaque étape. Il a l'avantage d'être lisible et rapide à exécuter. Il s'agit de plus d'une représentation calculable automatiquement par des algorithmes d'apprentissage supervisé.

Les arbres de décisions sont aussi utilisés dans divers domaines, tel que:

- Classification automatique d'image.
- La médecine.
- la classification du comportement animal.

La classification du comportement des vaches [Vázquez 2015], dans cette étude, ils ont développé un algorithme d'arbre de décision qui utilise des données d'un accéléromètre à trois axes provenant d'un capteur monté sur le cou pour classifier les comportements important chez les vaches laitières et de détecter les événements de transition entre les positions couchée et debout.

L'algorithme fonctionne en temps réel et étant donné sa structure simple, pourrait être mis en œuvre directement dans un capteur à distance.

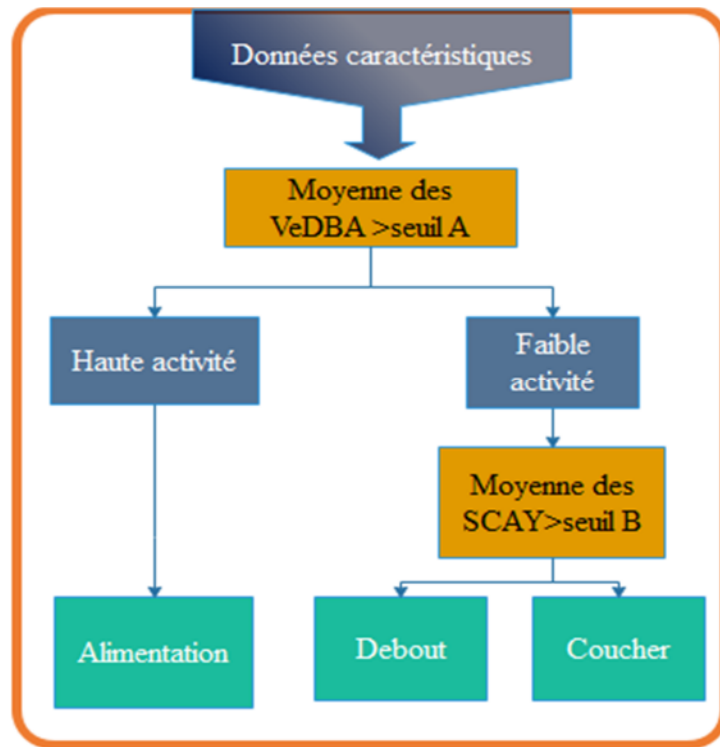


Figure 4: Algorithme de l'arbre de décision utilisé pour la classification des comportements chez les vaches laitières

Les règles de décision sont évaluées vers le bas jusqu'à ce que la classe finale comportementale soit attribuée. Le schéma contient la fonctionnalité caractéristique utilisée comme entrée de données pour la règle de décision à partitionner les données. A chaque règle de décision, les données sont partitionnées en grappes avec des propriétés similaires. La première règle de décision dans cet algorithme fait la distinction entre les activités à dépenses énergétiques élevées et faibles en utilisant l'accélération globale du corps dynamique (VeDBA). Haute énergie les activités de dépenses sont classées comme alimentation. Faible dépense énergétique les activités sont en outre classés en utilisant une deuxième règle de décision qui discriminer les données par la composante statique de l'accélération dans l'axe y (SCAY). Les données avec des valeurs supérieures au seuil B (-0,055 g) sont classées en position debout et les données avec des valeurs inférieures à ce seuil sont classées comme allongé (couché). Avec toutes les tailles de fenêtre, l'alimentation est classée avec une grande précision par l'algorithme de l'arbre de décision. Cependant, il est clair que l'algorithme de l'arbre de décision a tendance à mal classer le comportement debout comme couché (et vice versa). Par conséquent, il semble clair que les comportements les plus susceptibles d'être mal classés sont ceux qui ont le plus de similitudes dans le cou relatif aux mouvements de la vache, notez également que le nombre d'événements debout est significativement inférieur au nombre d'événements coucher ou d'alimentation.

I.5.2.1 Avantage d'un arbre de décision

Un avantage majeur des arbres de décision est qu'ils peuvent être calculés automatiquement à partir de bases de données par des algorithmes d'apprentissage supervisé. Ces algorithmes sélectionnent automatiquement les variables discriminantes à partir de données non-structurées et potentiellement volumineuses. Ils peuvent ainsi permettre d'extraire des règles logiques de cause à effet (des déterminismes) qui n'apparaissent pas initialement dans les données brutes. Il a aussi l'avantage d'être lisible et rapide à exécuter.

I.5.2.2 Inconvénients des arbres de décision

Les arbres de décisions ont aussi quelques désavantages [Bouk 06]:

- Manque de performance s'il y a beaucoup de classes. Les arbres peuvent être très complexes et ne sont pas nécessairement optimaux.
- Sensibilité au manque de données; les arbres peuvent être trop dépendants des données (data overfitting).
- Moins bonnes performances pour les prédictions portant sur des valeurs numériques.
- La construction des arbres de décisions nécessitent généralement beaucoup de temps de calcul: le choix du meilleur partitionnement lors de la construction et la comparaison de sous arbres sont souvent coûteux.
- La combinaison de différents attributs pour les tests n'est pas toujours bien traitée automatiquement; la plupart des algorithmes ne traitent qu'un attribut à la fois.

I.5.3 Machine à vecteurs de support (Support Vector Machine SVM)

Les machines à vecteurs de support ou séparateurs à vaste marge (en anglais support vector machine, SVM) sont un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de discrimination et de régression. Les SVM sont une généralisation des classifieurs linéaires.

Les séparateurs à vaste marge ont été développés dans les années 1990 à partir des considérations théoriques de Vladimir Vapnik sur le développement d'une théorie statistique de l'apprentissage : la théorie de Vapnik-Chervonenkis. Ils ont rapidement été adoptés pour leur capacité à travailler avec des données de grandes dimensions, le faible nombre d'hyper paramètres, leurs garanties théoriques, et leurs bons résultats en pratique.

Les SVM peuvent être utilisés pour résoudre des problèmes de discrimination ; c'est-à-dire décider à quelle classe appartient un échantillon, ou de régression ; c'est-à-dire prédire la valeur numérique d'une variable. La résolution de ces deux problèmes passe par la construction d'une fonction 'h' qui a un vecteur d'entrée 'x' fait correspondre une sortie 'y' : $y = h(x)$ [Wikipedia].

Comme les méthodes précédentes les SVM sont eux aussi utilisés dans plusieurs domaines. À titre d'exemple, dans la :

- Reconnaissance des visages, des textes manuscrits, de la parole.
- L'analyse des bases de données.
- Recherche d'information.

La classification du comportement des vaches [Weizheng 2015]. En tant que méthode d'apprentissage automatique, une machine à vecteurs de support peut être entraînée, ce qui est souvent utilisé pour faire face à des problèmes de classification. Alors que pour les problèmes de classification multi-classes, le support machine à vecteurs est également différent, une structure arborescente classificateur binaire multi-classes a été utilisé pour résoudre le problème. L'idée est de divisée en deux sous-classes de toutes les catégories; chaque classe est également divisée en deux sous-sous-catégories, et ainsi de suite jusqu'à ce que la catégorie finale soit divisée. Les caractéristiques de la méthode de construction du classificateur multi-classe du comportement des vaches laitières sont montrées à la figure suivante :

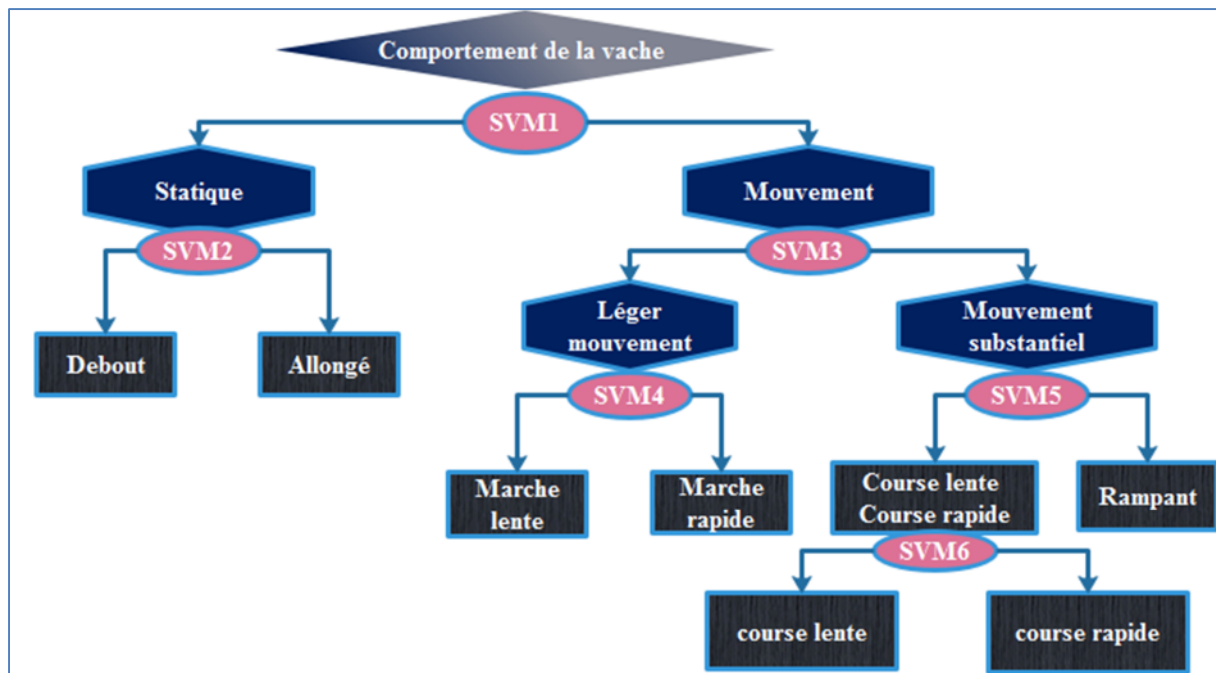


Figure 5: L'arbre binaire de la Classification SVM

I.5.3.1 Avantage des SVM

- Possibilité d'utilisation de structure de données comme les chaînes de caractères et arbres comme des entrées.
- Les capacités de généralisation et la simplicité d'entraînement des SVMs.
- Traitement des données à grandes dimensions : Les SVMs sont très efficaces sur les données à nombre élevé d'attributs, même avec peu d'exemples. Elles n'imposent aucune limite sur le nombre d'attributs sauf les limites imposées par le hardware.

I.5.3.2 Inconvénients des SVM

- Demande des données négatives et positives en même temps.
- Besoin d'une bonne fonction noyau.
- Problèmes de stabilité des calculs.

I.5.4 Les réseaux de neurones

Un réseau neuronal est l'association, en un graphe plus ou moins complexe, d'objets élémentaires, les neurones formels. Les principaux réseaux se distinguent par l'organisation du graphe (en couches, complets. . .) c'est-à-dire leur architecture, son niveau de complexité (le nombre de neurones, présence ou non de boucles de rétroaction dans le réseau), par le type

des neurones (leurs fonctions de transition ou d'activation) et enfin par l'objectif visé : apprentissage supervisé ou non, optimisation, systèmes dynamiques.

Les réseaux de neurones sont utilisés dans plusieurs domaines :

- Pour la classification d'espèces animales par espèce étant donnée une analyse ADN.
- Reconnaissance de motif ; par exemple pour la reconnaissance optique de caractères (OCR), et notamment par les banques pour vérifier le montant des chèques, par La Poste pour trier le courrier en fonction du code postal, etc. ; ou bien encore pour le déplacement automatisé de robots mobiles autonomes.
- Modélisation de l'apprentissage et amélioration des techniques de l'enseignement.

I.5.4.1 Avantage des réseaux de neurones

Les RNA n'exécutent pas d'instructions, et ne manipulent pas de nombres. La destruction d'une partie de ces circuits n'empêche pas le réseau de fonctionner. Ces principales propriétés sont :

- Parallélisme : c'est l'exécution de plusieurs tâches (données numériques et symboliques) en même temps, ce qui explique la grande rapidité de calcul des RNA.
- Capacité d'apprentissage : elle permet aux réseaux de tenir compte des contraintes et des données provenant de l'extérieur. Elle se caractérise dans certains réseaux par leur capacité d'auto-organisation qui assure leur stabilité.
- Généralisation : elle permet au réseau de trouver une solution généralisée applicable à tous les exemples du problème, même s'ils contiennent des erreurs ou sont inachevés ou ne sont pas présentés pendant l'apprentissage.
- Mémoire distribuée : la mémoire est répartie sur plusieurs entités et sur tout le réseau.

I.5.4.2 Inconvénients des réseaux de neurones

- L'absence de méthode systématique permettant de définir la meilleure topologie du réseau et le nombre de neurones à placer dans la (ou les) couche(s) cachée(s).
- Le choix des valeurs initiales des poids du réseau et le réglage du pas d'apprentissage, qui jouent un rôle important dans la vitesse de convergence.

- Le problème du sur apprentissage (apprentissage au détriment de la généralisation).

I.6 Conclusion

Dans ce chapitre nous avons présenté des généralités sur le comportement des vaches laitières, l'objectif de la classification du comportement animal, l'intérêt d'utilisation d'élevage de précision dans les troupeaux des vaches. Nous avons cité quelques maladies détectables à partir des changements du niveau d'activité. Ensuite nous avons présenté quatre méthodes de classification utilisées dans le domaine de la classification du comportement animal et quelques travaux réalisés dans ce domaine. Une étude détaillée sur les réseaux de neurones artificiels (RNA) sera présentée dans le deuxième chapitre.

II. Chapitre : Les réseaux neuronaux artificiels

II.1 Introduction

Les réseaux de neurones sont composés d'éléments simples (ou neurones) fonctionnant en parallèle. Ces éléments ont été fortement inspirés par le système nerveux biologique. Comme dans la nature, le fonctionnement du réseau de neurones est fortement influencé par la connexion des éléments entre eux. On peut entraîner un réseau de neurones pour une tâche spécifique (reconnaissance de caractères par exemple) en ajustant les poids (ou valeurs des connexions) entre les éléments (neurones).

En général, l'apprentissage des réseaux de neurones est effectué de sorte que pour une entrée particulière présentée au réseau corresponde une cible spécifique. L'ajustement des poids se fait par comparaison entre la réponse du réseau (ou sortie) et la cible, jusqu'à ce que la sortie corresponde (au mieux ?) à la cible. On utilise pour ce type d'apprentissage dit superviser un nombre conséquent de pair entrée/sortie.

Étudiés depuis la modélisation du neurone formel en 1943, les réseaux de neurones ont connu un essor considérable, notamment depuis la mise au point du premier algorithme d'apprentissage pour des neurones mis en réseaux en 1985. Ils constituent aujourd'hui un outil classique de l'intelligence artificielle. Ils ont pour avantage leur capacité d'apprentissage qui les rend utiles dans les problèmes pour lesquels il n'existe pas de solution algorithmique acceptable. De plus, leur parallélisme intrinsèque leur confère une grande rapidité de décision.

Le domaine de calcul par réseaux de neurones a mûri au cours de la dernière décennie et a trouvé de nombreuses applications industrielles. Les réseaux de neurones sont maintenant couramment utilisés dans les processus de contrôle, de fabrication, de contrôle de la qualité, de la conception du produit, d'analyse financière, de la détection des fraudes, de l'approbation du prêt, de la reconnaissance vocale et d'exploration de données.

Ce chapitre est une introduction pour le prochaines chapitre pour mieux comprend le fonctionnement des réseaux de neurones. Présente de manière générale les réseaux de neurones, la topologie, les types et le mode d'apprentissage de ces derniers.

II.2 Pourquoi les réseaux de neurone ?

Grâce aux résultats théoriques et pratiques obtenus au cours des dernières années, les réseaux de neurones sont devenus un outil de plus en plus utilisé dans divers domaines (industrie, banque, services). Ils sont utilisés essentiellement pour résoudre des problèmes de

classification, de reconnaissance de formes, d'association, d'extraction de caractéristique, et d'identification,... etc.

Les anciennes méthodes de construction de logicielle ont montré leur limite faces à des problèmes comme la classification. L'approche basée sur les réseaux de neurone permet une certaine souplesse et des imprécisions qui n'affectent pas la fiabilité de l'ensemble et qui garantis une sûreté de fonctionnement ainsi qu'une grande capacité d'adaptation, d'apprentissage et de tolérance aux pannes.

Un intérêt particulier sera donné dans ce chapitre pour la classification du comportement des vaches laitières par réseaux de neurone.

II.3 Du Neurone biologique au neurone artificiel

Le vocabulaire utilisé pour désigner les concepts dans le domaine des réseaux de neurones artificiels est inspiré de la biologie. Nous allons commencer par la présentation de ces concepts de biologie avant de les utiliser de manière métaphorique dans le domaine des RNA.

II.3.1 Neurone biologique

Le neurone biologique est une cellule vivante spécialisée dans le traitement des signaux électriques. Les neurones sont reliés entre eux par des liaisons appelées axones. Ces axones vont eux-mêmes jouer un rôle important dans le comportement logique de l'ensemble. Ils conduisent les signaux électriques de la sortie d'un neurone vers l'entrée (synapse) d'un autre neurone [BENDAOU 2014]. Les neurones font une sommation des signaux reçus en entrée et en fonction du résultat obtenu vont fournir un courant en sortie.

On compte des milliards de neurones dans le corps humain. En général, les neurones ont le même fonctionnement chez tous les animaux : réception, intégration et transmission de l'information. Il assure la transmission des signaux nerveux entre les organes de sens et le cerveau [RADJA2010]. Les différents états d'un neurone sont :

- L'état excité. Dans cet état, il transmet et propage, en fonction des informations qu'il reçoit, des signaux électriques.
- L'état inhibé. Le neurone est inactif et son activité est ralentie.

Les parties du neurone sont montrées dans la figure 6. L'information circule entre les neurones sous forme de signaux appelés influx nerveux. L'influx nerveux se transmet sous forme électrique. Le corps cellulaire reçoit de l'influx en provenance des dendrites et réalise leur intégration. Quand cet influx atteint un certain seuil, le corps cellulaire produit un potentiel d'action qu'il diffuse le long de l'axone grâce à des canaux à ions. A l'extrémité de l'axone, l'influx est transmis aux dendrites des neurones qui lui sont reliés via des synapses. Le poids synaptique définit alors la capacité d'une synapse à influencer le neurone récepteur.

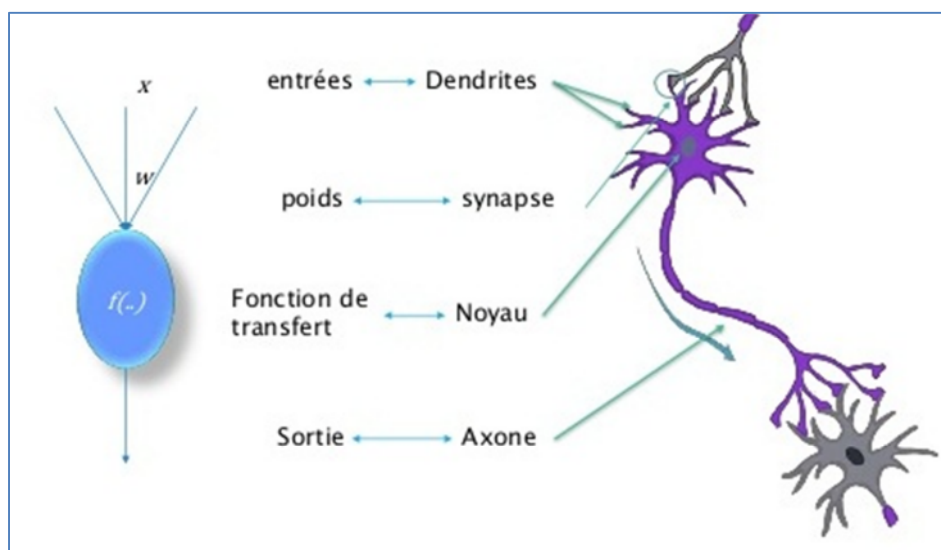


Figure 6: Mise en correspondance neurone biologique / neurone artificiel

L'apprentissage est alors basé sur le mécanisme d'auto-organisation des poids synaptiques et des liaisons neuronales. L'ensemble des neurones constitue le système nerveux. Le cerveau est l'organe central de coordination et de supervision des activités du corps humain.

II.3.2 Neurone formel (artificiel)

Définition : Un neurone formel est une représentation mathématique et informatique du neurone biologique. Un neurone est l'unité élémentaire de traitement d'un réseau de neurones. Il est connecté à des sources d'information en entrée (d'autres neurones par exemple) et renvoie une information en sortie.

Un neurone formel (ou unité neuronale) reçoit en entrée un vecteur d'attributs (entrées) et met en sortie une valeur (sortie). Ces entrées et cette sortie correspondent respectivement aux dendrites et au cône d'émergence (axone et synapses) du neurone biologique. Les entrées peuvent provenir de l'environnement ou des autres neurones.

II.3.2.1 Structure d'un neurone artificiel

Chaque neurone artificiel est un processeur élémentaire. Il reçoit un nombre variable d'entrées en provenance de neurones amont. A chacune de ces entrées est associé un poids w représentatif de la force de la connexion. Chaque processeur élémentaire est doté d'une sortie unique qui se ramifie ensuite pour alimenter un nombre variable de neurones avals.

Un neurone formel basé sur le système visuel peut être présenté comme le schéma de la figure suivante.

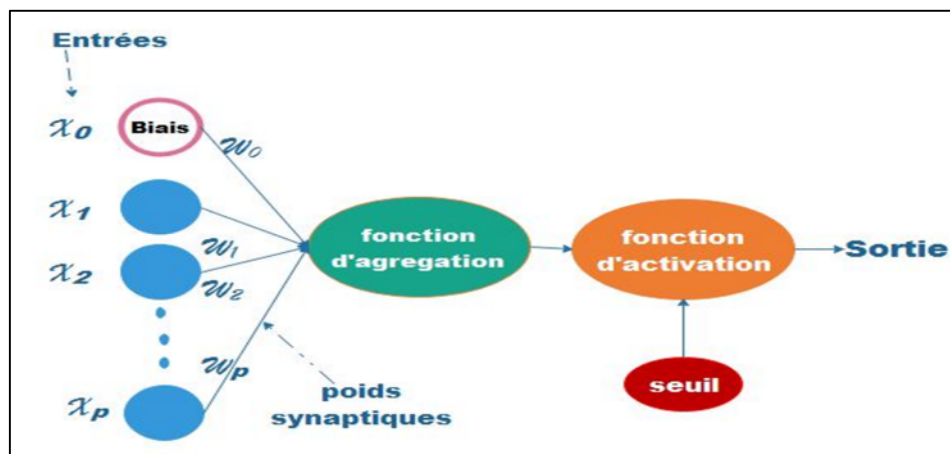


Figure 7: Neurone artificiel

II.3.2.2 Les Entrées

On note $x_i, 1 \leq i < p$ les p informations parvenant au neurone. De plus, chacune sera plus ou moins valorisée vis à vis du neurone par le biais d'un poids.

II.3.2.3 Poids de connexion

Un poids est simplement un coefficient w_i lié à l'information x_i . La i -ème information qui parviendra au neurone sera donc en fait $w_i * x_i$. Il y a toutefois un "poids" supplémentaire, qui va représenter ce que l'on appelle le *coefficient de biais*. Nous le noterons w_0 et le supposons lié à une information $x_0 = + -1$. Un vecteur de poids $[w_i, \dots, w_p]$ dont les valeurs sont estimées dans la phase d'apprentissage.

Le poids de connexion est un nombre réel $w_{i,j}$ associé à la connexion entre le neurone i et le neurone j . Il est aussi appelé poids synaptique. Le poids de connexion permet de mesurer l'influence du neurone j sur le neurone i . Ce nombre matérialise la liaison entre l'axone du neurone i et la synapse du neurone j .

II.3.2.4 Fonction d'agrégation ou de sommation

La fonction d'agrégation d'un neurone est une fonction définie sur les différentes entrées de ce neurone et qui émet comme résultat un réel. Ce réel est appelée Potentiel membranaire du neurone.

La fonction d'agrégation définit comment les entrées du neurone participent à son activité. Elle définit aussi le type du neurone. Elle permet d'intégrer l'information en provenance de différentes entrées. Parmi les fonctions de combinaison nous pouvons citer [[NEDJARI 1998], [YANG 1999]] :

- Le produit scalaire (pour le neurone produit scalaire). L'activité du neurone est une fonction du produit scalaire du vecteur des entrées x et du vecteur des poids de connexion W .

Formellement si A_i est cette activité alors sa valeur est calculée par l'équation 1.

$$A_i = \sum_j w_{i,j} x_j \quad (1)$$

- La distance (pour le neurone distance). L'activité du neurone est calculée comme étant la distance entre le vecteur des poids de connexion et le vecteur des entrées. Formellement cette activité entre les vecteurs w et x peut être calculée par la formule 2 (distance euclidienne).

$$A_i = \sqrt{\sum_j |w_{i,j} - x_j|^2} \quad (2)$$

II.3.2.5 Fonction de transfert

La fonction de transfert d'un neurone est la fonction qui calcule l'état de ce neurone à partir de son potentiel membranaire.

Encore appelé fonction d'activation, elle permet d'introduire la non-linéarité dans le fonctionnement du neurone [[DUCH 1999], [HERTZ 1991]].

Parmi les fonctions d'activation, nous présentons les suivantes :

- La fonction sigmoïde (ou logistique) dont la formule est donnée par la formule 3.

$$\varphi(x) = \frac{1}{1+e^x} \quad (3)$$

- La fonction tangente hyperbolique donnée par la formule 4.

$$\varphi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (4)$$

- La fonction de Heaviside est une fonction calculée par intervalle. Elle est calculée par la formule 5.

$$\varphi(x) = \begin{cases} 1, & \text{si } x < 0 \\ 0, & \text{sinon} \end{cases} \quad (5)$$

Les fonctions sigmoïde et tangente hyperbolique permettent d'introduire la non linéarité entre les neurones et leurs entrées alors la fonction de Heaviside permet de rendre les sorties binaires [DJ99]. Les fonctions sigmoïde et tangente hyperbolique sont aussi différentiables et continues.

II.4 Les réseaux de neurones artificiels

Les réseaux de neurones artificiels sont des réseaux fortement connectés, de processeurs élémentaires fonctionnant en parallèle. Chaque processeur élémentaire calcule une sortie unique sur la base des informations qu'il reçoit. Toute structure hiérarchique de réseaux est évidemment un réseau.

Mathématiquement, un modèle neuronal ou réseau de neurones artificiels est défini comme un graphe orienté, constitué d'un ensemble d'unités (neurones), chaque unité réalisant des calculs élémentaires, structuré (éventuellement) en couches successives capables d'échanger des informations au moyen des connexions qui les relient.

II.4.1 Structure du réseau

Un réseau de neurones est en général composé d'une succession de couches dont chacune prend ses entrées sur les sorties de la précédente. Chaque couche (i) est composée de N_i neurones, prenant leurs entrées sur les N_{i-1} neurones de la couche précédente.

Définition : Une couche est un ensemble de neurones n'ayant pas de connexions entre eux et ayant presque les mêmes neurones connectées à leurs entrées ou connectés en sortie presque

aux mêmes neurones. Les activités des neurones d'une même couche se font généralement de manière simultanée.

On distingue trois sortes de couches :

- La première couche est appelée couche d'entrée composée des neurones qui reçoivent l'information de l'extérieur. Sa taille est donc directement déterminée par le nombre de variables d'entrées.
- La deuxième couche est la couche cachée composée des neurones qui font des traitements intermédiaires. Dans cette couche, les fonctions d'activation sont en général non linéaires. Le choix de sa taille (nombre de neurones) n'est pas automatique et doit être ajusté. Il sera souvent préférable pour obtenir la taille optimale, d'essayer le plus de tailles possibles.
- La troisième couche est appelée couche de sortie composée des neurones qui émettent les résultats vers l'extérieur. Elle donne le résultat obtenu après compilation par le réseau des données entrée dans la première couche. Sa taille est directement déterminée par le nombre de variables dont nous avons besoin en sortie.

Les réseaux de neurones artificiels regroupent en réseaux un certain nombre de neurones formels connectés entre eux de diverses manières. Un réseau est défini par sa topologie, qui représente le type de connexion existant entre les divers neurones du réseau, par la fonction d'activation qui le caractérise et par les méthodes d'apprentissage utilisées pour trouver une relation non linéaire optimale par approximation entre les variables d'entrées et de sorties.

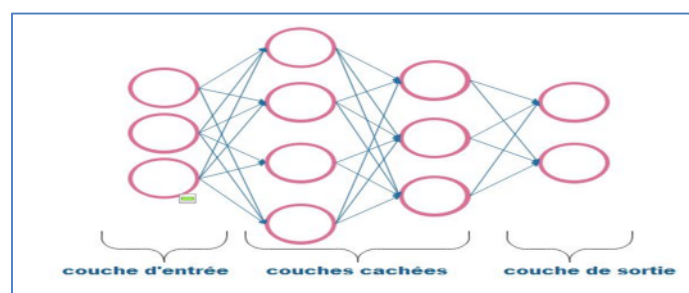


Figure 8: Un réseau de neurone artificiel

Le réseau de neurones peut également contenir des boucles qui en changent radicalement les possibilités mais aussi la complexité. De la même façon que des boucles peuvent transformer une logique combinatoire en logique séquentielle, les boucles dans un

réseau de neurones transforment un simple dispositif de reconnaissance d'entrées en une machine complexe capable de toutes sortes de comportements.

II.5 Topologie des réseaux de neurones

Les connexions entre les neurones qui composent le réseau décrivent la topologie du modèle [Claude 1992]. Elle peut être quelconque, mais le plus souvent il est possible de distinguer une certaine régularité.

II.5.1 Réseau multicouche

Les neurones sont arrangés par couche. Il n'y a pas de connexion entre neurones d'une même couche et les connexions ne se font qu'avec les neurones des couches avales [Claude 1992] .comme le montre la figure 9 chaque neurone d'une couche est connecté à tous les neurones de la couche suivante et à celle-ci seulement.

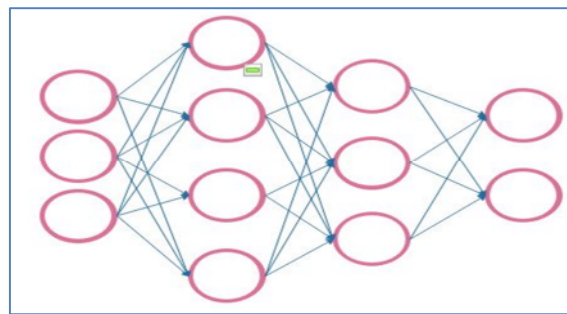


Figure 9: Réseau multicouches

II.5.2 Réseau à connexions locales

Il s'agit d'une structure multicouche. Chaque neurone entretient des relations avec un nombre réduit et localisé de neurones de la couche avale [Claude 1992]. Les connexions sont donc moins nombreuses que dans le cas d'un réseau multicouche classique.

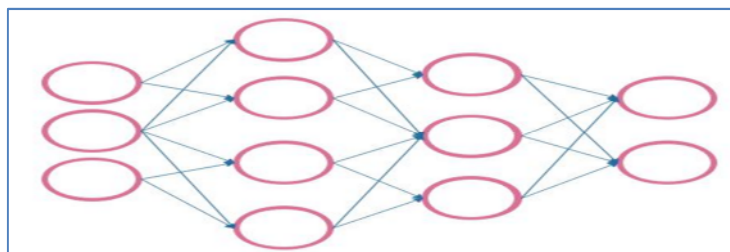


Figure 10: Réseau a connexions locales

II.5.3 Réseau à connexions récurrentes

Les connexions récurrentes ramènent l'information en arrière par rapport au sens de propagation défini dans un réseau multicouche [Claude 1992]. Ces connexions sont le plus souvent locales.

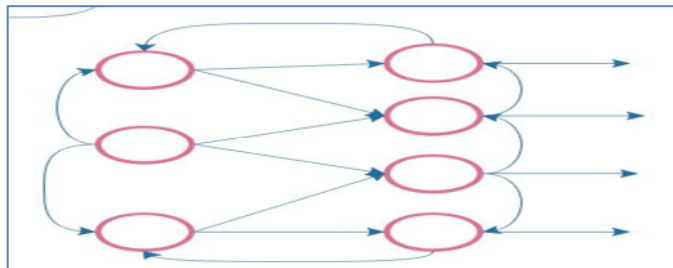


Figure 11: Réseau à connexions récurrentes

II.5.4 Réseau à connexions complètes

C'est la structure d'interconnexion la plus générale. Chaque neurone est connecté à tous les neurones du réseau.

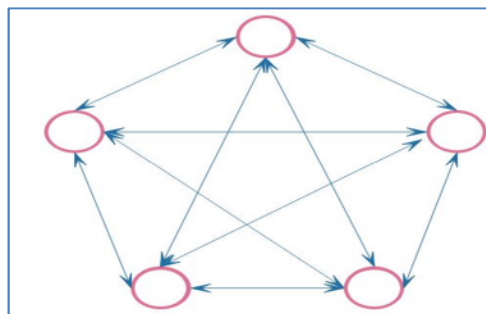


Figure 12: Réseau à connexion complète

II.6 Types de réseaux de neurone

La manière de propager les informations dans un réseau de neurone dépend fortement de la topologie du réseau. Par conséquent, on peut classer les réseaux de neurones en deux grandes catégories, selon la dépendance de l'évolution de ceux-ci en fonction explicite du temps.

- Les réseaux statiques ou non-boucles (FEED FORWARD).
- Les réseaux dynamiques ou récurrents (FEED BACK).

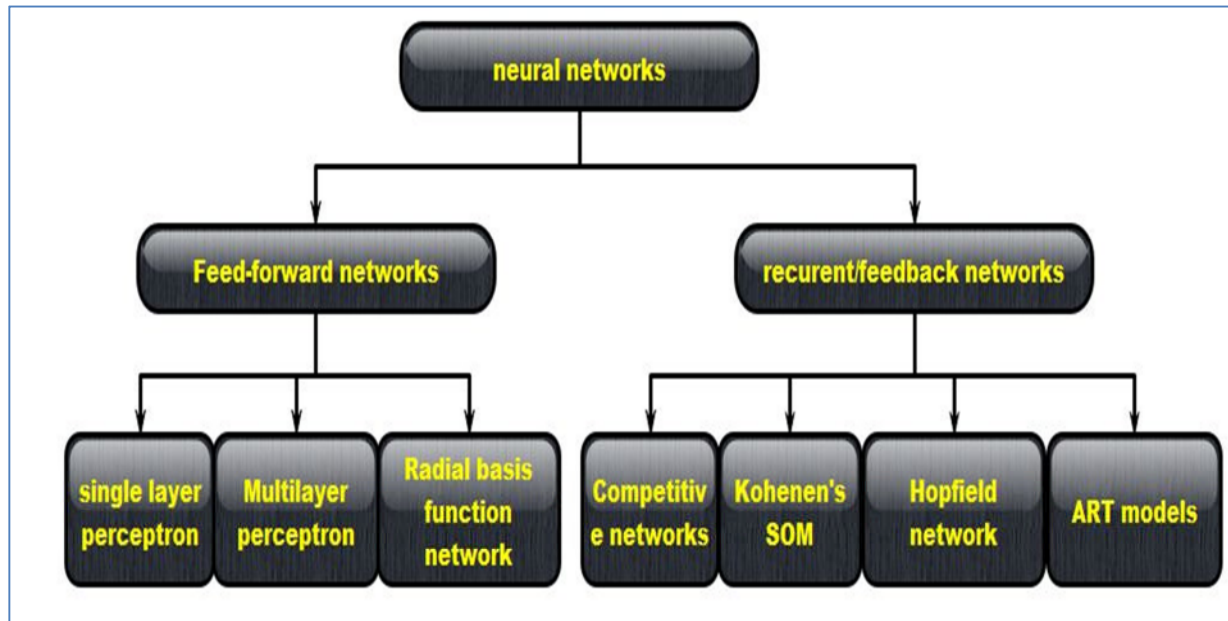


Figure 13: Topologie des réseaux de neurones

II.6.1 Les réseaux non-bouclés (FEED FORWARD)

Un réseau de neurones non bouclé (ou statique) est représenté graphiquement par un ensemble de neurones "connectés" entre eux, l'information circulant des entrées vers les sorties sans "retour en arrière" ; si l'on représente le réseau comme un graphe dont les nœuds sont les neurones et les arêtes les "connexions" entre ceux-ci, le graphe d'un réseau non bouclé est acyclique [DREYFUS]. Les réseaux de neurones non bouclés sont des objets statiques : si les entrées sont indépendantes du temps, les sorties le sont également. Ils sont utilisés principalement pour effectuer des tâches d'approximation de fonction non linéaire, de classification ou de modélisation de processus statiques non linéaires. Dans ce genre de réseau le temps n'est pas un paramètre significatif. En d'autres termes, la modification d'entrée n'entraîne que la modification stable de la sortie, mais elle n'entraîne pas le retour de l'information de cette entrée. Quelques réseaux statiques sont donnés ci-dessous :

II.6.1.1 Réseau de neurones monocouche le perceptron

Le perceptron est un modèle de réseaux de neurones proposé par Rosenblatt. Il est inspiré du modèle formel de neurone apprentissage. Sous sa forme la plus simple, un perceptron est constitué d'une couche contenant N entrées (couche d'entrée) nourrissant une unique unité qui est un neurone. Le perceptron est donc un réseau ayant une unique sortie

(couche de sortie)[Hugues G]. Le neurone de sortie utilise comme fonction d'activation la fonction de Heaviside.

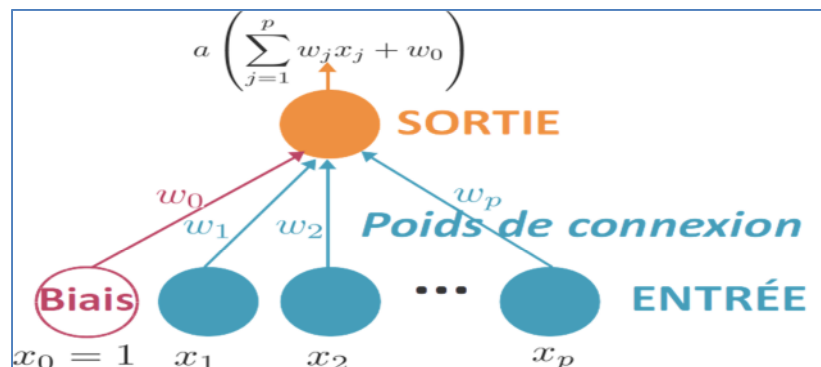


Figure 14: Perceptron monocouche

Dans le perceptron, le vecteur formé par les différentes entrées ne peut être associé qu'à une des deux informations possibles en sortie -1 et 1 (ou 0 et 1). Ce réseau est capable de résoudre des opérations logiques simples comme les opérations "ET logique" ou "OU logique".

Un perceptron est défini par p entrées $(x_1, x_2, x_3; \dots; x_p)$, une sortie O , des coefficients synaptiques $(w_1, w_2; w_3; \dots; w_p)$ et un seuil θ . (La sortie du perceptron dépend du résultat d'une combinaison linéaire effectuée sur les entrées comme suit :

$$v_j = (w_0 + \sum_{j=1}^p w_j x_j) \quad (6)$$

La sortie O est calculée avec l'équation suivante :

$$O = \begin{cases} 1 & \text{si } v_j \geq \theta \\ 0 & \text{si } v_j < \theta \end{cases} \quad (7)$$

II.6.1.2 Le perceptron multicouche (PMC)

Un PMC est un réseau feed-forward dans lequel chaque neurone de la première couche cachée reçoit toutes les données de la couche d'entrée et chaque neurone d'une couche d'un niveau supérieur (différent de la couche d'entrée) est connecté à tous les neurones de cette couche. Dans un PMC, les neurones peuvent utiliser une fonction d'activation de type sigmoïde [Hugues G]. Un PMC associe une entrée à une sortie et est capable de réaliser des tâches de classification, généralisation et de prédiction.

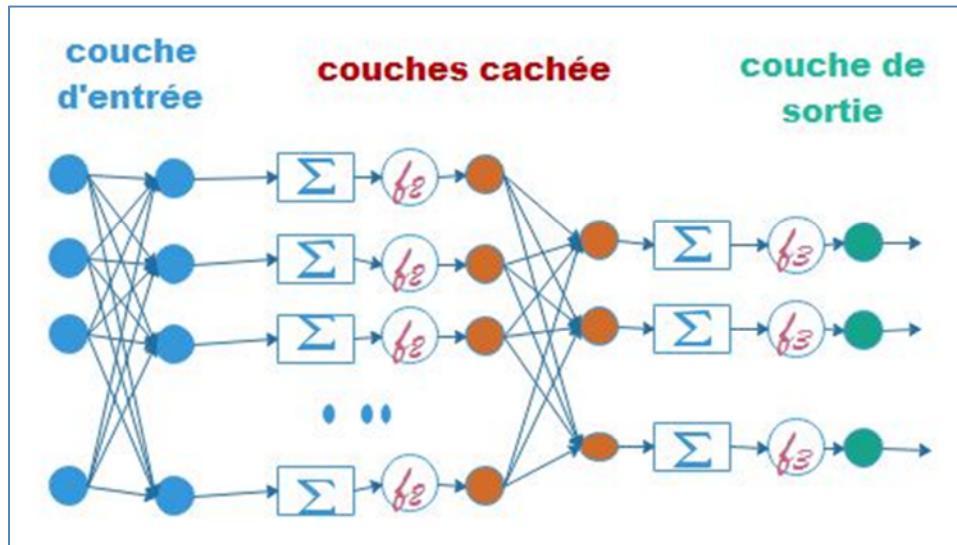


Figure 15: Perceptron multicouche

Ce réseau utilise le modèle d'apprentissage supervisé où l'on fournit au réseau un ensemble d'exemples, chaque exemple est constitué d'un vecteur d'entrée (valeurs numériques) et de son vecteur de sortie désirée. Les poids sont aux débuts aléatoires, et c'est par un mécanisme «correction d'erreur» que ce type de réseau évolue vers un état stable. Cela porte le nom de règle Rétro propagation du Gradient (minimisation d'une fonction dépendante de l'erreur). Le signal d'erreur est propagé dans le sens inverse de la couche de sortie aux nœuds de la couche cachée adjacente, chaque neurone reçoit une part de cette erreur selon sa contribution au vecteur de sortie, c'est-à-dire selon son poids synaptique [BENDAOU 2014]. Ce processus est répété à chaque itération, couche par couche, où les poids synaptiques vont être mis à jour, jusqu'à ce que le réseau converge vers un état stable.

II.6.2 Les réseaux récurrents (FEED BACK)

Contrairement aux réseaux de neurones non bouclés(ou statiques) dont le graphe de connexions est acyclique, les réseaux de neurones bouclés peuvent avoir une topologie de connexions quelconque, comprenant notamment des boucles qui ramènent aux entrées la valeur d'une ou plusieurs sorties. Pour qu'un tel système soit causal, il faut évidemment qu'à toute boucle soit associé un retard : un réseau de neurones bouclé est donc un système dynamique, régi par des équations différentielles [DREYFUS]. Comme l'immense majorité des applications sont réalisées par des programmes d'ordinateurs, on se place dans le cadre des systèmes à temps discret, où les équations différentielles sont remplacées par des équations

aux différences. Les réseaux de neurones bouclés sont utilisés pour effectuer des tâches de modélisation de systèmes dynamiques, de commande de processus, ou de filtrage. Quelques réseaux dynamiques sont donnés ci-dessous :

II.6.2.1 Réseau de Hopfield(HNN)

Ces réseaux sont des réseaux récurrents, un peu plus complexes que les perceptrons multicouches. Chaque cellule est connectée à toutes les autres et les changements de valeurs de cellules s'enchaînent en cascade jusqu'à un état stable [MezhoudN]. Il est formé de N neurones ne pouvant envoyer et recevoir que des valeurs binaires $\{1 \text{ et } 0\}$ ou bien $\{1, -1\}$. La fonction de transfert de chaque neurone est une fonction à seuil. Ce type de réseaux est très bien adapté à la reconnaissance de formes.

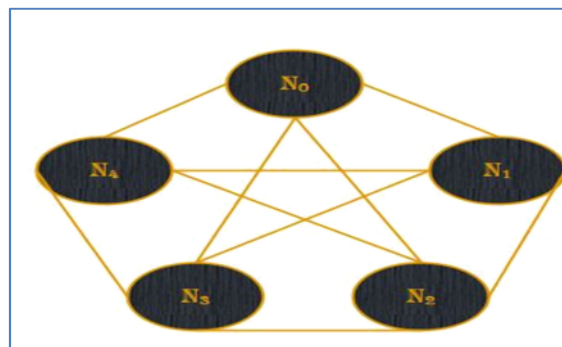


Figure 16: Réseau de Hopfield à 5 neurones

II.6.2.2 Les réseaux de neurones compétitifs

Un réseau à compétition est un réseau monocouche, la couche de sortie étant entièrement connectée, c'est-à-dire qu'une cellule de cette couche est connectée à toutes les autres, un lien excitateur revient sur lui-même, tandis que des liens inhibiteurs la connectent à ses voisins.

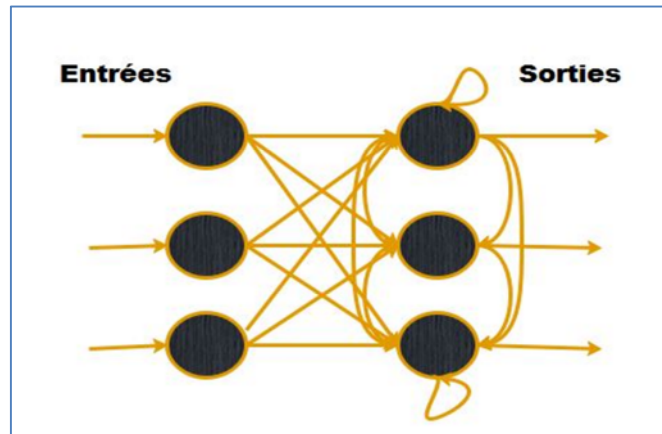


Figure 17: Les réseaux à compétition

Lorsque l'on présente un patron en entrée, les cellules de la couche de sortie reçoivent une activation [Volpi]. À ce moment, les cellules de la couche de sortie entrent en compétition pour déterminer quelle sera la cellule qui sera finalement activée.

L'inhibition est une propriété tirée du neurone biologique qui consiste à transmettre son excitation aux neurones voisins dans un rayon très court et inhibe les neurones situés à plus grande distance.

II.7 Apprentissage des réseaux de neurones

Définir une architecture spécifique d'un réseau est insuffisant pour résoudre un problème donné. Il faut déterminer une configuration de poids adaptée au problème, c'est le rôle du processus d'apprentissage. L'apprentissage consiste en un entraînement du réseau ; on présente au réseau des entrées et on lui demande de modifier sa pondération de telle sorte que l'on retrouve la sortie correspondante. L'algorithme consiste dans un premier temps à propager vers l'avant les entrées jusqu'à obtenir une sortie calculée par le réseau. La seconde étape compare la sortie calculée à la sortie réelle connue. On modifie alors les poids de telle sorte qu'à la prochaine itération, l'erreur commise entre la sortie calculée et connue soit minimisée.

Malgré tout, il ne faut pas oublier que l'on a des couches cachées. On rétro-propage alors l'erreur commise vers l'arrière jusqu'à la couche d'entrée tout en modifiant la pondération [AssaadM]. On répète ce processus sur tous les exemples jusqu'au temps que l'on obtienne une erreur de sortie considérée comme négligeable.

II.7.1 Type d'apprentissage

L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré.

II.7.1.1 Apprentissage supervisé

Dans ce cas, la connaissance à priori de la sortie désirée est nécessaire. On présente au réseau le vecteur d'entrée puis on calcule sa sortie, cette dernière sera comparée avec la sortie désirée, ensuite les poids sont ajustés de façon à réduire l'écart entre elles. Cette procédure est répétée jusqu'à ce qu'un critère de performance soit satisfait.

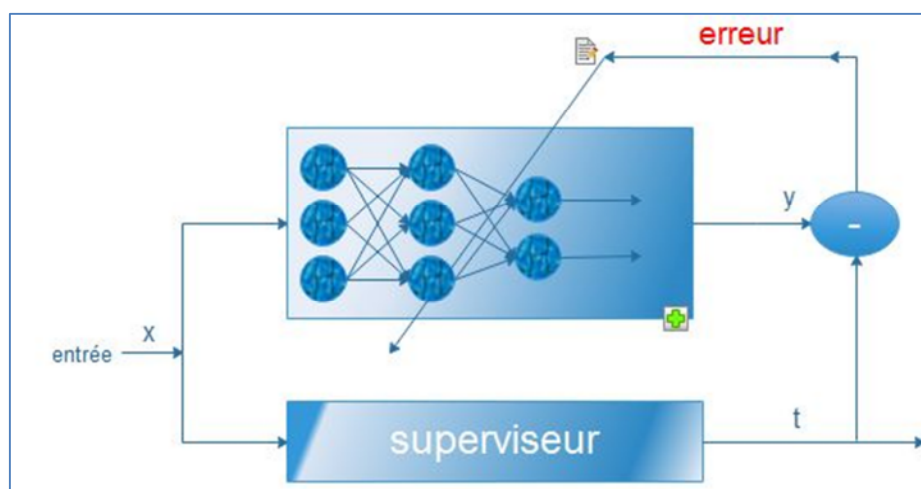


Figure 18: Apprentissage supervisé

II.7.1.2 Apprentissage non supervisé (ou auto-organisationnel)

Avec l'apprentissage non supervisé, le réseau de neurones s'ajuste uniquement en fonction de l'ensemble d'entrées. Il n'est pas nécessaire de connaître à priori les résultats et de constituer un ensemble de sorties désirées.

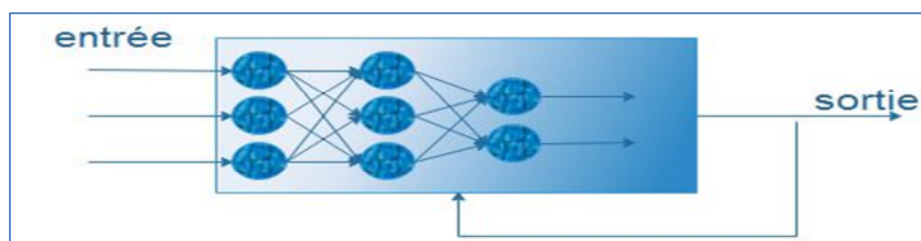


Figure 19: Apprentissage non supervisé

Ce type d'apprentissage utilise une fonction de auto organisation pour déterminer ses sortie en fonction des similarités des entrées du réseau .ce type d'apprentissage est dit compétitif et /ou coopératif.

II.7.1.3 Apprentissage hybride (mixte)

Dans l'apprentissage mixte, l'apprentissage à partir d'explications n'utilise qu'un seul exemple. L'apprenant n'est pas «parfait», il peut avoir besoin d'explications sur d'autres exemples pour bien comprendre un concept [Volpi]. Le mode mixte reprend en fait deux autres approches puisque une partie des poids va être déterminée par apprentissage supervisé et l'autre partie par apprentissage non supervisé.

II.7.2 Règles d'apprentissage

II.7.2.1 Règle de correction d'erreurs

Cette règle s'inscrit généralement dans le paradigme d'apprentissage supervisé, c'est à-dire dans le cas où l'on fournit au réseau une entrée et la sortie correspondante. Si on considère « y » comme étant la sortie calculée par le réseau, et « d »la sortie désirée, le principe de cette règle est d'utiliser l'erreur (« $d - y$ »), afin de modifier les connexions et de diminuer ainsi l'erreur globale du système.

Le réseau va donc s'adapter jusqu'à ce que « y » soit égal à « d ». Notez que l'apprentissage n'intervient seulement que lorsqu'une erreur est faite, dans le cas contraire, les poids demeurent inchangés [Volpi]. Cette règle est donc une forme modifiée de la règle d'apprentissage de Hebb.

II.7.3 Algorithmes d'apprentissage

II.7.3.1 Algorithme de rétro propagation

Minisky et Papert [Franck 2005] ont prouvé que pour pouvoir effectuer une transformation générale, il faut utiliser des neurones cachés. Cette conclusion arrêta les recherches sur les réseaux de neurones pendant plusieurs années, car on ne connaissait pas de méthode pour calculer la valeur de poids à donner à la couche cachée.

Il a fallu des années pour que des chercheurs (RUMELAHT, HINTON et WILLIAMS) introduisent la technique de rétropropagation des erreurs (années 80) « back-propagation » qui se résume comme suit [Éric 1993], [Xuezhi 2001].

- **Algorithme**

Cette méthode se résume donc à ces étapes suivantes :

1. Initialiser les données d'entraînement.
2. Normalisation de ces données d'entraînement.
3. Les permuter aléatoirement.
4. pour chaque donnée d'entraînement n :
 - * Calculer les sorties du réseau observées en propageant les entrées vers l'avant.
 - * Ajuster les poids en rétro propageant l'erreur observée.

$$\mathbf{w}_{ji}(n) = \mathbf{w}_{ji}(n - 1) + \eta \delta_j(n) \mathbf{y}_i(n)$$

Où le gradient local est défini par :

$$\delta_j(n) = \begin{cases} e_j(n) \mathbf{y}_j(n) [1 - \mathbf{y}_j(n)] & \text{si } j \text{ appartient à la couche de sortie} \\ \mathbf{y}_j(n) [1 - \mathbf{y}_j(n)] \sum \delta_k(n) \mathbf{w}_{kj}(n) & \text{si } j \text{ appartient à la couche cachée} \end{cases}$$

Avec : $0 \leq \eta \leq 1$ est le taux d'apprentissage, et $\mathbf{y}_j(n)$ représentent soit la sortie du neurone i sur la couche précédente si celui-ci existe, soit l'entrée i autrement.

5. répéter les étapes 3 et 4 jusqu'à un nombre maximum d'itérations ou jusqu'à ce que la racine de l'erreur quadratique soit inférieure à un seuil.

Cet algorithme est le plus utilisé. Il consiste en fait à suivre la ligne de plus grande pente de la surface d'erreur, qui doit logiquement conduire à un minimum (local ou global).

Sa capacité de convergence peut varier selon les problèmes posés, l'architecture du réseau, les fonctions d'activation retenues, et bien entendu, selon la base d'apprentissage et des variables sélectionnées en entrée du réseau.

II.8 Conclusion

Ce chapitre a permis de rappeler les propriétés principales des réseaux de neurones, ainsi les différents types, topologies et modes d'apprentissage existants des réseaux de neurone. Le grand avantage des réseaux de neurones réside dans leur capacité d'apprentissage automatique, ce qui permet de résoudre des problèmes sans nécessiter l'écriture des règles complexes, tout en étant tolérant aux erreurs.

Cependant, ce sont des véritables boîtes noires qui ne permettent pas d'interpréter les modèles construits. En cas d'erreurs du système, il est quasiment impossible d'en déterminer la cause.

***III. Chapitre: Système embarqué
pour la détection du
comportement des vaches
laitières***

III.1 Introduction

Les systèmes embarqués nous entourent et nous sommes littéralement envahis par eux. On en croise des dizaines par jour sans le savoir. Programmé une machine à café pour avoir un bon petit serré; c'est un système embarqué. Allumé la télévision et utilisé la télécommande ; ce sont des systèmes embarqués. Appelé un ami avec un téléphone portable; c'est un système embarqué. On peut continuer à énumérer tous les systèmes embarqués croisés sans le savoir au cours d'une journée. Ils sont donc partout, discrets, efficaces et dédiés à ce à quoi ils sont destinés. Omniprésents, ils le sont déjà et le seront de plus en plus. On parle en fait d'informatique (et d'électronique). Ils sont bourrés d'électronique plus ou moins complexe et d'informatique plus ou moins évoluée.

Les paramètres comportementaux d'un troupeau d'animaux peuvent être mesurés à l'aide de différents types de capteurs et des stratégies différentes. Ce chapitre est réalisé dans le but d'introduire notre travail, une description générale des systèmes embarqués sera présentée, ainsi le système réalisé pour la collecte des paramètres comportemental des vaches laitières sera présenté. L'ensemble de données collectées est utilisé comme des entrées pour un algorithme de classification (réseau de neurones).

III.2 Système embarqué

Un système embarqué peut être défini comme un système électronique et informatique autonome, qui est dédié à une tâche bien précise. Il contient généralement un ou plusieurs microprocesseurs destinés à exécuter un ensemble de programmes définis lors de la conception et stockés dans des mémoires. Le système matériel et l'application sont intimement liés, le logiciel embarqué étant enfoui, noyé dans le matériel. Le matériel et le logiciel ne sont pas aussi facilement discernables comme dans un environnement de travail classique de type ordinateur PC.

Afin d'optimiser les performances et la fiabilité de ces systèmes, des circuits numériques programmables FPGA (Field Programmable GateArray), des circuits dédiés à des applications spécifiques ASIC (Application Specific Integrated Circuits) ou des modules analogiques sont en plus utilisés.

III.2.1 Architecture d'un système embarqué

L'architecture d'un système embarqué se définit par le schéma suivant[TAYARI 2015]:

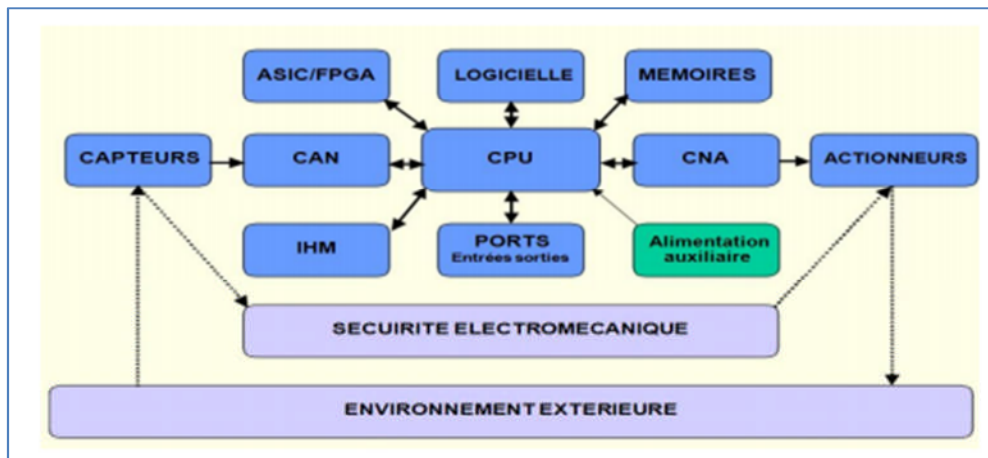


Figure 20: Architecture d'un Système embarqué typique.

Selon la figure ci-dessus on retrouve :

- En entrée des capteurs généralement analogiques couplés à des Convertisseurs Analogiques Numériques (CAN).
- En sortie des actionneurs généralement analogiques couplés à des Convertisseurs Numériques Analogiques (CNA).
- Au milieu, on trouve le calculateur mettant en œuvre un processeur embarqué et ses périphériques d'E/S. Il est à noter qu'il est complété généralement d'un circuit FPGA (Field Programmable GateArray) jouant le rôle de coprocesseur afin de proposer des accélérations matérielles au processeur. Des circuits dédiés à des applications spécifiques ASIC (Application Specific Integrated Circuits).
- Interface IHM (Interface Homme machine) d'un système embarqué peut être aussi simple qu'une diode électroluminescente LED (Light Emitter Diode) qui clignote ou aussi complexe qu'un système de vision de nuit en temps réel ; les afficheurs à cristaux liquides LCD (Liquid Crystal Display) de structure généralement simple sont couramment utilisés.
- Le logiciel a une fonctionnalité fixe à exécuter qui est spécifique à une application. L'utilisateur n'a pas la possibilité de modifier les programmes. Bien souvent, il n'a pas conscience d'utiliser un système à base des microprocesseurs.
- Les mémoires sont intégrés pour le stockage des données et des programmes

Sur ce schéma théorique se greffe un paramètre important: le rôle de l'environnement extérieur, un système embarqué doit faire face à des environnements plus hostiles. Il doit faire face à un ensemble de paramètres agressifs: Variations de la température, Vibrations, chocs, Variations des alimentations, Interférences RF, Corrosion, Eau, feu, radiations...etc.

III.2.1.1 Le fonctionnement du système

- Il reçoit des informations de l'environnement extérieur qu'il converti en signal numérique
- L'unité de traitement composée du CPU, de la mémoire, du logiciel, de l'ASIC et éventuellement de système externes traite l'information.
- Le traitement génère éventuellement une sortie qui est envoyée vers la sortie, les systèmes auxiliaire, les Actionneurs ou l'IHM.

III.2.2 Contrôler à distance

Les systèmes embarqués sont aujourd'hui fortement communicants. Cela est possible grâce aux puissances de calcul offertes par les processeurs pour l'embarqué (32 bits en particulier) et grâce aussi à l'explosion de l'usage de la connectivité Internet ou connectivité IP. La connectivité IP permet fondamentalement de contrôler à distance un système embarqué par Internet.

Ce n'est en fait que l'aboutissement du contrôle à distance d'un système électronique par des liaisons de tout type : liaisons RS.232, RS.485, bus de terrain...

Il faut aussi noter la montée en puissance des communications sans fil (Le Wifi et toutes les normes de réseaux sans fil IEEE 802.15 comme Zigbee) dans l'embarqué au détriment des communications filaires pour limiter le câblage et faciliter la mise en place du système embarqué.

Mais ne nous méprenons pas sur ces facilités et commodités, cela a bien sûr un revers : la sécurité du système embarqué puisque connecté à Internet. La sécurité des systèmes embarqués est donc cruciale aujourd'hui et doit être prise en compte dès leur conception.

III.2.3 Caractéristiques

Les caractéristiques principales d'un système embarqué sont :

- ✓ **Autonomes** : Une fois enfouis dans l'application ils ne sont (le plus souvent) plus accessibles.
- ✓ **Temps réel** : Les temps de réponses de ces systèmes sont aussi importants que l'exactitude des résultats.
- ✓ **Réactifs** : Il doit réagir à l'arrivée d'informations extérieures non prévues.

III.2.4 Domaines d'application

Bien que les systèmes embarqués prennent des formes très différentes et représentent une activité transverse de l'informatique, on peut les ranger dans quelques catégories couramment rencontrées :

III.2.4.1 Systèmes embarqués industriels

Ces systèmes ont essentiellement un rôle de pilotage et de supervision de processus de production industrielle. Le cœur du métier est souvent réalisé par des composants spécialisés (systèmes d'acquisition, FPGA, microcontrôleur, ASIC, etc.) dialoguant par l'intermédiaire de protocoles de communication adaptés (CAN, SPI, PCIe...) avec un processeur intégré sur une carte embarquée qui assure les fonctions secondaires (réseau, interface utilisateur, stockage...).

III.2.4.2 Électronique grand public

Les systèmes embarqués sont omniprésents dans les équipements technologiques actuels (smartphones, tablettes...). Ils gagnent sans cesse du terrain dans de nombreux appareils domestiques (électroménager, domotique, etc.). L'interface utilisateur joue un rôle crucial dans le choix du public, les systèmes d'exploitation et environnements logiciels doivent assurer une cohérence d'utilisation entre les différentes plates-formes. C'est le domaine de prédilection des environnements comme Android, iOS, Tizen, etc.

III.2.4.3 Objets connectés IoT(Internet of Things)

Cette nouvelle branche des systèmes embarqués connaît un essor très important depuis quelques années. Le modèle repose généralement sur la connexion d'objets du quotidien vers des plates-formes de Cloud Computing. L'exploitant stocke ainsi

d'énormes quantités de données en provenance de ses utilisateurs, qu'il peut traiter avec des algorithmes spécifiques et innovants (issu des recherches dans le domaine du Machine Learning) pour en extraire des informations pertinentes pour l'utilisateur (statistiques, prévisions, optimisations, etc.).

Un système embarqué qui a été conçu et mis en place avec le but de surveiller le comportement animal (vaches laitières). Les paramètres de comportement d'une vache peuvent être mesurés à l'aide de différents types de capteurs. L'utilisation de capteurs inertiels, comme l'accéléromètre, du gyroscope et de magnétomètre permet d'obtenir des informations sur l'ensemble des mouvements du corps de la vache.

III.3 Système réalisé pour la collecte des données des vaches

Un dispositif (IMU, carte Arduino.....) est construit dans le but de collecter l'ensemble des mouvements des vaches. Ce dispositif est placé sur le dos des vaches à l'aide de ceinture.

III.3.1 Description du système

Plusieurs composants électroniques ont été utilisés pour construire le dispositif :

III.3.1.1 Carte arduino NANO V 3.0

La carte Arduino Nano est basée sur un microcontrôleur ATmega328 cadencé à 16 MHz. Sa mémoire de 32 kB et son grand nombre d'E/S font de ce circuit compatible DIL30 un élément idéal pour les systèmes embarqués ou pour des applications robotiques nécessitant du multitâches.[gotronic 2018]

La Nano 3.0 peut se programmer avec le logiciel Arduino. Le microcontrôleur ATmega328 contient un bootloader qui permet de modifier le programme sans passer par un programmeur.

III.3.1.2 IMU (unité de mesure inertielle) 9DOF MPU9250

La famille MPU-9250 offre la première solution d'intégration sur 9 axes Motion Processing qui élimine le désalignement au niveau du package gyroscope, Accéléromètre, magnétomètre. Les appareils combinent un gyroscope 3 axes, un accéléromètre 3 axes et un magnétomètre 3 axes sur le même composants avec un processeur de mouvement numérique capable de traiter des algorithmes complexes Motion Fusion 9-axe.

Le circuit intégré MPU9250 contient :

- **Un accéléromètre** (Accélération linéaire sur 3 axes): Comme son nom l'indique, un accéléromètre est un capteur qui permet de mesurer l'accélération d'un corps auquel il est attaché.
- **Un gyroscope** (Vitesse angulaire sur les 3 axes) : Un gyromètre est un capteur inertiel qui mesure la vitesse angulaire du corps rigide dans le repère mobile par rapport au repère fixe.
- **Un magnétomètre** (Mesure du champ magnétique sur les 3 axes): Un magnétomètre est un capteur qui sert à mesurer la direction et l'intensité du champ magnétique, en particulier, le champ magnétique terrestre. Il est largement utilisé dans le cadre d'études géologiques et de détection d'anomalies magnétiques.

Le MPU9250 supporte les communications I2C jusqu'à 400kHz et SPI et a une tension VLOGIC qui définit ses niveaux de tension d'interface; La taille de l'empreinte des appareils MPU est de 3x3x0.9mm (QFN). Les dispositifs additionnels incluent un capteur de températures intégrées et un oscillateur sur puce avec une variation de $\pm 1\%$ sur toute la plage de température de fonctionnement. [IMU 2018]

III.3.1.3 RTC ds1307

Real Time Clock soit en français horloge temps réel, est un circuit intégré dont la principale fonction consiste à mesurer le temps. Ce module, une fois configuré, compte le temps qui passe et ce, de façon autonome et précise grâce à sa pile s'il n'est pas branché, Un module RTC est muni d'une petite pile qui peut durer plusieurs années.

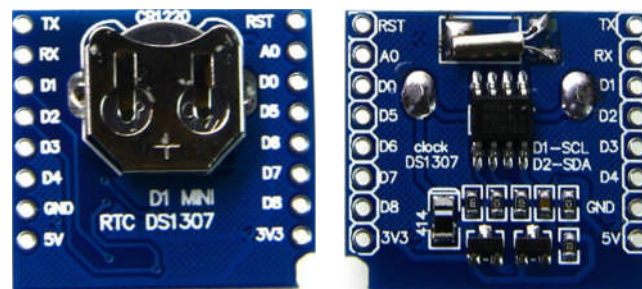


Figure 21: module RTC DS1307

III.3.1.4 Interface I2C (Inter-Integrated Circuit), interface SPI (Serial Peripheral Interface):

Les normes I2C (Inter-Integrated Circuit) et SPI (Serial Peripheral Interface) ont été créées pour fournir un moyen simple de transférer des informations numériques entre des capteurs et des microcontrôleurs.

Les bibliothèques Arduino pour I2C et SPI facilitent l'utilisation de ces deux protocoles. Le choix entre I2C et SPI est en général déterminé par les périphériques que l'on souhaite connecter.

Les deux connexions du bus I2C se nomment SCL (Serial Clock Line) pour l'horloge de synchronisation et SDA (Serial Data line) pour transporter les données.

Le bus SPI utilise quatre fils qui sont :

- **MOSI** (Master out, Slave In) : cette ligne transporte les données de l'Arduino vers le ou les périphériques SPI ;
- **MISO** (Master in, Slave out) : cette ligne transporte les données du ou des périphériques SPI vers l'Arduino ;
- **SS** (Slave select) : cette ligne permet de signifier au périphérique connecté sur le bus que nous souhaitons communiquer avec lui. Chaque périphérique SPI dispose de sa ligne SS connectée à l'Arduino ;
- **SCK** (Serial clock).

III.4 Fonctionnement du système

III.4.1 Le graphe de flux de données

Le graphe de flux de données (Figure 22) est un schéma fonctionnel du système montrant le flux d'informations. Les flèches se pointent de la source vers la destination. Les rectangles représentent les composants matériels et les ovales sont des modules logiciels. Nous utilisons des données graphiques de flux dans la conception de haut niveau, car ils décrivent le fonctionnement global du système, tout en masquant les détails de la façon dont il fonctionne.

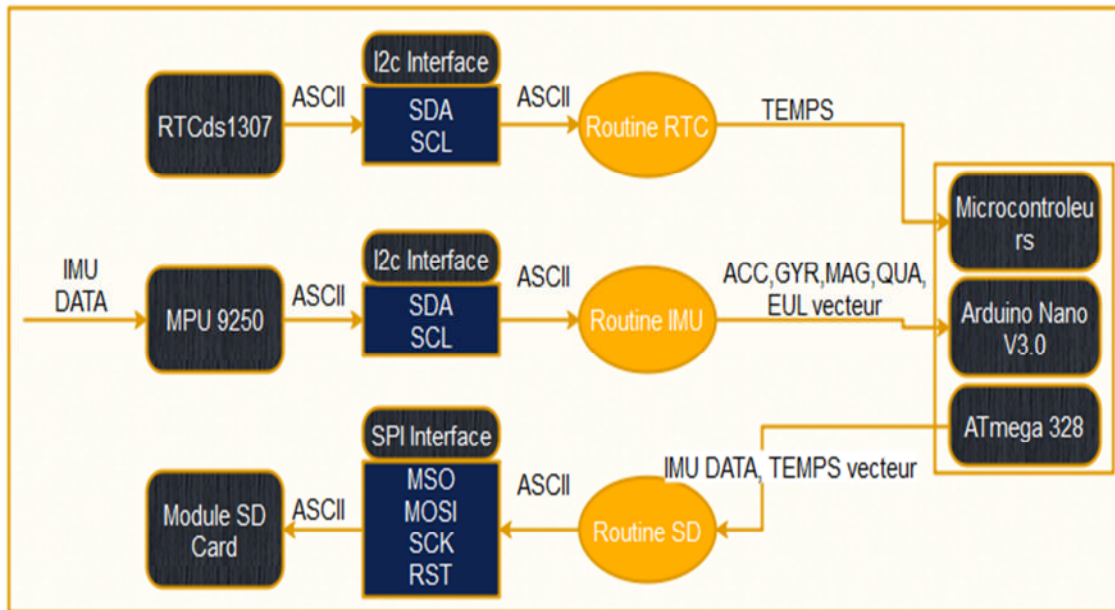


Figure 22: Flux de données pour le système d'acquisition des données

III.4.2 Capture des données

La capture des données fournit au total 16 signaux :

- 3 D accélération : a_x, a_y, a_z ;
- 3 D gyroscope : g_x, g_y, g_z ;
- 3 D Magnétomètre : m_x, m_y, m_z ;
- 4D quaternion : q_0, q_x, q_y, q_z ;
- 3D Euler angle (Yaw, pitch, Roll), suivie de 3 signaux temporelle (heure, minute, seconde).

Les quaternions ont été estimés à l'aide du filtre de Madgwick puis une conversion entre les quaternions et les angles d'Euler a été réalisée et cela en temps réel par le capteur lui-même.

III.4.2.1 Angles d'Euler

A partir de l'intégration de la vitesse angulaire, il est possible de déduire les 'attitudes' d'un objet, autrement dit les angles d'Euler.

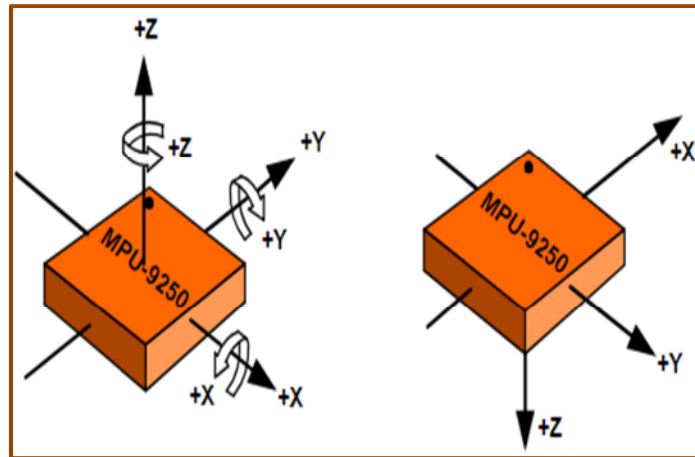


Figure 23: Les angles d'Euler

Il est possible d'estimer la position d'un objet dans un repère fixe à partir des données de la centrale inertielle, en calculant la matrice de rotation.

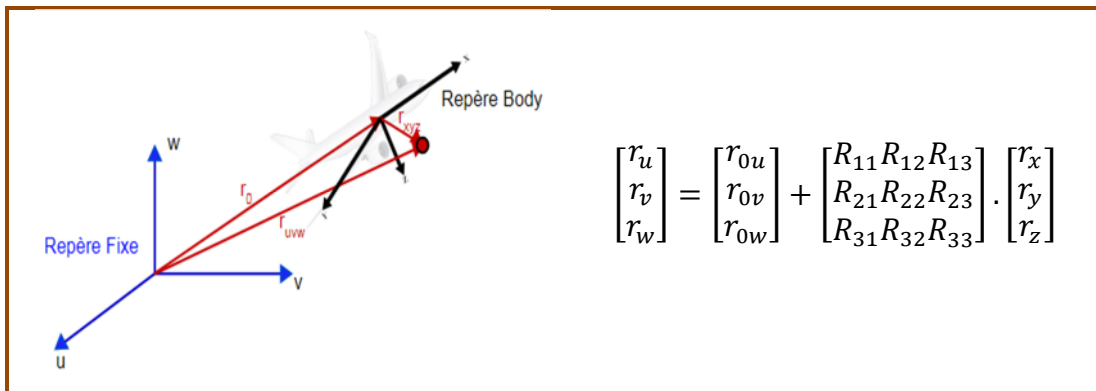


Figure 24: Matrice de rotation

III.4.2.2 Quaternions

Un Quaternion est un objet mathématique (avec un ensemble de propriétés) permettant de représenter le déplacement et la rotation d'un objet.

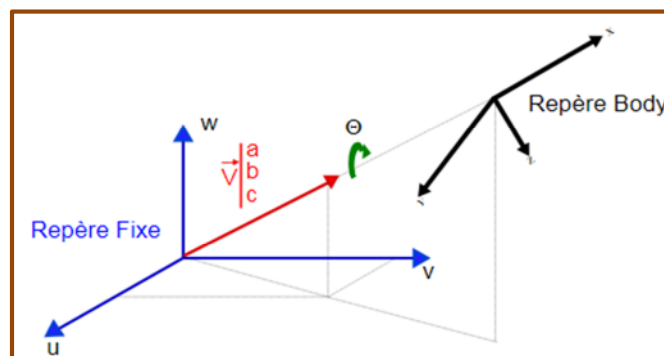


Figure 25: Rotation d'un objet

V direction du repère body, Q est le vecteur quaternion

$$Q = q_0, q_1, q_2, q_3$$

$$Q = \left[\cos\left(\frac{\theta}{2}\right), a \cdot \sin\left(\frac{\theta}{2}\right), b \cdot \sin\left(\frac{\theta}{2}\right), c \cdot \sin\left(\frac{\theta}{2}\right) \right]$$

III.4.2.3 Algorithme de Madgwick

Pour des IMUs bon marché, il est inévitable de réaliser une fusion des données pour que l'accéléromètre compense le défaut du gyroscope. Si l'utilisation d'un filtre de Kalman est possible, il existe d'autres algorithmes (plus rapides) comme celui de Madgwick.[Vincent]

L'idée est de compenser l'erreur de mesure du gyroscope en modulant ses valeurs par le résultat d'une comparaison entre une estimation du champ de gravité et le champ de gravité mesuré (avec l'accéléromètre).[Vincent]. Ce calcul met à jour le quaternion, duquel peuvent être calculées les attitudes (angles d'Euler). La figure 26 montre les étapes du calcul des angles d'Euler.

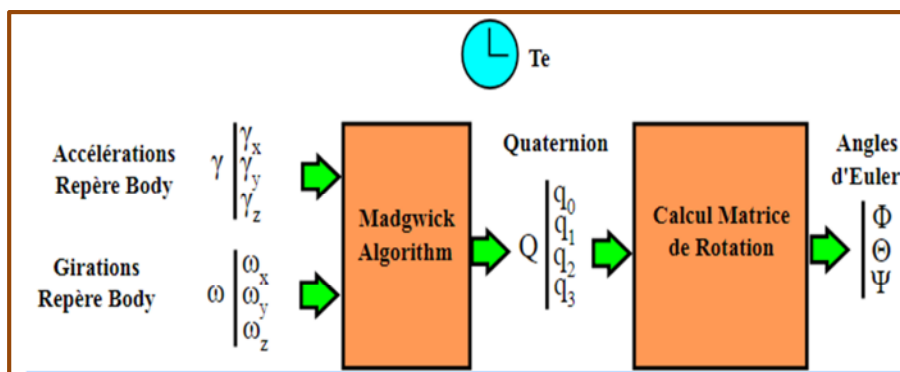


Figure 26: Calcul des Angles d'Euler à Partir du Quaternion.

III.4.3 Graphe des Raffinements successives

Le graphe des Raffinements successives permet de décomposer une tâche en un ensemble de sous-tâches plus simples, ces sous-tâches peuvent être converties en un code logiciel. Raffinement successif, affinement progressif, et la décomposition systématique sont trois termes équivalents à une technique pour convertir un énoncé de problème dans un algorithme logiciel.

Après l'initialisation des capteurs si la LED de débogage est éteinte donc le capteur ne fonction pas correctement sinon le système fonction correctement et lance la collecte des données et les calcule éventuelles.

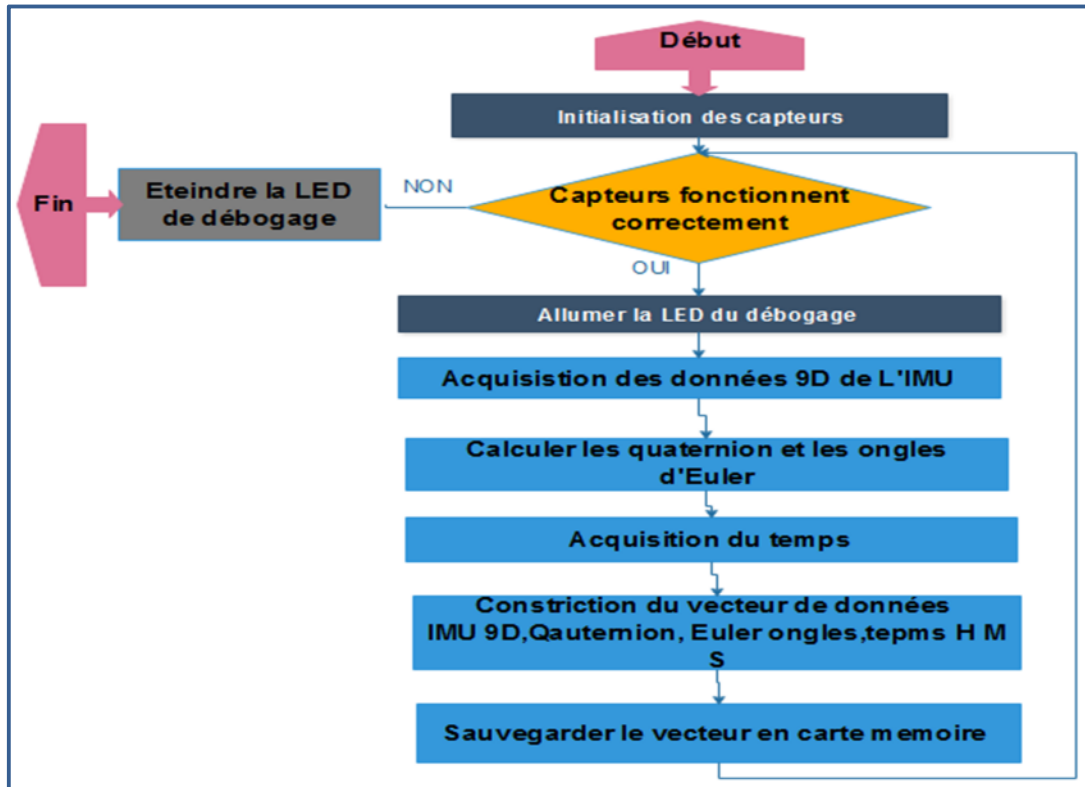


Figure 27: Raffinement successives pour le système d'acquisition de données.

III.5 Déploiement du système

L'expérience a été effectuée dans une ferme de vaches laitières sur 2vaches pour 6 h avec un taux d'échantillonnage de 80Hz. Le dispositif a été placé sur le dos de la vache à l'aide de ceinture pour l'empêcher de glisser vers la droite ou vers la gauche (figure 28).

Le dispositif rassemble l'information de la vache sur laquelle il est placé comme la montre la figure suivante.



Figure 28 : ceinture placée sur le dos d'une vache laitière

Compte tenu de nombre des vaches (2) dans l'expérimentation, la durée d'expérimentation 6h et le taux d'échantillonnage 80Hz, la longueur des données enregistrées dans cette étude est de :

$$2 \times 6 \times 3600 \times 80 = 3456000 \text{ (Échantillons)}$$

A savoir dans chaque échantillon y a 9 données IMU, 3 Données du filtre Madgwick et 3 Données temporelle (Heur minute seconde).

L'objectif principal du dispositif est en utilisant les informations obtenues pour classifié le comportement de la vache (entre 4 différents modes comportementale), utilisant ces données comme des entrée pour un réseau de neurones artificiels qui a été présenté dans le chapitre précédant.

III.6 Conclusion

Dans ce chapitre nous avons présenté les systèmes embarqués leur architecture et les domaines d'utilisation. Ainsi qu'une description du système réalisé pour collecte des paramètres comportementaux des vaches laitières.

Dans le chapitre suivant nous nous focaliserons sur la conception d'un modèle de classification basé sur les réseaux de neurones.

*IV. Chapitre: Classification par
Réseaux de Neurones
Artificiel*

IV.1 Introduction

L'utilisation d'un RNA se fait en deux temps. Tout d'abord nous avons une phase d'apprentissage qui est chargée de trouver la configuration des poids de connexion entre neurones pour qu'il associe à chaque configuration d'entrée une réponse adéquate, puis une phase d'utilisation proprement dite, où l'on présente au réseau des entrées et où il nous indique en retour sa sortie calculée. Construit à partir d'exemples de chaque classe qu'il a appris, un réseau de neurones qui est normalement capable de déterminer à quelle classe appartient un nouvel élément qui lui est soumis.

Dans ce chapitre nous allons développer un modèle RNA pour la classification du comportement des vaches laitières, dont les entrées sont les données des capteurs et les sorties sont les quatre modes du comportement (debout, couché, transition debout-couché et transition couché-debout).

Ce chapitre se compose de deux parties, une pour la construction du modèle RNA basée sur une architecture perceptron multicouches à rétropropagation avec 5 couches (une couche d'entrée, 3 couches cachées, une couche de sortie), ainsi nous avons présenté tous les paramètres du modèle construit. La deuxième partie est consacrée pour la présentation des résultats des tests effectués sur les données collectées en utilisant le modèle construit.

IV.2 Développement d'un RNA pour la classification du comportement des vaches

IV.2.1 Outils de programmation

Différents programmes de simulation par réseaux de neurones sont disponibles et commercialisés (Neurone Solution, Brain Maker, Matlab ToolBoxes, Statistica Neural Network, etc...). Le choix du simulateur se fait sur la base des avantages et des inconvénients de chacun d'eux. Le coût, la flexibilité d'utilisation, le type de plateforme et l'ergonomie de l'interface sont les principaux critères pour évaluer chaque simulateur. Le logiciel (MATLAB) a été développé pour être utilisé dans un environnement efficace et flexible de simulation pour la recherche et l'application des RNA, soit généralement par le développement d'un programme ou l'utilisation de ToolBoxes des réseaux de neurones. Ce dernier est développé pour fonctionner sous Windows avec une interface utilisateur graphique facile à utiliser. Il est aussi employé pour créer, manœuvrer, et visualiser les réseaux de neurones d'une part et importer, exporter les données et les résultats via ces réseaux d'autres part.

IV.2.2 Méthodologie et mise en œuvre

Le perceptron multicouche à rétropropagation (multi-layer perceptron) a été choisi dans cette étude en raison de sa popularité et de sa capacité de réussite en termes de classification, de prédiction et d'optimisation des modèles. Cette architecture est particulièrement bien adaptée à la classification d'ensembles de vecteurs d'entrée en ensembles de vecteurs de sortie correspondants. Dans sa conception standard, l'architecture multi-layer perceptron comporte trois couches de neurones : une couche d'entrée, une couche cachée et une couche de sortie. L'ensemble des données d'entrée est représenté par les données du capteur, cela représente les paramètres de la couche d'entrée qui sont transmis à la couche cachée. Enfin, la couche de sortie reçoit son entrée de la couche cachée. Dans notre cas la sortie de cette couche représente les 4 modes de comportement à classifier. Pour tirer profit au maximum du RNA, il est primordial de définir intelligemment le nombre de neurones et leurs fonctions dans chacune des couches créées.

La première étape consiste à identifier les paramètres à employer pour le modèle de classification, déclarer les matrices des entrées et des sorties sous la forme d'un fichier Excel. La normalisation des données (d'entrée et de sortie) est nécessaire afin qu'elles prennent des valeurs entre -1 et 1 pour être en accord avec la fonction de transfert sigmoïde utilisée.

Les figures suivantes représentent les données collectées par les capteurs (9D IMU ,3D angles d'Euler) après normalisation en utilisant la fonction « mapminmax ».

Ces figures ont été obtenues en utilisant la fonction matlab « plot ».

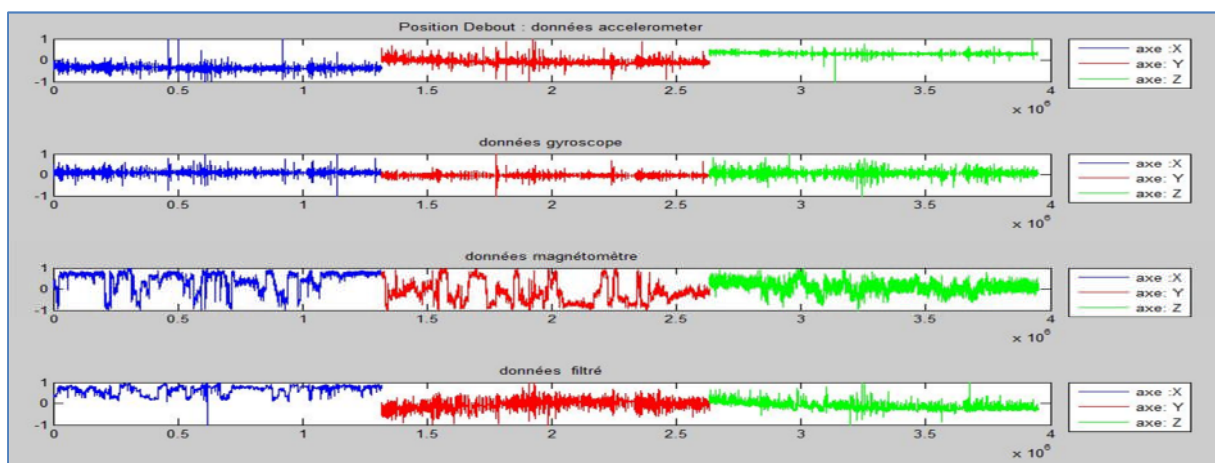


Figure 29: Données d'une vache en position "debout"

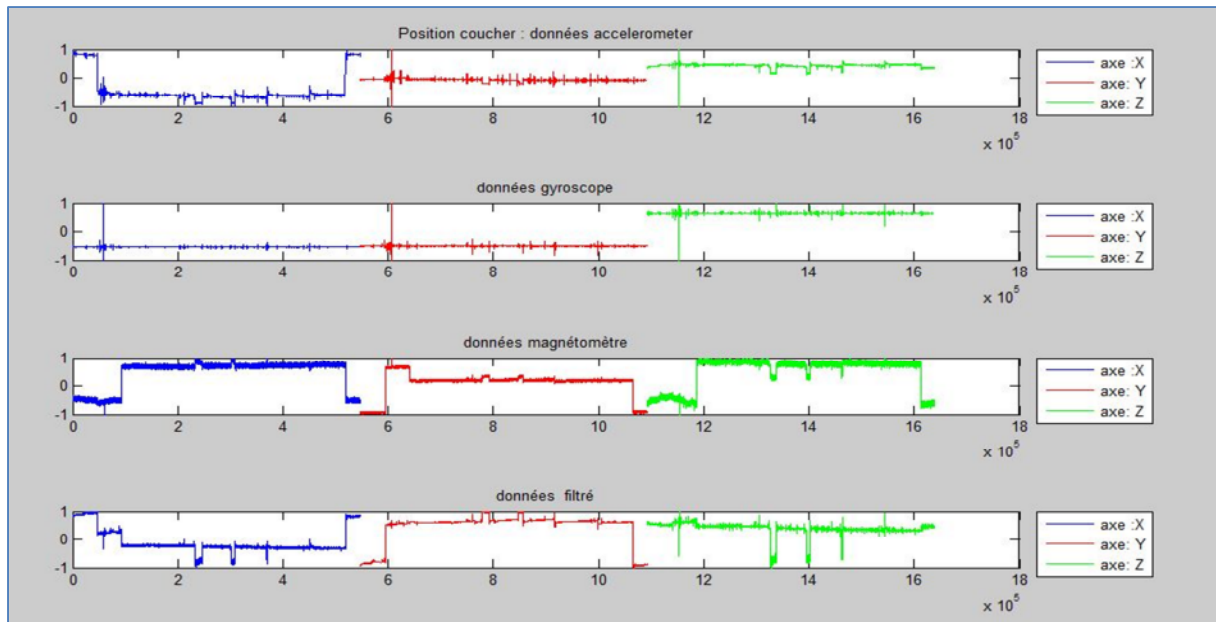


Figure 30:Données d'une vache en position "couché"

À partir des figures ci-dessus nous remarquons que les données montrent une grande différence entre les comportements debout et couché, pour le comportement « couché » les données sont stables par contre les données du comportement « debout » varient entre 1 et -1.

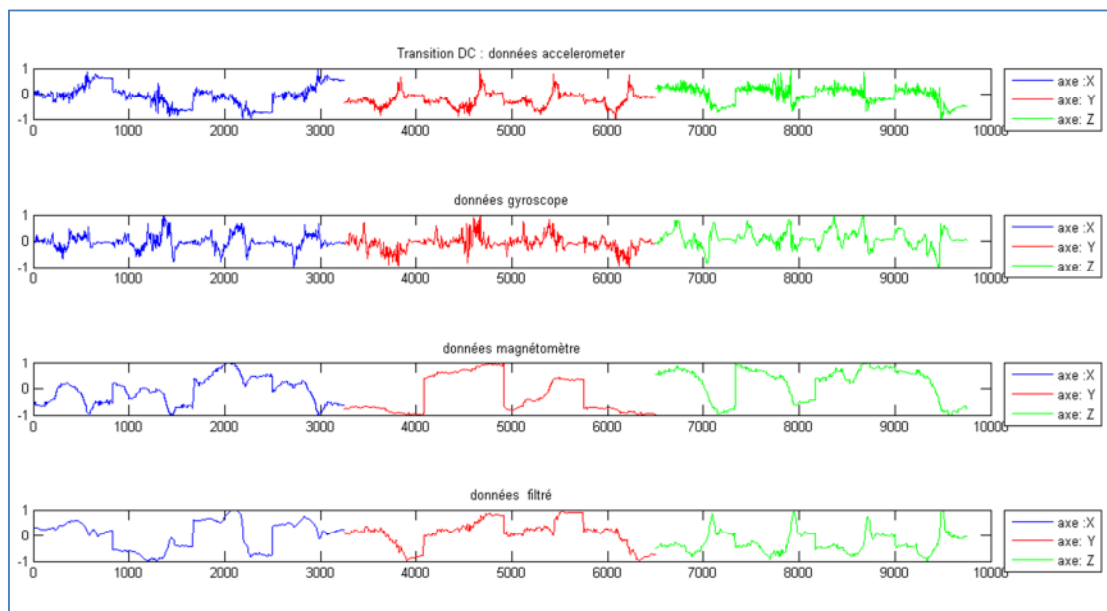


Figure 31: Données d'une vache en transition debout-couché.

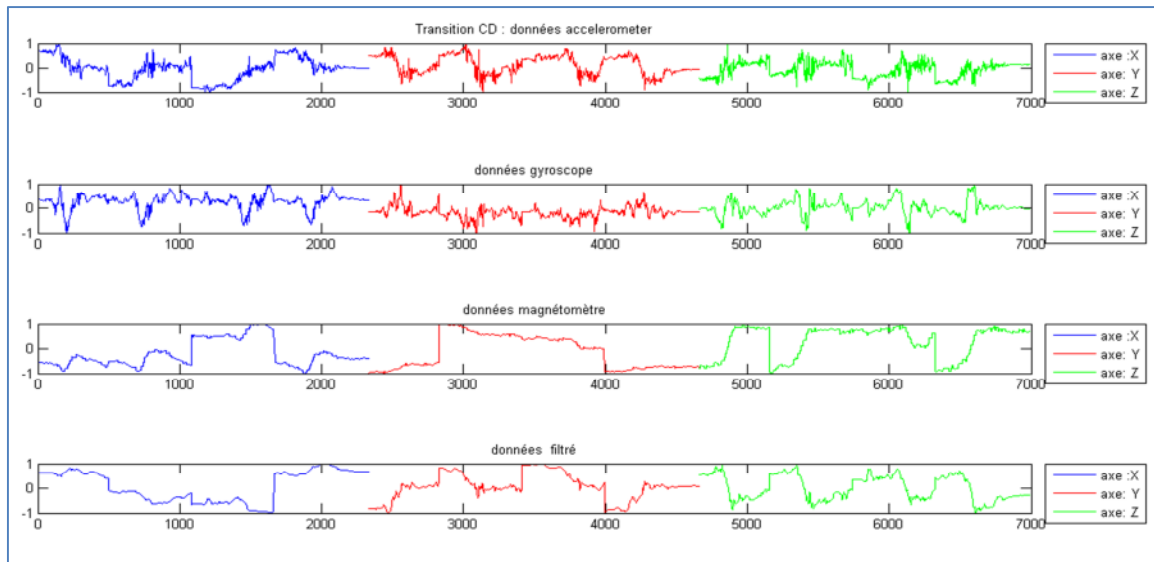


Figure 32: Données d'une vache en transition couché debout

Pour les deux transitions « debout-couché » et « couché-debout » nous remarquons qu'il n'y a pas grande différence.

Les données sont divisées aléatoirement en trois parties. 60% des données sont employées pour l'apprentissage, 20% pour la phase de test et 20% pour la validation.

La mise en œuvre d'un modèle RNA inclut ces trois phases : l'apprentissage, le test et la validation. La phase d'apprentissage ou d'entraînement a pour but de déterminer les paramètres de connexion du réseau à l'aide de la technique d'optimisation. La phase de test consiste à vérifier le réseau déterminé lors de la phase d'apprentissage sur des données non utilisées lors de cette dernière, et à examiner la capacité du réseau à généraliser les exemples d'apprentissage (en comparant la sortie réelle du réseau avec la sortie désirée). La phase de validation est effectuée sur la dernière partie des données.

IV.2.3 Architecture générale du RNA

IV.2.3.1 La couche de sortie

Le RNA devra être en mesure de déterminer les 4 modes du comportement de la vache (couché, debout, transition debout-couché, transition couché-debout). La couche de sortie devra donc avoir les spécificités suivantes :

- un neurone pour l'identification du comportement «couché ».
- un neurone pour l'identification du comportement «debout ».

- un neurone pour l'identification du comportement «transition debout-couché».
- un neurone pour l'identification du comportement «transition couché-debout ».

IV.2.3.2 La couche d'entrée

La couche d'entrée doit comporter les neurones recevant les données collectées par les capteurs. Ces entrées seront transmises au réseau et divisés en trois ensembles de données (60 % pour l'entraînement 20% pour le test et 20% pour l'évaluation). Le nombre de neurones dans la couche d'entrées varie selon les tests effectués sur les données :

- 3 neurones en utilisant seulement les données de l'accéléromètre (MA).
- 3 neurones en utilisant les données du filtre Madgwick(les angles d'Euler) (MF).
- 6 neurones en utilisant les données du l'accéléromètre et du gyroscope (MAG).
- 6 neurones en utilisant les données du l'accéléromètre et du magnétomètre (MAM)
- 9 neurones en utilisant les données du l'accéléromètre, du gyroscope et du magnétomètre (MAMG).

IV.2.3.3 La couche cachée

La topologie multi-layer perceptron comporte encore une ou plusieurs couches cachées. Quelques tests avec des couches cachées modifiées vont être effectués afin d'observer une éventuelle incidence sur les résultats obtenus.

La Figure 33 permet de mieux saisir l'architecture développée, présente le RNA dans sa totalité avec 9 entrées, 3 couches cachées comportant [20 10 10] neurones et une couche de sortie avec 4 neurones correspondant au 4 mode de comportement (couché, debout, transition debout-couché, transition couché-debout).

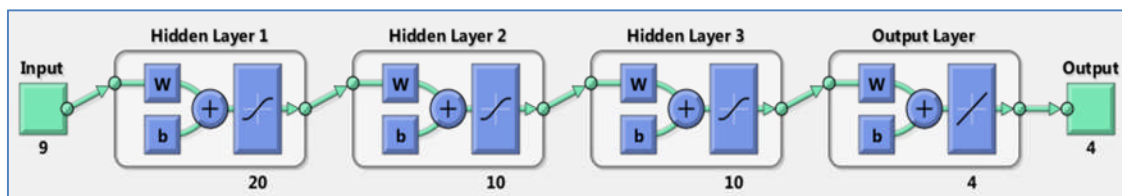


Figure 33: Architecture du RNA développée avec 3 couches cachées de [20 10 10] neurones

IV.2.4 Fonctionnement

Le fonctionnement des réseaux de neurones à rétropropagation peut être divisé en deux étapes : introduction des entrées et rétropropagation de l'erreur. Durant ce processus, les poids synaptiques sont constants. A la fin de la première itération, le vecteur de sortie est comparé au vecteur voulu, et une valeur d'erreur est calculée pour chaque nœud de sortie. Ce signal d'erreur est propagé dans le sens inverse de la couche de sortie aux nœuds de la couche cachée adjacente, chaque neurone reçoit une part de cette erreur selon sa contribution au vecteur de sortie, c'est-à-dire selon son poids synaptique. Ce processus est répété à chaque itération, couche par couche, où les poids synaptiques vont être mis à jour, jusqu'à ce que le réseau converge vers un état stable.

IV.2.5 Initialisation du RNA

Avant l'entraînement du RNA, chaque pondération doit être initialisée à une valeur. Il existe plusieurs algorithmes d'initialisation d'un RNA, le plus courant (et le plus intuitif) reste encore l'initialisation aléatoire des poids sur les connexions en utilisant l'algorithme Nguyen-Widrow. A partir de cette initialisation aléatoire, le RNA va modifier les pondérations par apprentissage pour diminuer l'erreur entre la sortie désirée et la sortie réelle.

IV.2.6 Apprentissage

L'apprentissage est supervisé, c'est-à-dire qu'on associe une configuration d'entrée à une configuration de sortie. L'algorithme de rétropropagation est utilisé pour l'apprentissage du réseau. Cet algorithme est un gradient itératif conçu pour minimiser l'erreur quadratique moyenne (MSE) entre la sortie obtenue et la sortie désirée. Cette minimisation est réalisée par une configuration des poids adéquate. L'erreur (e) est la différence entre la valeur désirée (t) pour le neurone de sortie et sa valeur calculée par propagation (y).

L'apprentissage est divisé en cycles (epoch en anglais). A chaque cycle d'apprentissage tous les patterns de l'ensemble d'entraînement sont présentés une fois à l'algorithme de correction d'erreur. Les pondérations du réseau de neurones artificiels sont alors modifiées afin d'améliorer la classification vers le résultat désiré. Le nombre de cycles d'apprentissage peut être défini par l'utilisateur. Plus les cycles seront nombreux, plus l'erreur sur l'ensemble d'entraînement deviendra petite. L'erreur diminuera aussi sur l'ensemble de test jusqu'à un certain nombre de cycles. Passé ce nombre, l'erreur sur l'ensemble de test commencera à augmenter à nouveau, ceci est dû au fait que plus l'apprentissage est long, plus le réseau

apprend les spécificités propres à l'ensemble d'entraînement, et ainsi perd son pouvoir de généralisation (objectif pourtant recherché d'un RNA).

IV.2.6.1 Paramétrage de la phase d'apprentissage

IV.2.6.2 Performances du réseau de neurones multicouche à rétropropagation

La performance d'un modèle de réseau de neurones peut être évaluée en utilisant différentes expressions, notamment :

- L'erreur quadratique moyenne (MSE : Mean Square Error).

$$MSE = \frac{1}{n} \sum_{i=1}^n (t_i - o_i)^2 \quad (1)$$

Avec : t_i : La sortie désirée du modèle i ;

o_i : La sortie réelle du modèle i et n : le nombre total de données.

D'autres fonctions peuvent être utilisées pour évaluer le taux de réussite de la classification :

- Sensibilité $= \frac{TP}{TP+FN} \times 100$ (2)

- Précision $= \frac{TP}{TP+FP} \times 100$ (3)

- Accuracy (taux de classification) $= \frac{TP}{TP+FP+TN+FN} \times 100$ (4)

Avec:

- ✓ TP (true positive) : est le nombre d'exemples dont le comportement est bien classifié.
- ✓ FN (false negative) : est le nombre d'exemples dont le comportement est observé mais il est classifié comme un autre comportement.
- ✓ FP (false positive) : est le nombre d'exemples dont le comportement n'est pas classifié correctement mais n'est pas observé en réalité.
- ✓ TN (truenegative) : est le nombre d'exemples dont le comportement est classifié correctement mais n'est pas observé.

Ces valeurs seront calculées à partir d'une matrice de confusion.

Définition : La matrice de confusion, dans la terminologie de l'apprentissage supervisé, est un outil servant à mesurer la qualité d'un système de classification. Chaque colonne de la matrice représente le nombre d'occurrences d'une classe estimée, tandis que chaque ligne représente le nombre d'occurrences d'une classe réelle (ou de référence) [wik 2018].

1322 38.3%	0 0.0%	0 0.0%	0 0.0%
0 0.0%	1376 39.9%	2 0.1%	0 0.0%
0 0.0%	1 0.0%	464 13.4%	1 0.0%
2 0.1%	0 0.0%	2 0.1%	280 8.1%

Figure 34: Matrice de confusion dont les lignes représentent les classes observées,

Les colonnes représentent les classes prédites

TP (class i) = matrice(i,i).

FP (class i) = some (matrice (i,:))- matrice(i,i).

FN (class i) = some (matrice (:,i))- matrice (i,i).

TN (class i) = some(matrice(:)) – some(matrice (i,:))- some(matrice(:,i)) + matrice(i,i).

IV.2.6.2.1 Algorithme d'apprentissage

Dans un second temps, nous procédons à la détermination du type d'algorithme d'apprentissage à utiliser dans notre étude. Après plusieurs tentatives, l'algorithme de Levenberg-Marquardt (trainlm) s'avère le plus efficace et le plus fiable. Le tableau ci-dessus montre une comparaison entre différents algorithmes d'apprentissage en termes dunombre d'itérations maximale, du temps nécessaire pour la convergence et du degré de performance atteint. Dans ce cas nous avons fixé 1000 itérations comme étant la limite du processus de convergence.

Tableau 2: Comparaison entre les algorithmes d'apprentissage

Algorithme d'apprentissage	Meilleure performance a l'epoch	Temps d'apprentissage(min)	Performance(mse)
Trainlm	40	02:22	0.0006
Traingdm	1000	00:32	0.0799
Traingd	1000	00:56	0.0830
Trainbfg	199	01:31	0.0043
Traingda	109	00:03	0.0800

- Trainlm (Levenberg-Marquardt): est une fonction de formation de réseau qui met à jour les poids et les valeurs biais après optimisation de Levenberg-Marquardt.[Trainlm]
- Traingdm(Gradient descentwithmomentum) : est une fonction de formation de réseau qui met à jour les valeurs biais et poids selon l’algorithme descente de gradient avec inertie. [Traingdm]
- Traingd(Gradient descent) : est une fonction de formation de réseau qui met à jour les valeurs biaiset poids selon l’algorithme descente de gradient. [Traingd]
- Trainbfg (quasi-Newton): est une fonction de formation de réseau qui met à jour les valeurs des poids selon la méthode de quasi-Newton BFGS.[Trainbfg]
- Traingda(Gradient descentwith adaptive learning rate) : est une fonction de formation de réseau qui met à jour les valeurs des poids selon l’algorithme descente de gradient avec taux d’apprentissage adaptatif.[Traingda]

IV.2.6.3 Architecture des couches cachées :

Une fois l’algorithme d’apprentissage choisi, nous procédons à la détermination du nombre de couches cachées et de neurones par couche cachée. Ceci a été effectué en faisant des simulations numériques avec différentes architectures du réseau (en faisant varier le nombre de neurones de chaque couche cachée). Nombre de neurones dans chaque couche cachée.

Tableau 3: Résumé de l’effet du nombre de couches cachées sur la performance.

Nombre de neurones dans chaque couche cachée	meilleure performance a l'epoch	temps d'apprentissage (min)	performance	
			mse	Taux de classification(accuracy)
[10 10]	36	00:43	0.0019	99,6
[20 20]	39	08:27	0.0014	99,8
[10 10 10]	40	02:22	0.0006	99,8
[20 10 10]	39	04:58	0.0013	99,93
[25 10 10]	52	09:38	0.0015	99,96
[30 10 10]	49	17:03	0.0008	99,97
[30 20 10]	63	39:25:00	0.0003	99,98

Les résultats semblent montrer une légère différence dans l'exécution avec le changement de la taille de la couche cachée. Les réseaux avec plus de neurones ont légèrement surpassé ceux avec un nombre moyen de neurones dans les couches cachées. Autrement dit, les erreurs associées diminuent dans des grands réseaux beaucoup plus rapidement que dans des réseaux à nombre moyen de neurones mais ça prend un temps considérable, prenant le cas de la première architecture et la dernière, la différence du temps d'apprentissage est presque 9 minutes. Basé sur les observations ci-dessus, l'analyse des résultats obtenus à partir des phases d'apprentissage, de test et de validation conduit à l'utilisation d'un réseau de 3 couches cachées comportant chacune respectivement 20, 10, 10 neurones.

Le tableau suivant récapitule les différents paramètres d'apprentissage et l'architecture du réseau de neurone à rétropropagation adapté à notre application.

Tableau 4: L'architecture du réseau de neurone à rétropropagation adapté à notre application.

Architecture des couches cachées	Algorithme d'apprentissage	Fonction de transfert des couches cachées	Fonction de transfert couche de sortie	Nombre d'epochs Maximum	Fonction de performance
[20 10 10]	Levenberg-Marquardt à Rétropropagation	sigmoïde	Pure linéaire	1000	Mse Précision. sensibilité. Accuracy

Au début du processus d'apprentissage, tous les poids de connexion ont été aléatoirement initialisés entre -1 et +1. Le nombre maximum de cycles ou itérations a été fixé à 1000 pour le modèle de réseau de neurone global, tous les exemples d'entrée ont été présentés dans un ordre aléatoire dans chaque cycle. L'apprentissage a été basé sur l'algorithme de Levenberg-Marquardt à rétropropagation avec les fonctions de transfert standard sigmoïde et Pure linéaire correspondantes respectivement aux couches cachées et de sortie.

IV.3 Résultats des tests effectués sur les données utilisant le model RNA construit :

Avant qu'un réseau de neurones puisse être employé avec n'importe quel degré de confiance. Il faut établir la validité des résultats qu'il produit. C'est la phase d'utilisation du

réseau ou phase de test. Il existe une autre phase dite de validation, souvent conduite avec la phase de test. Habituellement les phases de test et de validation impliquent d'évaluer la performance du réseau sur des données qui n'ont pas été employées pour la phase d'apprentissage. Les solutions désirées et celles produites par le réseau peuvent être comparées d'une façon qualitative, ou d'une façon quantitative en utilisant un test statistique.

IV.3.1 Test sur les données de deux vaches

Ces premiers tests ont été effectués sur les données de deux vaches, la fréquence des données est de 80hz nous avons utilisé un échantillon de 9000 exemples pour la classe « couché », 9000 pour la classe « debout », 3200 pour la classe « transition debout-couché » et 2300 pour la classe « transition couché-debout ».

Chaque exemple est une combinaison des données des capteurs ou du filtre, comme le montre la figure 35 ces combinaisons diffèrent selon le modèle RNA utilisé :

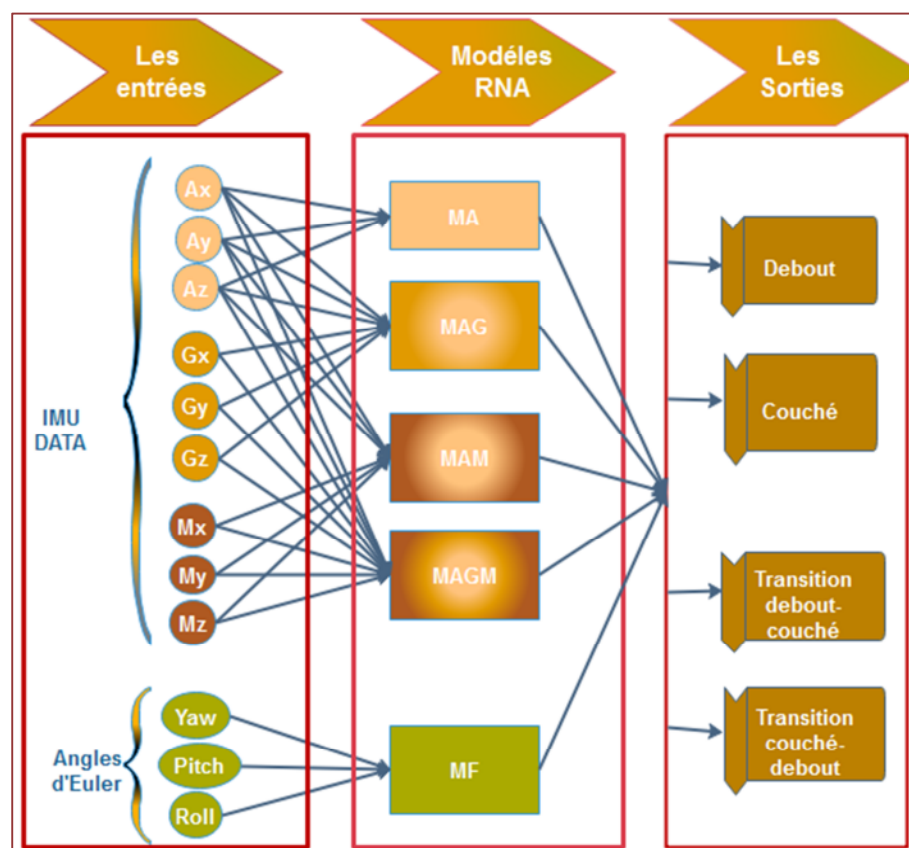


Figure 35: Les modèles RNA utilisés dans les tests.

- Un modèle à 3 neurones avec les données de l'accéléromètre (MA) comme des entrées.

- Un modèle à 3 neurones avec les données du filtreMadgwick (MF) comme des entrées.
- Un modèle à 6 neurones avec les données de l'accéléromètre et du magnétomètre (MAM) comme des entrées.
- Un modèle à 6 neurones avec les données de l'accéléromètre et de gyroscope (MAG) comme des entrées.
- Un modèle à 9 neurones avec les données de l'accéléromètre, de magnétomètre et du gyroscope (MAGM) comme des entrées de ce modèle.

Les résultatsdes tests sont présentés dans le tableau suivant :

Tableau 5: Performance du modèle RNA développé et testé sur les données de 2 vaches.

	Modèle RNA	Fonction Performance	Comportement				Taux de Classification (accuracy)
			Debout	Couché	Transition Debout - Couché	Transition Couché - Debout	
vache 1	MAGM	Précision	100,00	100,00	99,80	100,00	99,97
		sensibilité	100,00	100,00	100,00	99,70	
	MA	Précision	96,60	97,90	84,70	97,40	95,5
		sensibilité	97,40	99,80	87,80	82,70	
	MAM	Précision	99,50	99,00	98,10	99,10	99,2
		sensibilité	100,00	99,50	96,90	98,60	
	MAG	Précision	97,50	98,50	93,50	95,60	97,2
		sensibilité	98,50	100,00	89,20	91,90	
	MF	Précision	99,90	99,80	97,00	98,20	99,4
		sensibilité	99,90	100,00	97,60	96,40	
vache 2	MAGM	Précision	99,90	99,00	98,40	100,00	99,4
		sensibilité	100,00	99,60	97,80	98,60	
	MA	Précision	96,70	98,10	87,10	94,70	95,7
		sensibilité	96,60	99,80	87,60	87,50	
	MAM	Précision	100,00	98,30	99,40	99,70	99,2
		sensibilité	100,00	98,70	95,5	99,40	
	MAG	Précision	97,70	98,70	94,10	95,30	97,4
		sensibilité	97,80	99,90	91,40	94,20	
	MF	Précision	99,50	99,40	95,20	99,70	99
		sensibilité	98,90	99,90	96,60	98,00	

Comme première remarque les résultats obtenus en utilisant les données de la vache1 ou les données de la vache2 sont presque identiques.

Les résultats que nous voyons dans le tableau ci-dessus montrent que l'accéléromètre est le capteur qui a le plus d'informations précieuses avec un taux de classification de 94,7%, la sensibilité et la précision entre 96% et 99% pour les comportements (debout, couché) de 82% à 97% pour les deux transitions, tandis que le gyroscope et le magnétomètre améliorent la définition du comportement. Comparer les résultats des modèles «Accéléromètre Magnétomètre » et «Accéléromètre Gyroscope » on trouve que le premier a atteint un taux de 99,2% qui est un très bon résultat avec une différence de 2% par rapport au deuxième modèle.

Les meilleurs résultats ont été obtenus en utilisant le modèle qui utilise les données des trois capteurs (Accéléromètre Magnétomètre gyroscope) avec un taux de classification de 99,97 %, la sensibilité et la précision ont abouti 100 % de classification correcte pour les comportements (debout, couché) et entre [99 % ,100 %] pour les deux transits.

En ce qui concerne le modèle qui utilise les données du filtre le taux de succès est autour de 99,4 %, de même pour la sensibilité et la précision les valeurs sont proches de 100% pour les comportements (debout, couché) et entre [96%-99%] pour les deux transits.

Après l'analyse des résultats des 5 modèles de classification nous avons conclu que le modèle qui utilise les données des 3 capteurs et le modèle qui utilise les données du filtre ont réussi à aboutir des résultats très satisfaisant en termes de taux de classification, de sensibilité et de précision. Ainsi d'autres tests seront effectués pour valider ces résultats en utilisant ces deux modèles.

IV.3.2 Résultat des Tests sur des données avec différentes fréquences

Sachant que les fréquences élevée ont des influences négatives sur la durée de vie du capteur et que la quantité d'informations échangées augmente la consommation d'énergie du capteur, le but de ce test est de déterminer si la fréquence de 80hz est la fréquence optimale pour les modèles de classification ou bien on peut trouver une fréquence plus basse qui donne les meilleures performances. À cette fin, les mêmes tests qui ont été effectués dans le paragraphe précédent sur les modèles MF et MAGM ont été effectué avec différentes fréquences de 10Hz jusqu'à 80 Hz.

IV.3.2.1 Test sur le modèle RNA (MAGM)

Pour ce modèle nous avons effectué 8 tests dont la fréquence des données s'incrémente avec 10Hz pour chaque test donc de 10Hz jusqu'à 80Hz.

IV.3.2.1.1 Matrice de confusion

La matrice de confusion de l'ensemble des tests est présentée ci-dessous, les cellules colorées montrent le nombre d'exemples bien classifiés pour chaque comportement.

Tableau 6: Matrice de confusion pour la classification des quatre comportements (debout, couché, transition debout-couché, transition couché-debout) avec des fréquences de données de (10, 20, 30, 40, 50, 60, 70, 80 Hz). Utilisant les données des trois capteurs.

Fréquences	Comportement Observé	Comportement Predit			
		Debout	Couché	Transition Debout-Couché	Transition Couché-Debout
10Hz	Debout	1349	0	0	1
	Couché	0	1353	0	7
	Transition Debout-Couché	0	0	55	0
	Transition Couché-Debout	0	0	0	24
20Hz	Debout	1324	0	0	0
	Couché	0	1366	0	0
	Transition Debout-Couché	0	0	311	3
	Transition Couché-Debout	0	0	3	224
30Hz	Debout	1335	0	0	0
	Couché	0	1360	0	1
	Transition Debout-Couché	0	0	168	0
	Transition Couché-Debout	0	0	0	102
40Hz	Debout	1343	0	0	0
	Couché	0	1327	0	0
	Transition Debout-Couché	0	0	236	0
	Transition Couché-Debout	0	0	0	148
50Hz	Debout	1339	0	0	0
	Couché	0	1367	0	0
	Transition Debout-Couché	1	0	246	1
	Transition Couché-Debout	0	0	1	188
60Hz	Debout	1377	0	0	0
	Couché	0	1332	0	0
	Transition Debout-Couché	0	0	314	0
	Transition Couché-Debout	0	0	0	208
70Hz	Debout	1361	0	0	2
	Couché	0	1328	0	0
	Transition Debout-Couché	0	0	361	1
	Transition Couché-Debout	0	0	2	265
80Hz	Debout	1346	0	0	0
	Couché	0	1383	0	0
	Transition Debout-Couché	0	0	426	1
	Transition Couché-Debout	0	0	0	294

La matrice de confusion pour la fréquence de 10 Hz nous montre que 7 exemples du comportement « couché » sont classifiés comme des « transitions couché-debout », ce nombre a diminué lorsqu'on incrémente la fréquence, le nombre d'exemples qui n'ont pas été bien classifiés a varié entre 1 et 2 pour les deux transitions.

Pour les fréquences 40Hz et 60Hz nous remarquons que le modèle RNA a réussi à classifier correctement les 4 comportements (couché, debout, transition debout-couché » et transition couché-debout).

IV.3.2.1.2 Performance du modèle

A partir de la matrice de confusion, nous avons calculé la sensibilité la précision le taux de classification, ce sont des valeurs qui nous permettent d'évaluer la performance du modèle.

Tableau 7: Performance du modèle RNA dans la classification des quatre modes du comportement.

Fréquences	Fonction Performance	Données Accéléromètre Magnétomètre Gyroscope				Taux de Classification (accuracy)
		Debout	Couché	Transition Debout-Couché	Transition Couché - Debout	
10Hz	Précision	99,90	99,50	100,00	100,00	99,7
	sensibilité	100,00	100,00	100,00	75,00	
20Hz	Précision	100,00	100,00	99,00	98,70	99,79
	sensibilité	100,00	100,00	99,00	98,70	
30Hz	Précision	100,00	99,90	100,00	100,00	99,99
	sensibilité	100,00	100,00	100,00	99,00	
40Hz	Précision	100,00	100,00	100,00	100,00	100
	sensibilité	100,00	100,00	100,00	100,00	
50Hz	Précision	100,00	100,00	99,20	99,50	99,84
	sensibilité	99,90	100,00	99,60	99,50	
60Hz	Précision	100,00	100,00	100,00	100,00	100
	sensibilité	100,00	100,00	100,00	100,00	
70Hz	Précision	99,90	100,00	99,70	99,30	99,89
	sensibilité	100,00	100,00	99,40	98,90	
80Hz	Précision	100,00	100,00	99,80	100,00	99,97
	sensibilité	100,00	100,00	100,00	99,70	

Nous voyons que tous les résultats sont très proches, plus exactement la fréquence de 10Hz nous avons obtenu la plus basse valeur en terme de sensibilité qui est 75% pour le comportement de « transition couché-debout », ainsi pour les comportements « couché, debout » les valeurs sont 100%.

Les cellules vertes montrent les meilleures performances du modèle qui sont obtenues par les fréquences 40Hz et 60Hz avec un taux de classification correcte de 100%, la précision et la sensibilité ont atteint les 100% pour les quatre modes du comportement.

IV.3.2.2 Test sur le modèle RNA (MF)

Les mêmes tests qui ont été effectués sur le modèle précédent sont refaits avec ce modèle mais avec les données du filtre (Angles d'Euler).

IV.3.2.2.1 Matrice de confusion

La matrice de confusion de l'ensemble des tests est présentée ci-dessous, les cellules vertes montrent le nombre d'exemples bien classifiés pour chaque comportement.

Tableau 8: Matrice de confusion pour la classification des quatre comportements (debout, couché, transition debout-couché, transition couché-debout) avec des fréquences de données de (10, 20, 30, 40, 50, 60, 70, 80 Hz). utilisant les données du filtre.

Fréquences	Comportement Observé	Comportement Predit			
		Debout	Couché	Transition Debout Couché	Transition Couché-Debout
10Hz	Debout	1387	0	2	0
	Couché	0	1317	1	3
	Transition Debout-Couché	2	0	48	0
	Transition Couché-Debout	0	0	1	28
20Hz	Debout	1336	0	0	0
	Couché	0	1367	0	2
	Transition Debout-Couché	1	0	100	0
	Transition Couché-Debout	0	0	2	69
30Hz	Debout	1299	0	1	1
	Couché	0	1380	0	1
	Transition Debout-Couché	1	0	159	2
	Transition Couché-Debout	0	0	4	119
40Hz	Debout	1361	0	0	0
	Couché	0	1343	1	0
	Transition Debout-Couché	0	0	220	2
	Transition Couché-Debout	0	0	6	120
50Hz	Debout	1330	0	3	0
	Couché	0	1358	4	0
	Transition Debout-Couché	0	0	256	9
	Transition Couché-Debout	0	0	3	180
60Hz	Debout	1352	0	0	0
	Couché	0	1379	2	0
	Transition Debout-Couché	0	0	274	6
	Transition Couché-Debout	0	0	8	210
70Hz	Debout	1358	0	4	0
	Couché	0	1347	2	1
	Transition Debout-Couché	0	0	342	11
	Transition Couché-Debout	0	0	4	251
80Hz	Debout	1780	0	1	0
	Couché	0	1845	4	0
	Transition Debout-Couché	2	0	453	12
	Transition Couché-Debout	0	0	6	322

Pour ce modèle nous remarquons que les confusions se trouvent au niveau des deux transitions (debout-couché et couché-debout) et plus nous incrémentons la fréquence plus le nombre d'exemple qui ne sont pas bien classifié s'incrémentes

IV.3.2.2.2 Performance du modèle

Pour mieux comprendre les résultats de la matrice de confusion nous avons évalué les performances du modèle RNA, les résultats sont présentés dans le tableau suivant.

Tableau 9: Performance du modèle RNA dans la classification des quatre modes du comportement, le modèle à filtre.

Fréquence	Fonction Performance	Données du filtre				Taux de Classification (Accuracy)
		Debout	Couché	Transition Debout Couché	Transition Couché Debout	
10Hz	Précision	99,90	99,70	96,00	96,60	99,64
	sensibilité	99,90	100,00	92,30	90,30	
20Hz	Précision	100,00	99,90	99,00	97,20	99,68
	sensibilité	99,90	100,00	98,00	97,20	
30Hz	Précision	99,80	99,90	98,80	96,70	99,66
	sensibilité	99,90	100,00	97,00	96,70	
40Hz	Précision	100,00	99,90	98,70	95,20	99,7
	sensibilité	100,00	100,00	96,90	98,40	
50Hz	Précision	99,80	99,70	96,60	98,40	99,38
	sensibilité	100,00	100,00	96,20	95,20	
60Hz	Précision	100,00	99,90	97,90	96,30	99,5
	sensibilité	100,00	100,00	96,50	97,20	
70Hz	Précision	99,70	99,80	96,90	98,40	99,3
	sensibilité	100,00	100,00	97,20	95,40	
80Hz	Précision	99,90	99,80	97,00	98,20	99,4
	sensibilité	99,90	100,00	97,60	96,40	

Si on prend les comportements « couché » et « debout » nous constatons que la précision et la sensibilité prennent des valeurs entre 99% et 100% chose qui est bonne, mais qui ne nous permet pas de faire la distinction entre les fréquences, passant au deux autre comportement on trouve qu'il y a une baisse dans les valeurs de la sensibilité et la précision surtout pour la fréquence de 10Hz ou la sensibilité prend une valeur de 90% pour la « transition couché-debout ».

En général le taux de classification de toutes les fréquences est excellent mais le meilleur taux est obtenu par la fréquence de données de 40Hz avec 99,7%.

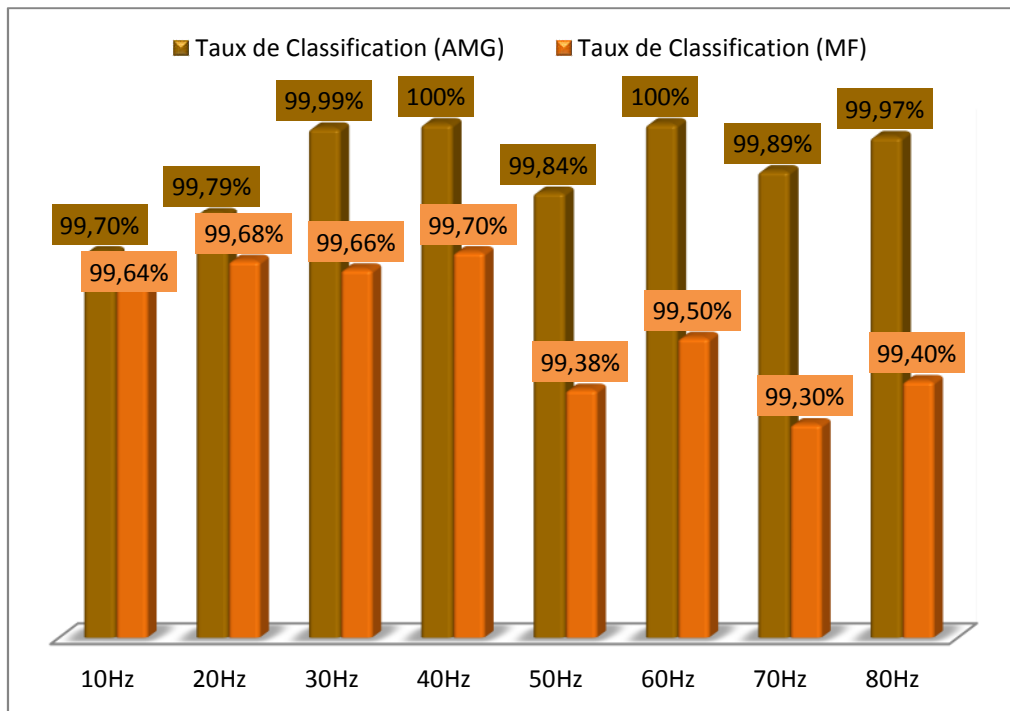


Figure 36: Taux de classification pour toutes les fréquences

Après avoir comparé les résultats des deux modèles présentés dans le graphe ci-dessus on trouve que la fréquence optimale est 40Hz pour laquelle nous avons eu un taux de classification de 100% pour le modèle qui utilise les données des trois capteurs et 99,7% pour le modèle qui utilise les données du filtre Madgwick.

IV.4 Conclusion

Dans ce chapitre nous avons réalisé l'objectif de notre travail, dont le but a été de construire un modèle RNA pour la classification du comportement des vaches laitières et de l'utiliser sur l'ensemble des données collectées par le système présenté dans le chapitre trois, les résultats ont été satisfaisants dans tous les tests effectués en utilisant les données du filtre Madgwick ou avec les données de l'IMU.

Conclusion générale

Conclusion générale

L'objectif de cette contribution portait sur la conception et la réalisation d'un modèle de classification capable de transformé les paramètres de comportement des vaches laitières dans les modes de comportements correspondants (debout, couché, transition debout-couché, transition couché-debout).

Ce travail a débuté d'abord, par une étude générale sur l'intérêt de la classification du comportement des vaches laitières, ainsi une présentation de quelques travaux réalisés dans ce domaine basés sur quatre méthodes de classification à savoir (logique floue, arbre de décision, les séparateurs a vaste marge(SVM), les réseaux de neurones. Nous avons choisi les réseaux de neurones artificiels, étant donné que les réseaux de neurones sont à l'heure actuelle offrant une meilleure précision à résoudre les problèmes de classification. Les différentes notions et aspects théoriques qui ont servi pour la partie développement dans ce travail ont été exposés dans le deuxième chapitre. Un aperçu général sur le système de collecte de paramètres de comportement des vaches est présenté dans le troisième chapitre. Ensuite nous avons développé dans un premier temps, un modèle de classification basé sur les RNA. Dans un second temps, une étude paramétrique pour l'ajustement des paramètres du modèle RNA a été faite et des tests de validation sont présentés à la fin du quatrième chapitre.

Ce projet nous a permis de participer à une expérience intéressante qui a renforcé et enrichi nos connaissances et ce à travers :

- l'acquisition d'une précieuse expérience relative au fonctionnement d'une ferme et d'avoir contribué à l'amélioration des besoins des agriculteurs au même temps les besoins des animaux (vache).
- Nous avons eu la chance de pratiquer une méthode de classification qui est les « réseaux de neurones artificiels RNA » qui va forcément nous servir pour d'autres projets dans l'avenir.
- l'amélioration de nos connaissances en utilisation du logiciel MATLAB.

Pour conclure, on souhaite que ce modeste travail puisse répondre favorablement aux besoins des futurs utilisateurs et servir comme outil d'aide et de documentation pour les promotions à venir.

Bibliographie

Bibliographie

- [AFIMILK] Vaches Laitières 2018, lien :
<http://www.afimilk.com/fr/needs-solutions/cows/health>
- [Anderson, 2001] Anderson, NG. Time lapse video opens our eyes to cow comfort and behavior, Proc American Assoc Bovine Practitioners Conference, 2001, no 34, p. 35-42.
- [AssaadM] Mohamed Assaad HAMIDA, Introduction à la commande par réseaux de neurones, le lien : https://elearn.univ-ouargla.dz/.../INTRODUCTIONAUXMETHO/.../Cours_HAMIDA...
- [BENDAOU D 2014] BENDAOU D YOUCEF, Prédiction Des Résistances Mécaniques Des Bétons à Base Des Ciments Composés En Utilisant Les Réseaux Neurones Artificiels, le lien :
<https://bu.umc.edu.dz/theses/gcivil/BEN6585.pdf>
- [BENNANI 2006] Y. BENNANI. Apprentissage Connexionniste. Hermès science, 2006.
- [Bouk 06] Boukadoum, Mounir, Évaluation de variables hydro-électriques: une comparaison de différentes approches d'apprentissage automatique, Notes de séminaire en informatique cognitive, Université du Québec à Montréal, 23 février 2006.
- [BV-2014] Robin Vergonjeanne « Mammites L'impact économique atteint 230 €/vache/an » lien :
<http://www.web-agri.fr/conduite-elevage/sante-animale/article/l-impact-economique-atteint-230-vache-an-1184-97330.html>
- [ciwf 2018] Des systèmes d'élevage variés 2018, lien :
<https://www.ciwf.fr/animaux-de-ferme/vaches-laitieres/elevage-standard/>

- [Claude 1992] Claude TOUZET, LES RESEAUX DE NEURONES ARTIFICIELS INTRODUCTION AU CONNEXIONNISME, le lien : www.touzet.org/Claude/Web-Fac-Claude/Les_reseaux_de_neurones_artificiels.pdf
- [CM2010] A. CORNUEJOLS et L. MICLET. Apprentissage Artificiel : Concepts et algorithmes. Eyrolles, 2ieme édition, 2010.
- [cowAlert 2017] CowAlert. 2017 lien: <http://www.icerobotics.com/products/#cowaalert>.
- [cow-intelligence 2016] Allflex. Cow Intelligence. lien: <http://www.scrdairy.com/cow-intelligence>, 2016
- [DREYFUS] G. DREYFUS, LES RÉSEAUX DE NEURONES ; le lien: http://www.neurones.espci.fr/Articles_PS/GAMI.pdf
- [DSM2002] G. DREYFUS, M. SAMUELIDES, J. M. MARTINEZ, M. B. GORDON, F. BA-DRAN, S. THIRIA, et L. HÉRAULT. Réseaux de Neurones : Méthodologie et applications. Eyrolles, 2002.
- [DUCH 1999] W. DUCH et N. JANKOWSKI. « Survey of neural transfer functions ». Neural computing Surveys, 2(1-2):163–212, 1999.
- [Éric 1993] Éric DAVALO, Patrick Naim « Des réseaux de neurones»; 2ème Editions Eyrolles 1993.
- [Fehlmann 2017] Fehlmann G, O’Riain MJ, Hopkins PW, O’Sullivan J, Holton MD, Shepard ELC, King AJ. Identification of behaviours from accelerometer data in a wild social primate. AnimBiotelem. 2017;5(1):6. Lien: <https://doi.org/10.1186/s40317-017-0121-3>.
- [Franck 2005] Franck Morand, «Techniques d’observation sans capteur de vitesse en vue de la commande des machines asynchrones ». Thèse CEGELY,

Lyon (INSA de Lyon), 2005.

- [gotronic 2018] <https://www.gotronic.fr/art-module-arduino-nano-12422.htm> 2018
- [Grünewälder 2012] Grünewälder S, Broekhuis F, Macdonald DW, Wilson AM, McNutt JW, Shawe-Taylor J, Hailes S. Movement activity based classification of animal behaviour with an application to data from cheetah (*Acinonyx jubatus*). PLoS ONE. 2012;7(11):49120.
- [HANZEN et al, 1996] HANZEN C., HOUTAIN JY., LAURENT Y et ECTORS F, (1996). Influence des facteurs individuels et de troupeau sur les performances de reproduction bovine. Ann.Méd.Vét, 140, PP195-210.
- [HERTZ 1991] J. HERTZ, A. KROGH, et R. G. PALMER. Introduction to the theory of neural computation. Lecture Notes, Santa Fe Institute, Addison Wesley Publishing, 1991.
- [HK2001] J. HAN et M. KAMBER. Datamining : Concepts and Techniques. Morgan Kauffman Publishers, 2nde edition, 2001.
- [Hugues G] Hugues Gérard NONO WOUAFO ; Architectures matérielles numériques intégrées et réseaux de neurones à codage parcimonieux (thèse de doctorat) ; le lien : <https://tel.archives-ouvertes.fr/tel-01332180/document>
- [IMU 2018] IMU 9DOF MPU9250, Lien : <https://drotek.com/shop/fr/imu/421-mpu9250-gyro-accelerometre-magnetometre.html>, 2018
- [KZ2002] W. KLÖSGEN et J. M. ZYTKOW. Handbook of Datamining and knowledge discovery. Oxford University Press, 2002
- [M.S. Shahriar 2016] M.S. Shahriar, D. Smith, A. Rahman, M. Freeman, J. Hills, R. Rawnsley, et al. Detecting heat events in dairy cows using accelerometers and unsupervised learning, Elsevier Comput Electron Agric, 128 (2016), pp. 20-26

- [MANIRVA
2017] D. Gutierrez-Galan, Juan P. Dominguez-Morales, E. Cerezuela-Escudero, A. Rios-Navarro, R. Tapiador-Morales, M. Rivas-Perez, M. Dominguez-Morales, A. Jimenez-Fernandez, A. Linares-Barranco « Embedded neural network for real-time animal behavior classification » , 2017 ,p14-20
- [MartiskainenP
2009] Martiskainen P, Järvinen M, Skön J-P, Tiirikainen J, Kolehmainen M, Mononen J. Cow behaviour pattern recognition using a three-dimensional accelerometer and support vector machines. *ApplAnimBehav Sci.* 2009;119(1):32–8.
- [McClune 2014] McClune DW, Marks NJ, Wilson RP, Houghton JD, Montgomery IW, McGowan NE, Gormley E, Scantlebury M. Tri-axial accelerometers quantify behaviour in the Eurasian badger (*Melesmeles*): towards an automated interpretation of field data. *AnimBiotelem.* 2014;2(1):5.
- [McCluneDW
2015] McClune DW, Marks NJ, Delahay RJ, Montgomery WI, Scantlebury DM. Behaviour-time budget and functional habitat use of a free-ranging European badger (*Melesmeles*). *AnimBiotelem.* 2015;3(1):7. Lien: <https://doi.org/10.1186/s40317-015-0025-z>
- [MezhoudN] Nassima Mezhoud, Une approche hybride pour la segmentation d'images Application aux images médicales ; le lien : <https://bu.umc.edu.dz/theses/informatique/MEZ6197.pdf>
- [MNN2005] E. MEPHU NGUIFO et P. NJIWOUA. Treillis de concepts et classification supervisée. Dans *RSTI-TSI*, volume 24 (4), pages 449–488, 2005.
- [mol 2001] R. M. de Mol* , W. E. Woldt , Application of Fuzzy Logic in Automated Cow Status Monitoring, *Sci.* 84:400–410, 2001, p400-402.

- [Moreau M 2009] Moreau M, Siebert S, Buerkert A, Schlecht E. Use of a tri-axial accelerometer for automated recording and classification of goats grazing behaviour. *ApplAnimBehav Sci.* 2009;119(3):158–70.
- [Nathan R 2012] Nathan R, Spiegel O, Fortmann-Roe S, Harel R, Wikelski M, Getz WM. Using tri-axial acceleration data to identify behavioural modes of free-ranging animals: general concepts and tools illustrated for griffon vultures. *J Exp Biol.* 2012;215(6):986–96.
- [NEBEL 2003] NEBEL R.L (2003). Components of a successful Heat Detection Program - *Advances in dairy Technology* 15 PP.191-203
- [NEDJARI 1998] T. NEDJARI. Réseaux de neurones artificiels et connaissances symboliques : Insertion, Raffinement et Extraction. Université de Paris 13, Thèse de Doctorat, 1998.
- [NJONG, 2006] N JONG, Adaptation des vaches à haut potentiel de production laitière en milieu tropical : cas de bovins Holstein introduits en 2002 dans la ferme de Wayembam au Sénégal. Thèse pour obtenir le Grade de Doctorat de Médecine Vétérinaire. P91.
- [P.L. Greenwood 2016] P.L. Greenwood, D.R. Paull, J. McNally, T. Kalinowski, D. Ebert, B. Little, et al. Use of sensor-determined behaviours to develop algorithms for pasture intake by individual grazing cattle, *Crop Pasture Sci* (2016), 10.1071/CP16383.
- [P.L. Greenwood 2014] P.L. Greenwood, P. Valencia, L. Overs, D.R. Paull, I.W. Purvis New ways of measuring intake, efficiency and behaviour of grazing livestock *Animal Prod Sci*, 54 (2014), pp. 1796-1804
- [Painter MS 2016] Painter MS, Blanco JA, Malkemper EP, Anderson C, Sweeney DC, Hewgley CW, Červený J, Hart V, Topinka V, Belotti E, Burda H,

- Phillips JB. Use of bio-loggers to characterize red fox behavior with implications for studies of magnetic alignment responses in free-roaming animals. *AnimBiotelem*. 2016;4(1):20.
Lien: <https://doi.org/10.1186/s40317-016-0113-8>.
- [PYH1995] R. PAREKH, J. YANG, et V. HONAVAR. « Constructive Neural Networks Learning Algorithms for Multi-Category Classification ». Tech report isucstr 95-15, Department of Computer Science Iowa State University, 1995. Describes the package natbib.
- [PYH1997a] R. PAREKH, J. YANG, et V. HONAVAR. « Constructive Neural Network learning Algorithm for Multi-Category Real Valued Pattern Classification ». Rapport de recherche, Department of Computer Science Iowa State University, 1997.
- [RADJA2010] RADJA MAROUF, Amélioration de l'apprentissage d'un Modèle Neuronal pour la reconnaissance des anomalies cardiaques
- [Robert B 2009] Robert B, White B, Renter D, Larson R. Evaluation of three-dimensional accelerometers to monitor and classify behaviour patterns in cattle. *Comput Electron Agric*. 2009;67(1):80–4.
- [Shamoun 2012] Shamoun-Baranes J, Bom R, van Loon EE, Ens BJ, Oosterbeek K, Bouten W. From sensor data to animal behaviour: an oystercatcher example. *PLoS ONE*. 2012;7(5):37997.
- [Soltis J 2012] Soltis J, Wilson RP, Douglas-Hamilton I, Vollrath F, King LE, Savage A. Accelerometers in collars identify behavioural states in captive African elephants *Loxodonta africana*. *Endanger Species Res*. 2012;18(3):255–63.
- [TAYARI 2015] TAYARI LASSAAD, Introduction aux systèmes embarqués 2015, p 6.
[Trainbfg] Trainbfg, lien: <https://www.mathworks.com/help/nnet/ref/trainbfg.html>

- ,2018
- [Traingd] Traingd, lien : <https://www.mathworks.com/help/nnet/ref/traingd.html>
,2018
- [Traingda] Traingda,lien: <https://www.mathworks.com/help/nnet/ref/traingda.html>
,2018
- [Traingdm] Traingdm, lien:
<https://www.mathworks.com/help/nnet/ref/traingdm.html>
,2018
- [Trainlm] Traingda, lien:<https://www.mathworks.com/help/nnet/ref/trainlm.html>
,2018
- [VAILES 1990] VAILES L.D, (1990). Influence of footing surface on mounting and other sexual behaviour of estrus Holstein Cows – J. Anim. Sci. 68. PP2333.-2339
- [Vázquez 2015] Jorge A. Vázquez Diosdado¹, Zoe E. Barker , Holly R. Hodges, Jonathan R. Amory , Darren P. Croft , Nick J. Bell and Edward A. Codling,
Classification of behaviour in housed dairy cows using an accelerometer-based activity monitoring system,VázquezDiosdado et al. Animal Biotelemetry (2015) 3:15 .
- [VázquezDiosdado 2015] VázquezDiosdado JA, Barker ZE, Hodges HR, Amory JR, Croft DP, Bell NJ, Codling EA. Classification of behaviour in housed dairy cows using an accelerometer-based activity monitoring system. AnimBiotelem. 2015;3(1):15. Lien: <https://doi.org/10.1186/s40317-015-00458>.
- [Vincent] Vincent Kerhoas, Projet Réseau de Capteurs
Lien : https://www.enib.fr/~kerhoas/rescapt_cours_mpu9250.html ,
2018
- [Volpi] Karine Volpi, Réseau de neurones ;le lien :
https://nanopdf.com/download/karinevolpi-1-reseau-de-neurones_pdf
- [Watanabe N2008] Watanabe N, Sakanoue S, Kawamura K, Kozakai T. Development of an

automatic classification system for eating, ruminating and resting behaviour of cattle using an accelerometer. Grassl Sci. 2008;54(4):231–7.

[Weizheng2015] WeizhengShen, Congcong Chen, ShuangZheng, Shanjun He and Mingda Li,
The Design of System about Cow Activity Based on SVM, International Journal of Smart Home Vol. 9, No. 3 (2015)

[Wikipedia] Machine à vecteurs de support, lien :
https://fr.wikipedia.org/wiki/Machine_%C3%A0_vecteurs_de_support

[Xuezhi 2001] Wu, Xuezhi, and Lipei Huang, « Direct torque control of three-level inverter using neural networks as switching vector selector »; Industry Applications Conference, IEEE. Vol. 2, pp 939-944; 2001.

[YANG 1999] J. YANG, R. PAREKH, et V. HONAVAR. « Distal : An Inter-pattern Distancebased Constructive Learning Algorithm ». Intelligent Data Analysis, 3 :55–73, 1999