

REPUBLIQUE DU SENEGAL



Un Peuple – Un But – Une Foi

-----  
MINISTERE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE

UNIVERSITÉ GASTON BERGER DE SAINT-LOUIS



L'Excellence au service du Développement

**U.F.R DE SCIENCES APPLIQUÉES ET DE  
TECHNOLOGIE**

**MASTER 2 INFORMATIQUE**  
**Gestion de Données et Ingénierie Logicielle (GDIL)**

**MÉMOIRE : Réseaux de Neurones et IoT Appliquées dans le Domaine  
Environnemental**

Présenté par :  
Papa Laity NDIAYE

Sous la direction de :  
M. DEMBELE

2019/2020

## Table des matières

### Introduction

Chapitre I : Généralités	12
I. Internet des Objets	12
I. 1 Caractéristiques de l'IdO .....	12
I. 2 Architecture de l'IdO .....	14
I. 3 Les Applications de l'IdO .....	16
I. 4 Avantages et Inconvénients de l'Internet des Objets .....	17
II. Edge Computing	19
II. 1 Edge Computing: Définition .....	19
II. 2 Edge Computing : Utilité.....	21
II. 3 Edge Computing vs Cloud Computing.....	22
II. 3. 1 L'Edge Computing est-il plus sécurisé que le Cloud Computing ?	23
II. 4 Quels sont les avantages et les cas d'usage de l'Edge Computing ? .....	23
III. Intelligence Artificielle	26
III. 1 Machine Learning (Apprentissage Automatique AA).....	27
III. 2 Qualité d'apprentissage et Étapes d'un projet ML .....	28
III. 3 Méthodes d'Apprentissage .....	29
III. 3. 1 Apprentissage Supervisé	29
III. 3. 2 Apprentissage Non-Supervisé	30
III. 3. 3 Apprentissage Semi-Supervisé	32
III. 3. 4 Apprentissage par Renforcement	32
III. 3. 5 Apprentissage par Transfert	32
III. 4 Background Machine Learning .....	32
III. 4. 1 Classification	32
III. 4. 2 Régression	36
III. 4. 3 Clustering	37

III. 4. 4	Apprentissage par renforcement	38
III. 5	Avantages et Controverses de l'AA .....	41
III. 5. 1	Avantages de L'Apprentissage Automatique	41
III. 5. 2	Controverses	41
Chapitre II :	Etat de l'Art	43
I.	Machine Learning appliquée dans le domaine des IdO (Exécution des données du Domaine Agricole)	43
II.	Etude sur les différentes méthodes de prédiction de l'Humidité du sol	46
Chapitre III :	Traitement en périphérie et Évaluation de performances	51
I.	Architectures (Système et Gateway)	52
I. 1	Architecture Système.....	52
I. 2	Architecture déployée sur le Gateway .....	54
II.	Jeux de Données et Prévisualisation	57
II. 1	Les Données Environnementales.....	57
II. 2	Prévisualisation des données .....	59
III.	Modèles d'apprentissage	64
III. 1	Perceptron Multi Couche.....	65
III. 2	Support Vector Machine pour la Régression SVR.....	66
III. 3	Modèle de Régression (Réseaux de Neurones) avec une couche cachée.....	67
III. 4	Modèle de Régression (Réseaux de Neurones) avec plusieurs couches cachées .....	68
IV.	Comparaison des Modèles:	70
IV. 1	Mise en pratique du Système / Expérimentations Gateway .....	71

## Conclusion

## **Table des Illustrations**

Figure 1: Architecture IdO .....	15
Figure 2: les différentes couches de traitement et leurs capacités.....	21
Figure 3: Apprentissage supervisé .....	30
Figure 4: Apprentissage non supervisée.....	31
Figure 5: Opérations internes de quelques modèles de classification .....	33
Figure 6: Réseaux de neurones récurrents.....	35
Figure 7: Différentes étapes de la tâche de classification et de l'exécution .....	35
Figure 8: Exemples de (a) K-means Clustering avec K=3, (b) hierarchical Clustering.....	38
Figure 9: Comportement de l'apprentissage par Renforcement et de leurs interactions .....	39
Figure 10: Architecture système .....	54
Figure 11: Architecture Gateway .....	56
Figure 12: Organisation Parcelles et disposition des capteurs .....	58
Figure 13: Entête des données de la station météorologique .....	60
Figure 14: Évolution de l'humidité du sol, de la Température de L'Air et de l'irrigation du champ à la date du 23 Février .....	61
Figure 15: Évolution de l'humidité du sol, de la Température de L'Air et de l'irrigation du champ au mois de Février. ....	62
Figure 16: Évolution de la pression, de la Température de L'Air, de l'humidité de l'air, de l'irrigation du champ et de la vitesse du vent au mois de Février.....	63
Figure 17: Raspberry Pi 3.....	64
Figure 18: Diagramme en Barre MSE Scaling Data(a) & No Scaling Data (b) Multi layer Perceptron.....	66
Figure 19: Diagramme en Barre MSE No Scaling Data (a) et Scaling Data (b) SVR.....	67
Figure 20: Courbe d'Évolution MSE Validation Data (Scale Data).....	68
Figure 21: Courbe d'Évolution MSE Validation Data(1) & Diagramme en Barre MSE No Scaling Data .....	68
Figure 22: Courbe d'évolution de décroissance du MSE Données Brutes(a) et Données Standardisées (b) .....	69
Figure 23: Diagramme en Barre MSE modèle de régression à plusieurs Couches Données Brutes (1) et Données Standardisées (2) .....	70

Figure 24: Diagramme de Comparaison (MSE) des différents modèles: Données Brutes(a) et Données Standardisées(b) .....	71
---	----

## *Remerciements*

*Alhamdoulilah,  
Je ne me suis pas fait seul,  
Je me suis fait avec ceux qui cacheront mes erreurs.  
Par la grâce d'ALLAH  
Je tiens à remercier l'ensemble des personnes  
Qui m'ont de près ou de loin assisté à la réalisation de ce document.  
Papa, Maman, Baye Moustapha  
Je vous serez gré pour votre accompagnement et vos vœux.  
Je remercie ma famille et mes amis qui ont été toujours là pour moi.  
Sans oublier Kamsa et les personnes qui fréquentent sa place.  
C'est un cadre d'échange idéal pour divers type de connaissance.  
Mention Spéciale à Ousmane DIENG et à Babacar DIOP  
Vous avez été la tout au long de l'aventure et n'avez ménagé aucun effort  
Pour la réalisation de ce projet...*

## *Dédicace*

*Je dédie ce document à ma mère*

# Introduction

Grâce aux avancées rapides dans différentes technologies telles que les semi-conducteurs, les technologies de communication sans fil et les capteurs, nous avons pu constater une diffusion massive des équipements interconnectés. Ces appareils embarqués dotés de capteurs et de dispositifs de communication, constituent les composants clés de l'Internet des objets (IdO). L'Internet des objets ou IdO (en anglais (*the Internet of Things* ou IoT) peut être défini comme étant une infrastructure dynamique d'un réseau global. Ce réseau global a des capacités d'auto-configuration basées sur des standards et des protocoles de communication interopérables. Dans ce réseau, les objets physiques et virtuels ont des identités, des attributs physiques, des personnalités virtuelles et des interfaces intelligentes, et ils sont intégrés au réseau d'une façon transparente [1].

L'appellation désigne un nombre croissant d'objets connectés à l'Internet permettant ainsi une communication entre nos biens dits physiques et leurs existences numériques. Ces formes de connexions permettent de rassembler de nouvelles masses de données sur le réseau et donc, de nouvelles connaissances et formes de savoirs.

Les équipements connectés interagissent avec le monde physique à l'aide de capteurs et/ou d'actionneurs. Chacun de ces derniers (capteurs / actionneurs) peut être catégorisé. On distingue des équipements simples qui n'interagissent qu'avec leur environnement. Et d'autres capteurs qui peuvent disposer d'autres ressources supplémentaires à l'exemple d'une capacité d'exécution ou des modules de communication réseau. On peut les appeler des appareils intelligents. L'architecture de l'IdO constitue un système distribué se composant de plusieurs couches de traitement. Les données collectées peuvent être exploitées sur l'une quelconque de ces couches ou même par l'utilisateur (sur ordinateur).

D'une part, les équipements IdO se trouvant à la périphérie ne disposent pas souvent de capacités pour traiter les données d'entrée et initier des actions. Soit les données sont récupérées en local et traitées sur une machine plus puissante ou elles sont transférées sur le Cloud qui dispose assez de capacité pour traiter ces dernières.

D'autre part, les équipements intelligents peuvent traiter les données collectées, évaluer la situation et agir de manière indépendante.



Par exemple, l'utilisation d'une application sur smartphone pour allumer le système de chauffage à distance dépend de la décision de l'utilisateur, tandis qu'un thermostat intelligent peut régler le chauffage de manière autonome en fonction de l'occupation de la maison ou du confort de l'utilisateur.

Dans le paradigme traditionnel du Cloud computing, l'ensemble du traitement est effectué sur le Cloud, ce qui rend les dispositifs/actionneurs IdO distants et connectés. Cependant, le nombre de périphériques IdO augmente rapidement et la quantité énorme de données collectées devient de plus en plus difficile à gérer par le Cloud. Des spécialistes comme Helder Antunes de Cisco estiment qu'il est irréaliste de penser que toutes ces données pourront être transmises entre leurs sources et les Data Center (centres de données) Cloud de façon stable et rapide afin d'être analysées [2]. Les raisons en sont l'énorme charge de travail sur le réseau IoT et le temps de latence long et peu fiable.

De plus le Cloud computing n'est pas accessible dans les zones dépourvus de réseau ne permettant pas de faire usage d'internet pour la communication entre objets ainsi que le transfert des données vers les Data Center pour un éventuel traitement.

Serait-il possible de trouver d'autres issues pour faire face à ce problème de transfert de données afin d'optimiser le temps d'exécution et d'alertes en temps pour une prise de décision?

Est-elle une obligation de disposer de l'accès à Internet pour pouvoir procéder au traitement des données?

Aurait-on besoin de transférer les données jusqu'au Cloud pour pouvoir y extraire des informations dans le but de prendre des décisions en rapport au contexte?

Le paradigme de traitement des données à la périphérie du réseau (Edge Computing) suggère de repousser le traitement des données jusqu'à la couche périphérique de l'IdO (comprenant des passerelles et des périphériques intégrés) près du lieu de collecte des données. Dans de nombreuses applications IdO, le calcul peut être réparti sur différentes couches. Par exemple, les équipements IdO (microcontrôleurs et passerelle) peuvent effectuer le prétraitement sur les données et transmettre les résultats intermédiaires au Cloud, où le reste du traitement est effectué.

Dans le domaine du traitement des données, l'Apprentissage Automatique (AA) est une technique essentielle pour l'IdO qui pourrait faciliter l'inférence d'informations, le traitement de données et une meilleure gestion de ces objets. Du traitement de données volumineuses sur le Cloud à l'intelligence intégrée, l'Apprentissage Automatique (AA) peut être une solution prometteuse efficace dans différents domaines d'application de l'IdO.

Afin de pouvoir mieux appréhender le thème que l'on va aborder, il serait bien de contextualiser et faire une étude de l'ensemble des domaines qui entrent en jeu. C'est ainsi que l'on abordera la thématique de l'IdO en nous intéressant à l'Edge Computing qui consiste à faire du traitement en périphérie.

De plus, avant de pouvoir extraire des informations sur les données, il serait utile de se documenter sur les techniques habilitées à le faire particulièrement l'apprentissage automatique, son utilisation en périphérie et son application dans les tâches de traitement et de gestion de données.

Une fois le contexte établi, nous essayerons de proposer une solution nous permettant de faire du traitement de données en périphérie et faire abstraction du transfert de données sur le cloud ainsi que les contraintes associées.

Les principales parties de ce travail de recherche sont les suivantes:

Dans la première partie, nous ferons une étude générale des différents domaines que l'on abordera au cours de notre mémoire.

- Nous parlerons de L'IdO, des principales définitions, de son architecture ainsi que de son cycle de vie.
- A cela s'ajoute une description de l'edge computing comme étant une perspective de traitement de données dans l'objectif d'optimiser la latence du cloud ainsi que le traitement de données en temps réel.
- Nous présentons une vue complète de l'apprentissage automatique. Cette présentation sera orientée sur les techniques de l'AA et leurs caractéristiques. Nous verrons aussi le rôle de ce dernier sur l'IdO en analysant l'impact du traitement suivant les différentes couches du Cloud et de la périphérie.

Dans la seconde partie, nous aborderons l'ensemble des travaux sur le traitement en périphérie de l'IdO dans le domaine de l'agriculture qui ont été présentés en se référant aux travaux

scientifiques publiés. Nous présenterons de même un ensemble de techniques de prédiction de l'humidité du sol utilisées et leurs résultats en comparaison avec d'autres techniques

Dans la troisième partie, nous proposons une contribution essayant d'être un apport contre les problèmes posés. Un système en mesure de récupérer les données et de les interpréter en périphéries tout en faisant abstraction au cloud ou serveur distant. Ceci optimisant le temps de transmission et d'interprétation des données ainsi qu'une prise de décision en temps réel et sans accès à internet. Néanmoins cette solution est juste une alternative et présente certaines limites. En dernière partie, nous ferons une comparaison de l'ensemble des techniques de traitement. Ce dernier sera un moyen de voir l'apport de notre solution, décèlera les contraintes dont nous serons exposés et les perspectives qui nous permettront d'améliorer nos résultats.

## Chapitre I : Généralités

### I. Internet des Objets

L'Internet des Objets (IdO) est un ensemble (réseaux) d'équipements connectés entre eux pour échanger des données et des informations à travers des capteurs et des actionneurs. Ces derniers sont embarqués dans les équipements ce qui leur permet l'échange [3].

En termes simples, IdO signifie le fait de permettre à certains équipements technologiques de se connecter à internet ou entre eux afin de moderniser et rendre la tâche facile à certaines activités.

L'expression "Objets" se réfère ici aux appareils des objets connectés. Ces derniers pourraient être des caméras, capteurs, ou tout autre équipement comme des appareils mobiles, PC, véhicules... Ce sont des objets dans le réseau qui peuvent disposer de leur propre adresse @IP ou être reliés à un dispositif qui en possède et qui ont la possibilité d'envoyer comme de recevoir des données sur le réseau. Les données collectées sur les appareils peuvent être analysées en local comme sur un Cloud. Ces appareils peuvent être connectés à un Gateway qui est une passerelle dotée d'une capacité de communication. Différents protocoles de communication sont utilisés suivant le type d'application déployée. Les appareils eux même font tout le travail sans l'intervention humaine.

#### I. 1 Caractéristiques de l'IdO

- L'IdO fournit des services au niveau mondial par l'interconnexion de divers périphériques physiques utilisant une infrastructure globale. Il est basé sur des technologies existantes et en évolution.

- **Système Intelligent:** L'IdO peut être considéré comme un système intelligent car étant une combinaison de matériels, de logiciels, d'algorithmes et de systèmes complexes. L'association de ces derniers influe sur l'amélioration de la capacité des objets ce qui permet une interaction entre les différents appareils.
- **Connectivité:** Sa connectivité lui permet une interaction de divers objets d'utilisation quotidienne. Cela contribue à l'intelligence globale du réseau. Cela ouvre également la voie à de nouvelles opportunités de marché en créant un réseau d'objets et d'applications intelligentes. De plus, le réseau sera plus accessible et compatible.
- **Dynamicité:** Les équipements IdO capturent les données de leur environnement. Cela se fait par les changements dynamiques qui se produisent autour de ces appareils. Les capteurs sont un composant important de l'IdO sans lequel les modifications de l'environnement ne peuvent être ni détectées ni mesurées. Ces capteurs interagissent avec l'environnement pour détecter et collecter des données. Ces dernières détectées par le capteur sont fondamentalement les données de l'environnement qui peuvent fournir des informations précieuses.
- **Complexité Interopérabilité:** Dans un proche avenir, le nombre d'équipements connectés au réseau pour la communication sera beaucoup plus important qu'aujourd'hui. En outre, il deviendra beaucoup plus complexe de les gérer et de gérer les données de ces périphériques. Selon des statistiques, plus de 5 millions de nouveaux appareils sont connectés chaque jour et leur nombre ne fera qu'augmenter.
- **Hétérogénéité:** Une des principales caractéristiques de l'internet des objets est la diversité ou l'hétérogénéité. Les périphériques IdO ont différentes plateformes matérielles et réseaux et peuvent communiquer avec d'autres périphériques via différents réseaux. Le réseau IdO est capable de prendre en charge la connectivité entre des réseaux distincts. Les principales exigences de cette diversité sont l'évolutivité, la modularité, l'extensibilité et l'interopérabilité.

- **Sécurité** : En termes de sécurité, il en existe actuellement des problèmes de même que sur la confidentialité avec le réseau IdO qui disparaîtront avec davantage de développement dans ce domaine. Il est très important de sécuriser les données lors de leur transfert entre appareils.

## I. 2 Architecture de l'IdO

Il n'y a pas d'architecture standard pour l'IdO. L'une de ses architecture proposée est principalement appelée architecture à quatre étapes (*figure 1*).

L'étape 1 comprend les capteurs et les actionneurs (phase d'acquisition). L'étape 2 comprend les systèmes d'agrégation et le convertisseur analogique-numérique. À l'étape 3, le traitement des données est effectué par une technologie. À l'étape 4, les données sont déplacées vers les systèmes du centre de données.

Dans la première étape, les données sont collectées par les capteurs de l'environnement ou d'un objet et sont converties en données utiles. Un actionneur agit également en fonction du changement des conditions physiques. Les capteurs et actionneurs sont utilisés dans presque tous les domaines, des industries aux soins de santé. La portée de l'IdO augmente jour après jour en fonction des demandes. La puissance de traitement des appareils IoT est limitée. Les données peuvent être traitées sur le capteur.

Dans la deuxième étape, il y a la passerelle Internet. Les données collectées par les capteurs sont sous forme analogique. Il est converti en format numérique à l'aide de systèmes d'acquisition de données (DAS) pour un traitement ultérieur. Le DAS effectue des conversions analogiques-numériques avec agrégation de données. La passerelle Internet reçoit ces données agrégées et numérisées et assure leur routage via le réseau sans fil ou câblé. Les données analogiques sont converties en données numériques, car les flux analogiques créent de gros volumes de données. De plus, les données analogiques nécessitent des logiciels spécialisés pour le traitement en fonction de leurs timings et structures.

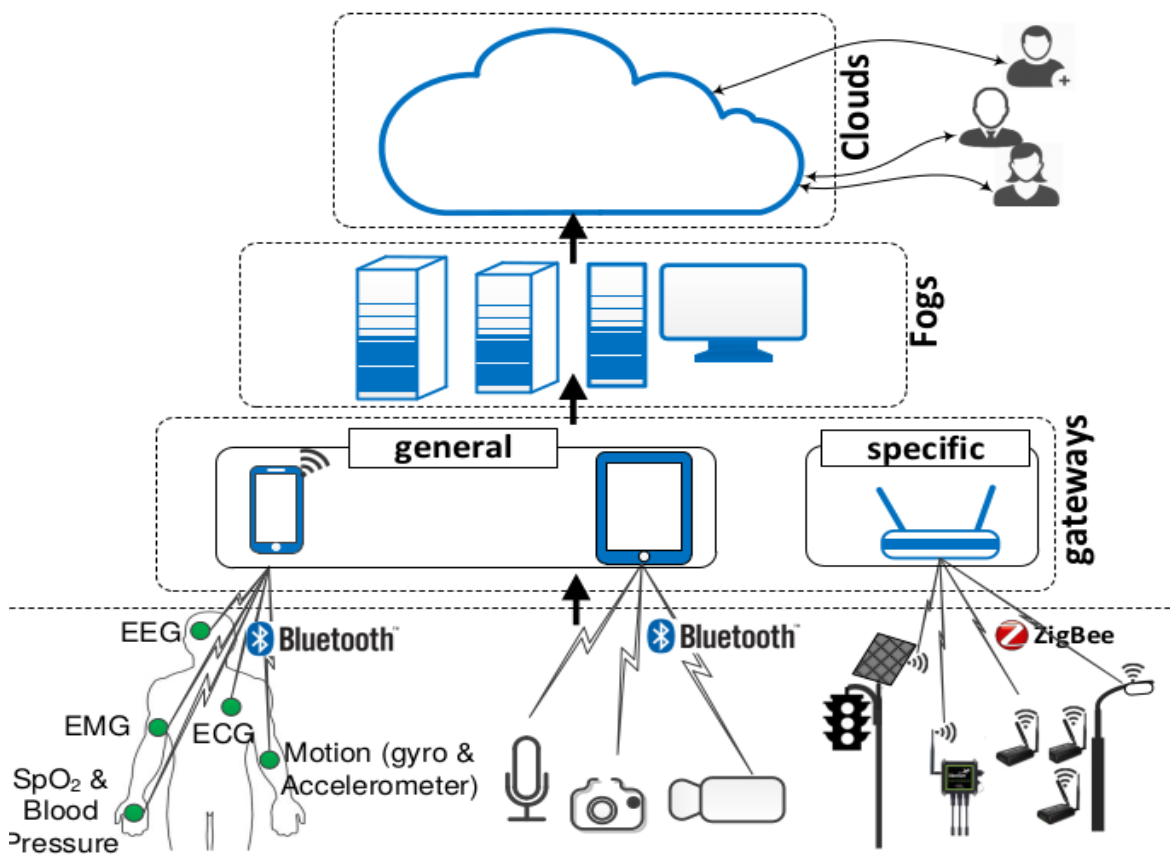


Figure 1: Architecture IdO

Src : [Computation Offloading and Resource Allocation for Low-power IoT Edge Devices](#)

Une fois les données numérisées et agrégées, elles entrent dans la troisième étape pour davantage de traitement. Les systèmes informatiques effectuent ce traitement grâce à une analyse détaillée. Ces derniers, plus généralement, des systèmes informatiques périphériques (Applications mobiles, Web,...) peuvent être situés hors site de déploiement ou sur site (connexion sur un serveur à distance).

C'est à ce stade que les données sont stockées dans des centres de données et dans le cloud. Les données ne nécessitent pas de retour d'information immédiat et nécessitent un traitement plus détaillé; elles sont transférées vers des centres de données dotés de systèmes informatiques plus avancés pour effectuer des analyses approfondies. Ce type de traitement a lieu sur le site.

## I. 3 Les Applications de l'IdO

Plusieurs domaines d'application sont touchés par l'internet des objets. Parmi ces derniers nous pouvons en citer:

- **Smart Homes:** Smart Homes est la fonctionnalité la plus tendance de l'IdO. Ils veulent que leurs maisons soient converties en maisons intelligentes afin de mener une vie plus confortable et plus commode. Les produits Smart Home sont conçus pour économiser temps, argent et énergie. Les maisons intelligentes deviendront bientôt une caractéristique commune, tout comme les smartphones.
- **Gadgets portables:** Il existe une forte demande d'équipements connectés sur le marché. Des capteurs et des logiciels installés dans ces dispositifs IdO portables rassemblent des informations précieuses sur l'utilisateur et leur traitement génère des informations utiles pour l'utilisateur. Ces appareils sont principalement destinés à la santé, au fitness et au divertissement.
- **Voitures connectées:** Ce type de voitures peut fonctionner et être entretenu par ses propres moyens grâce à des capteurs et à la connectivité Internet pour le confort des passagers. Les grandes marques y travaillent pour apporter une nouvelle révolution aux systèmes de véhicules.
- **Industries:** Internet industriel est un sujet brûlant dans le monde industriel. Il vise à doter les industries de capteurs, de logiciels et d'analyses afin de fabriquer des machines plus avancées et plus brillantes. Les principaux avantages seront le contrôle de la qualité, la durabilité, le suivi des marchandises et l'échange d'informations en temps réel.
- **Villes intelligentes ~ Smart Cities:** Depuis les maisons intelligentes, les applications de l'IdO s'étendent maintenant aux villes intelligentes. Est-ce que toutes les fonctionnalités y seront incluses? Surveillance intelligente, gestion automatisée des



transports, gestion de l'énergie, distribution de l'eau, surveillance de la sécurité et de l'environnement. L'Internet des objets s'engage à résoudre les problèmes auxquels les habitants des villes sont souvent confrontés, comme le trafic, la pollution, etc.

- **Agriculture:** La demande en approvisionnement alimentaire augmente en raison de l'augmentation de la population mondiale. L'IdO a tendance à développer certaines techniques dans le domaine de l'agriculture pour augmenter la production alimentaire. En outre, les agriculteurs peuvent également obtenir des informations utiles concernant les besoins en sol et en humidité, etc.
- **Energie:** Le concept de réseau intelligent attire l'attention dans le monde entier. Il vise à améliorer l'efficacité de l'électricité et à mesurer la consommation d'électricité des consommateurs.
- **Santé:** Les systèmes de santé intelligents seront en mesure de collecter des informations sur la santé d'un individu. Il vise à offrir une vie plus saine aux patients.

## I. 4 Avantages et Inconvénients de l'Internet des Objets

L'avènement de l'IdO présente des avantages ainsi que des inconvénients. Nous pouvons déceler certains avantages comme

- Communication, Automatisation et contrôle - L'IdO permet une communication de meilleure qualité entre les périphériques, appelée communication de machine à machine (M2M). Les machines communiquent les unes avec les autres sans intervention humaine, donnant ainsi plus d'automatisation et de contrôle. Cela fournira une sortie rapide et un temps de réponse plus rapide. De plus, il y a plus de transparence entre les appareils.

- Information - Avec l'IdO, nous pouvons collecter des informations précieuses à travers des capteurs et des actionneurs de l'environnement. Il est également efficace, ce qui permet d'obtenir des résultats plus précis. Cela fera gagner du temps et permettra aux gens de faire un autre travail créatif.
- Meilleure qualité de vie - Ces appareils sont conçus pour offrir une meilleure qualité de vie dans le confort et la commodité des personnes. Le plus grand avantage des objets connectés est l'économie de temps et d'argent. Cela profitera aux personnes dans leur vie quotidienne, car les appareils peuvent communiquer entre eux par eux-mêmes.

Bien qu'il vise à offrir un confort dans la vie, il convient de prendre en compte certains inconvénients de l'IdO comme:

- Compatibilité - Étant donné que les périphériques du réseau appartiennent à des fabricants différents, des problèmes de compatibilité peuvent se poser, par exemple si le périphérique est capable de se connecter ou non. Il n'y a pas non plus de normes internationales définies pour l'interconnexion de périphériques. Ainsi L'IdO est un système complexe, les chances d'échec sont donc plus grandes. De plus, il peut y avoir un bug ou une erreur dans la fonctionnalité. Ce problème peut être surmonté en concevant certaines normes.
- Problèmes de sécurité / confidentialité - Les problèmes de sécurité et de confidentialité persistent pendant le transfert des données d'un utilisateur à un autre. Des informations peuvent être volées et mal utilisées.
- Vie dépendante de la technologie et moins d'emploi - Nous deviendrons esclaves de la technologie. Même la jeune génération est plus accro à cette technologie. Nous serons dépendants de la technologie pour tous. Avec cet avènement, de plus en plus de personnes vont perdre leur emploi car tout sera automatisé. Cela entraînera un taux de chômage élevé dans la société.

## II. Edge Computing

Les récentes avancées technologiques en matière de capteurs, de communication sans fil et de processeurs intégrés ont permis la conception d'équipements de petite taille, à faible consommation et à faible coût, pouvant être mis en réseau ou connectés à Internet. Ce sont les composants clés du paradigme émergent de l'Internet des objets (IdO) [4].

Dans un contexte très propice aux déploiements de l'IdO, la croissance des terminaux ou des réseaux de capteurs a un effet très clair : la hausse du volume de données propres à l'IdO. L'un des défis de l'Internet des Objets est de traiter et d'analyser une énorme quantité de données provenant de périphériques hétérogènes [4].

Jusqu'à récemment, les entreprises se tournaient vers le cloud pour traiter et analyser d'énormes quantités de données. Mais en ce qui concerne l'IdO, le cloud ne suffit pas. La gestion de toutes ces données collectées avec des serveurs cloud centraux est inefficace et même parfois irréalisable pour les raisons suivantes:

- la limitation de la capacité d'exécution, de la communication et de stockage ;
- l'énergie globale et le coût ;
- latence non fiable [4].

C'est là qu'entre en jeu l'Edge Computing, on parle également d'informatique de proximité ou périphérie. En traitant et en analysant des données aussi proches que possible de la source d'origine, l'Edge, associé aux objets connectés, devient la nouvelle norme.

### II. 1 Edge Computing: Définition

Traduit en français par "traitement des données à la périphérie du réseau", l'Edge Computing désigne une architecture de technologie d'information distribuée qui se caractérise par une puissance de traitement décentralisée. Concrètement, il permet de traiter des données de façon directe par le périphérique qui les produit (ou par un ordinateur local). Il peut dans ce cas ne pas être nécessaire de transmettre les données à un Datacenter distant pour les analyser. L'IDC (International Data Corporation) définit de son côté l'Edge Computing comme un réseau maillé de micro Datacenter capables de traiter ou de stocker des données localement [2]. On retrouve cette technologie principalement dans le domaine de l'IdO, où elle vient combler certaines contraintes sur le Cloud Computing.

Le volume de données générées par les objets connectés et autres appareils mobiles (smartphones et tablettes) est en pleine explosion. Dans ce contexte, des spécialistes comme Helder Antunes de Cisco estiment qu'il est irréaliste de penser que toutes ces données pourront être transmises entre leurs sources et les Data Center (centres de données) Cloud de façon stable et rapide afin d'être analysées [2]. Or, les entreprises de nombreuses industries comme la manufacture, la santé, les télécommunications et la finance ont besoin d'être en mesure d'analyser les données les plus importantes le plus rapidement possible, presque en même temps qu'elles sont collectées. Face à cette problématique d'avenir, l'Edge Computing pourrait être la solution.

Près de 82% des entreprises de la zone EMEA utiliseront l'edge computing en 2020, selon l'étude Data Age 2025 réalisée par le cabinet IDC en décembre 2018 [5]. Cette technologie dite "à la frontière du réseau" consiste à intégrer de l'intelligence artificielle dans un microcontrôleur positionné sur une machine ou à proximité afin d'en analyser l'état de fonctionnement en temps réel sans passer par le cloud. "L'usage de l'edge computing se développe depuis quelques années déjà avec le besoin de traiter les gros volumes de données transmises par l'IdO", explique Sadissa Babeni, responsable marketing chez l'éditeur canadien Ormuco [5].

## II. 2 Edge Computing : Utilité

Pour faire simple, le Edge Computing ou traitement des données à la périphérie du réseau est une architecture informatique distribuée ouverte. Plus précisément, selon la définition d'IDC, le Edge Computing peut être considéré comme un réseau maillé de Micro Datacenter qui traitent ou stockent les données critiques localement. Les données sont ensuite transmises vers un Data Center central ou un stockage cloud.

Le plus souvent, l'Edge Computing est utilisé dans le domaine de l'internet des objets. Une partie des nombreuses données collectées par les appareils connectés sont traitées localement, afin de réduire le trafic en direction du Cloud ou des Data Center et de permettre une analyse des données importantes en temps réel (ou presque) (figure 2).

L'objet connecté transfère généralement les données vers un appareil local de petite taille capable d'effectuer le traitement et le stockage. Les données sont ainsi traitées en périphérie du réseau avant d'être transmises vers le Cloud ou le centre de données.

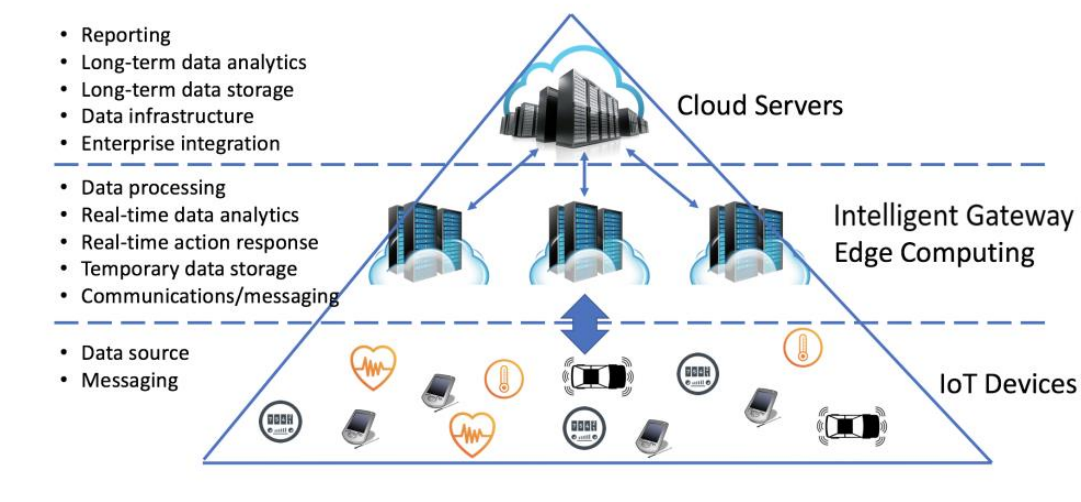


Figure 2: les différentes couches de traitement et leurs capacités

Src : [A Survey on the Edge Computing for the Internet of Things](#)

## II. 3 Edge Computing vs Cloud Computing

Faisant une comparaison entre edge computing et cloud computing, chaque technologie met en avant ses points forts pour apparaître comme la meilleure solution de traitement et de stockage des données. L'edge computing est réputé pour réduire le temps de latence du traitement de l'information. Les flux de données sont ainsi traités en temps réel. On attribue également à l'edge computing un meilleur niveau de sécurité puisque les données demeurent stockées à proximité de leur source et n'ont plus à traverser un réseau pour rejoindre un Datacenter public. Sur ce sujet, les partisans du cloud computing pointent la vulnérabilité des appareils en périphérie de réseau. Ils vantent de leur côté le niveau de sécurité extrêmement élevé des environnements cloud. Côté finance, l'edge computing permet bien souvent de réaliser des gains économiques importants, un autre atout qui justifie l'intérêt grandissant des entreprises pour ce type d'architecture. Le *tableau 1* permet de faire une comparaison de manière générale et suivant plusieurs paramètres du cloud et de l'edge computing

	Edge	Cloud
Déploiement	Distribué	Centralisé
Composants	Nœuds périphériques	Ressources virtuelles
Capacité d'exécution	Limité	Illimité
Storage	Limité	Illimité
Temps de réponse	rapide	lent
Big Data	Process	Process

*Tableau 1: Caractéristique de l'edge computing et du Cloud computing*

Src : [A Survey on the Edge Computing for the Internet of Things](#)

### II. 3. 1 L'Edge Computing est-il plus sécurisé que le Cloud Computing ?

En un sens, on peut considérer que la sécurité est plus élevée en périphérie du réseau, car les données ne traversent pas le réseau et restent à proximité de leur source. De fait, en cas de compromission d'un environnement Cloud ou d'un Data Center interne, le nombre de données vulnérables est minimisé.

Si l'edge computing suscite autant d'intérêt, c'est qu'il offre de nombreux avantages aux industriels en termes de sécurité. En analysant les données dans le microcontrôleur et en envoyant sur le cloud uniquement des messages d'alerte de dysfonctionnement, la technologie de edge computing évite un partage de données de production. Il est par ailleurs plus aisé pour les éditeurs de crypter un message qu'un flux de données. Cette sécurisation est l'une des principales préoccupations des industriels dans un projet de maintenance prédictive, selon une étude du cabinet de conseil BearingPoint réalisée en 2017 [5].

Cependant, les équipements en périphérie du réseau sont à priori plus vulnérables que le Cloud ou les Datacenter. Par conséquent, un environnement Edge Computing peut en réalité être moins sécurisé qu'un environnement Cloud ou un Data Center interne. De ce fait, afin d'assurer la sécurité d'un environnement Edge Computing, un soin tout particulier doit être apporté à la sécurité : chiffrement des données, contrôle des accès, utilisation d'un réseau privé virtuel (Virtual private network VPN) sont des mesures indispensables.

### II. 4 Quels sont les avantages et les cas d'usage de l'Edge Computing ?

L'Edge Computing peut s'avérer pertinent dans de nombreuses situations. Par exemple, lorsque les objets connectés IoT ont une faible connectivité, il n'est pas efficace de les laisser constamment connectés à un Cloud central. Le traitement à la périphérie permet de remédier à ce problème.

De même, il permet de réduire la latence du traitement de l'information, car les données n'ont pas besoin de traverser un vaste réseau pour atteindre un Data Center ou un serveur cloud distant. Cette réduction de la latence est particulièrement importante dans les domaines des services financiers ou de la fabrication.

On peut citer divers problèmes auxquels l'Edge Computing apporte des solutions:

- **La Latence:**

D'une manière générale, la latence d'une application est le produit de deux composants: le temps d'exécution et celui de transmission. Le temps de calcul indique le temps passé au traitement des données, qui dépend de la capacité de calcul du système. Il est clair que les capteurs sont souvent des périphériques intégrés avec une capacité de calcul limitée, alors que les serveurs de réseau auront une capacité importante pour fournir un traitement rapide des données. Néanmoins, la transmission de données entre les périphériques finaux et les serveurs cloud entraînera une augmentation significative du temps de transmission. Par conséquent, le défi pour l'informatique à la périphérie du réseau est de déterminer le compromis idéal entre temps de calcul et temps de transmission, ce qui nécessite une décentralisation optimale des tâches. Il serait bien de mettre en place un système permettant de voir si une tâche de traitement de données doit être effectuée localement, être exécutée sur les serveurs de périphérie / cloudlet ou encore sur les serveurs de cloud distant [6].

- **La Bande Passante:**

La plupart des équipements qui font tourner des logiciels et génèrent des données doivent être connectés au cloud pour stocker et traiter ces données. En tant que tel, il faut d'énormes quantités d'énergie et de bande passante pour transmettre des données IoT au cloud. Les données volumineuses consommeront une immense bande passante réseau et entraîneront un certain nombre de problèmes, tels que le délai de transmission et la perte de paquets.

L'Edge Computing permet justement de réduire cette utilisation de la bande passante, car les données peuvent être traitées à proximité de la source.



Prenons l'exemple de caméras embarquées de la police. Elles peuvent réduire la bande passante en analysant les flux vidéo des caméras de contrôle, n'envoyant que les données pertinentes. Celles-ci génèrent d'immenses volumes d'enregistrements vidéo et audio, en temps réel.

- **La Sécurité liée à l'IdO**

La connexion de tous les appareils au cloud et l'envoi de données brutes via Internet peuvent avoir des conséquences sur la confidentialité et la sécurité, surtout sur les données sensibles soumises à la réglementation de différents pays. En traitant les données à proximité de la source, les entreprises peuvent conserver les données à l'intérieur de leurs frontières et garantir le respect des lois sur la souveraineté

Avec cette notion d'Edge dans l'IdO, les données peuvent être stockées dans des Datacenter locaux, garantissant ainsi le contrôle et la limitation de l'accès.

### III. Intelligence Artificielle

L'Intelligence Artificielle, terme créé par John McCarthy peut être défini comme étant la construction de programmes informatiques s'adonnant à des tâches qui sont pour l'instant accomplies de façon plus satisfaisantes par les êtres humains car elles demandent des processus mentaux de hauts niveaux tels que l'apprentissage perceptuel, l'organisation de la mémoire et le raisonnement critique. En d'autres termes, l'IA est un ensemble de techniques permettant à des machines d'accomplir des tâches et de résoudre des problèmes normalement réservés aux humains [7]. Tâches que ces derniers arrivent à faire de manière simple à l'exemple de reconnaître et localiser les objets dans une image. Par contre ce n'est pas le cas pour les machines car ces tâches sont assez complexes et requiert certains notions.

Depuis quelques années, on associe presque toujours l'intelligence aux capacités d'apprentissage. C'est grâce à l'apprentissage qu'un système intelligent capable d'exécuter une tâche peut améliorer ses performances avec l'expérience. C'est grâce à l'apprentissage qu'il pourra apprendre à exécuter de nouvelles tâches et acquérir de nouvelles compétences. Le machine Learning et le deep Learning sont des domaines de l'IA [7].

Ainsi les méthodes manuelles pour mettre en place des programmes pour des tâches apparemment faciles sont très difficiles. A l'exemple pour une machine, une image est un tableau de nombre représentant les couleurs en pixel, et les signaux sonores des suites de nombre indiquant la pression de l'air à chaque instant. La question que l'on se pose est: comment la machine arrive-t-elle à transcrire les signaux à l'égard des bruits ? Ou à pouvoir reconnaître une chaise dans la suite de nombres de l'image lorsque celle-ci varie constamment? Il est impossible d'écrire un programme robuste pour ces situations. C'est là qu'intervient le machine Learning, domaine que nous étudierons tout au long de cette partie : son principe, ses méthodes et composants et ses controverses.

Le machine Learning est utilisé par les grands systèmes d'Internet qui y ont recours depuis longtemps pour filtrer les contenus indésirables, ordonner les réponses à une recherche, sélectionner les informations intéressantes pour chaque utilisateur.

On peut en citer la voiture autonome de Google, la classification des mails dans Gmail, la traduction en temps réel de Skype, la reconnaissance vocale Siri d'Apple, la détection de fraude dans le monde de la banque, la reconnaissance faciale, les moteurs de recherche, l'aide au diagnostic médical notamment la bio-informatique, l'analyse financière dont l'analyse du marché boursier, la classification des séquences d'ADN, les jeux, génie logiciel, adaptation de sites Web, locomotion de robots, analyse prédictive en matière juridique et judiciaire...

### III. 1 Machine Learning (Apprentissage Automatique AA)

Le Machine Learning est un sous domaine d'Intelligence Artificiel qui confère aux ordinateurs la capacité d'apprendre sans être explicitement programmés et consiste à mettre en place des algorithmes ayant pour objectif d'obtenir une analyse prédictive à partir de données dans un but précis [8].

L'apprentissage automatique est une branche de l'intelligence artificielle qui donne aux systèmes la capacité d'apprendre automatiquement et de s'améliorer à partir de l'expérience sans être explicitement programmé ou sans l'intervention humaine. Son objectif principal est de faire en sorte que les ordinateurs apprennent automatiquement à partir de cette expérience. En d'autres termes, le ML vise à créer des modèles basés sur des observations et des expériences, puis à utiliser le modèle pour prédire des données futures ou découvrir des modèles.

Un système entraînable peut être vu comme une boîte noire avec une entrée, par exemple une image, un son, ou un texte, et une sortie qui peut représenter la catégorie de l'objet dans l'image, le mot prononcé, ou le sujet dont parle le texte. On parle alors de systèmes de classification ou de reconnaissance des formes.

Alors, que faut-il pour créer de tels systèmes d'apprentissage automatique? Voici les éléments nécessaires à la création de tels systèmes d'apprentissage automatique:

- Données : Les données d'entrée sont nécessaires pour prédire la sortie.
- Algorithmes : L'apprentissage automatique dépend de certains algorithmes statistiques pour déterminer les modèles de données.
- Automatisation : C'est la capacité de faire fonctionner les systèmes automatiquement.
- Itération : Le processus complet est itératif, c'est-à-dire une répétition du processus.
- Evolutivité : La capacité de la machine peut être augmentées ou réduites.
- Modélisation : Les modèles sont créés en fonction de la demande par le processus de modélisation.

### III. 2 Qualité d'apprentissage et Étapes d'un projet ML

La qualité et l'efficacité d'un apprentissage dépendent de plusieurs paramètres notamment de la compétence de l'opérateur à préparer l'analyse, de la complexité du modèle de même que du mode des résultats pour l'utilisateur final.

Le projet de Machine Learning évolue suivant plusieurs étapes différentes des algorithmes. Passant par l'acquisition des données qui seront préparées et nettoyées afin d'éliminer toute sorte de bruit. Ceci s'ensuit d'une création de modèle et de son évaluation là où interviennent les algorithmes pour aboutir au déploiement qui mettra le projet en production.

L'apprentissage automatique utilise des processus similaires à ceux de l'exploration de données. Les algorithmes d'apprentissage automatique sont décrits en termes de fonction cible (f) qui mappe la variable d'entrée (x) à une variable de sortie (y). Ceci peut être représenté comme:

$$Y = f(x) \quad (1)$$

Il y a aussi une erreur qui est indépendante de la variable d'entrée x. Ainsi, la forme plus généralisée de l'équation est la suivante:

$$Y = f(x) + e \quad (2)$$

Dans la machine, le mappage de  $x$  à  $y$  est effectué pour les prédictions. Cette méthode est connue sous le nom de modélisation prédictive pour faire des prédictions plus précises. Il existe différentes hypothèses pour cette fonction.

### III. 3 Méthodes d'Apprentissage

Les algorithmes de Machine Learning utilisent une phase d'apprentissage. Ces programmes détectent des schémas dans les données et ajustent leur fonctionnement en conséquence. Ceci permettra la création d'un modèle efficace pour les données à prédire. Ainsi il y a différents types d'apprentissage. Elles sont classées dans certaines catégories. Ceux-ci sont:

- Apprentissage Supervisé: utilise des ensembles de données de formation étiquetés avec la sortie correcte
- Apprentissage Non Supervisé: utilise une collection de données non étiquetées pour rechercher des modèles sous-jacents
- Apprentissage par renforcement: le résultat correct n'est pas disponible a priori, mais de manière empirique, le résultat prédit peut être évalué par une récompense positive ou négative qui indique si le résultat prédit était bon ou mauvais.
- Apprentissage Semi-Supervisé: la règle de décision est induite sur la base d'exemples étiquetés et non-étiquetés.

L'apprentissage automatique peut être appliqué à différents types de données, tels des graphes, des arbres, des courbes, ou plus simplement des vecteurs de caractéristiques, qui peuvent être continues ou discrètes.

#### III. 3. 1 Apprentissage Supervisé

Dans ce type d'emploi, les étiquettes (résultat attendu pour un ensemble donné en entrée) sont prédéterminées et les exemples connus, le système apprend à classer selon un modèle de classement. Le processus se passe en deux phases. Lors de la première phase (dite d'apprentissage), il s'agit de déterminer un modèle des données étiquetées. La seconde phase (dite de test) consiste à prédire l'étiquette d'une nouvelle donnée, connaissant le modèle préalablement appris [8]. Parfois il est préférable d'associer une donnée non pas à une classe unique, mais une probabilité d'appartenance à chacune des classes prédéterminées (on parle alors d'apprentissage supervisé probabiliste).

On parle de classification ou de classement si les étiquettes sont discrètes, ou de régression si elles sont continues. Ainsi il y'a divers approches avec l'apprentissage supervisée

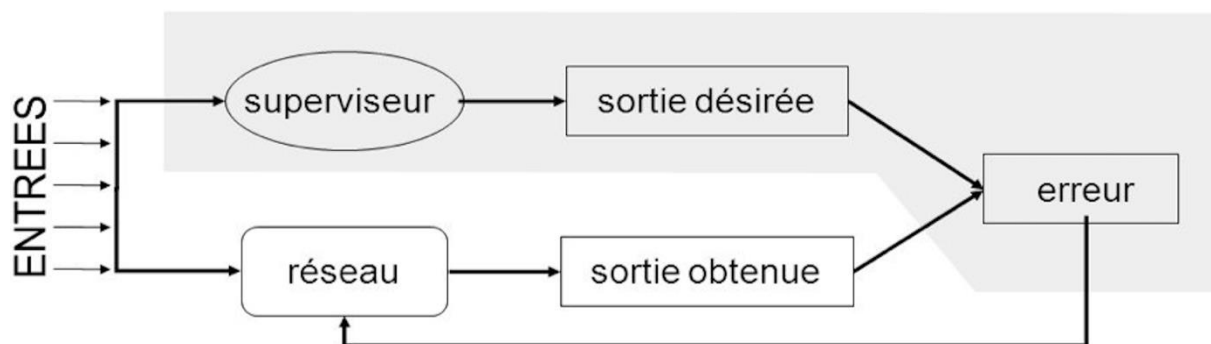


Figure 3: Apprentissage supervisé

Src : <https://images.app.goo.gl/nVVJMXjDULMsL7Ra6>

Les algorithmes pouvant être utilisés sont :

La Règle de Bayes, la classification naïve bayésienne, la régression multi variée, la régression régularisée, les k plus proches voisins, Arbre de décision (Decision Tree DT), Forêt aléatoire (Random Forest RF), Perceptron Simple, Perceptron multicouche, Les réseaux de neurones, Deep Learning.

### III. 3. 2 Apprentissage Non-Supervisé

Il est utilisé quand le système ou l'opérateur ne disposent que d'exemples, mais non d'étiquettes, et que le nombre de classes et leur nature n'ont pas été prédéterminés. L'algorithme

doit découvrir par lui-même la structure plus ou moins cachée des données. Le partitionnement de données, data clustering en anglais, est un algorithme d'apprentissage non supervisé.

Le système doit dans l'espace de description cibler les données selon leurs attributs disponibles, pour les classer en groupe homogènes d'exemples. La similarité est généralement calculée selon une fonction de distance entre paires d'exemples. C'est ensuite à l'opérateur d'associer ou déduire du sens sur chaque groupe et pour les motifs (patterns en anglais) d'apparition de groupes dans leur "espace". Divers outils mathématiques et logiciels peuvent l'aider. On parle aussi d'analyse des données en régression (ajustement d'un modèle par une procédure de type moindres carrés ou autre optimisation d'une fonction de coût). Si l'approche est probabiliste (c'est-à-dire que chaque exemple, au lieu d'être classé dans une seule classe, est caractérisé par un jeu de probabilités d'appartenance à chacune des classes), on parle alors de "soft clustering" (par opposition au "hard clustering").

Les algorithmes utilisés sont K-moyennes (K-means), les Cartes auto-organisatrices font partie des algorithmes auxquels on a recours pour faire de l'apprentissage non-supervisé.

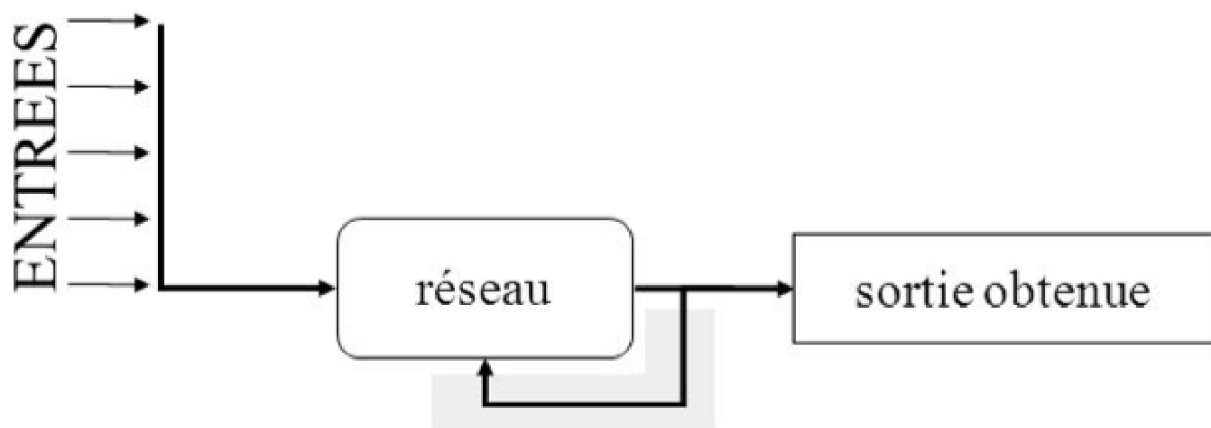


Figure 4: Apprentissage non supervisé

Src <https://images.app.goo.gl/HX2sabWE8bbUywwSA>

### III. 3. 3 Apprentissage Semi-Supervisé

Probabiliste ou non, quand l'étiquetage des données est partiel. C'est le cas quand un modèle énonce qu'une donnée n'appartient pas à une classe A, mais peut-être à une classe B ou C (A, B et C étant 3 maladies par exemple évoquées dans le cadre d'un diagnostic différentiel).

### III. 3. 4 Apprentissage par Renforcement

L'algorithme apprend un comportement étant donné une observation. L'action de l'algorithme sur l'environnement produit une valeur de retour qui guide l'algorithme d'apprentissage.

L'algorithme de Q-learning est un exemple classique.

### III. 3. 5 Apprentissage par Transfert

L'apprentissage par transfert peut être vu comme la capacité d'un système à reconnaître et appliquer des connaissances et des compétences, apprises à partir de tâches antérieures, sur de nouvelles tâches ou domaines partageant des similitudes. La question qui se pose est:

Comment identifier les similitudes entre la ou les tâche(s) cible(s) et la ou les tâche(s) source(s), puis comment transférer la connaissance de la ou des tâche(s) source(s) vers la ou les tâche(s) cible(s) ?

## III. 4 Background Machine Learning

L'utilisation de ML peut être divisée en quatre catégories principales:

### III. 4. 1 Classification

La sortie de la classification appartient à un ensemble fini de valeurs discrètes prédéfinies ou de classes. Selon le nombre de classes, les problèmes de classification tombent dans l'une de ces deux catégories:



- i) La classification binaire: il s'agit de deux étiquettes seulement, telles que False / True ou 0/1.
- ii) la classification multi-classes: elle implique plus de deux classes. Par exemple, surveiller le trafic de données d'un appareil IoT et détecter si elle participe ou non à une attaque DDoS est une classification binaire. Mais la reconnaissance de l'activité humaine au moyen de dispositifs portables (pour détecter «marcher», «courir», « être assis», «debout», etc.) est une classification à plusieurs classes.

Il existe plusieurs modèles de classification qui incluent mais sans s'y limiter, la régression logistique (LoR), la machine à vecteurs de support (SVM), l'arbre de décision (DT), Naive Bayes, les k-voisins les plus proches (KNN), les réseaux de neurones artificiels (ANN) et l'analyse discriminante linéaire (LDA). Notez que (LoR) malgré son nom, est un modèle de classification et non de régression. Les opérations internes de certains modèles de classification sont décrites schématiquement à la (figure 5).

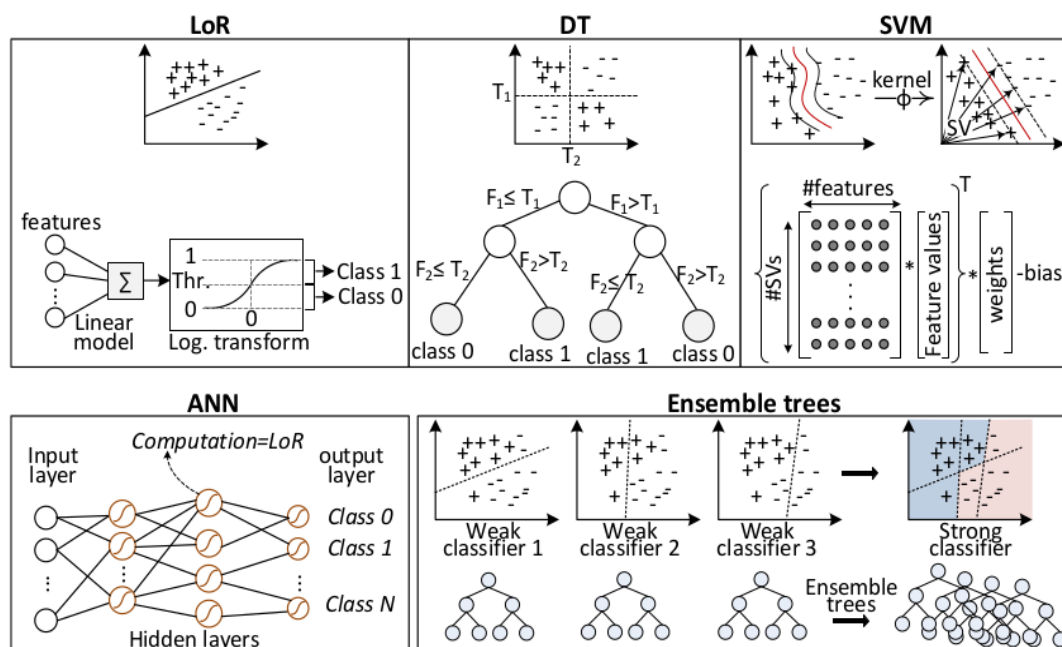


Figure 5: Opérations internes de quelques modèles de classification

Src : [9]

Les réseaux de neurones profonds (Deep Neural Networks DNN), contrairement à l'ANN, ne s'appuient pas sur les fonctionnalités sélectionnées par le concepteur. Au lieu de cela, ils apprennent les fonctionnalités automatiquement et directement à partir des données. Pour ce faire, DNN nécessite généralement davantage de couches cachées afin de détecter de manière hiérarchique les entités de différentes couches d'abstraction. Le nombre plus élevé de couches cachées rend ces réseaux «profonds». Les réseaux de neurones convolutifs (CNN) sont une classe de réseaux de neurones profonds qui constituent le principal modèle pour la classification d'images et la vision par ordinateur. CNN préserve la structure spatiale des sous-figures de manière hiérarchique et utilise des filtres de convolution pour détecter les cartes de caractéristiques. Dans les réseaux de neurones récurrents (RNN), la sortie de certains neurones peut être renvoyée et alimenter les entrées des neurones de la même couche ou des couches précédentes (voir la *figure 6*). Ces boucles de rétroaction dans la structure du réseau aident le RNN à conserver les informations de série chronologique et les modèles temporels. Les RNN fonctionnent très bien pour la reconnaissance vocale, la classification vidéo et d'autres applications dont les entrées ou les sorties sont des séquences.

Les modèles d'ensemble consistent en plusieurs classificateurs complémentaires qui peuvent même être faibles individuellement mais peuvent être combinés pour former un classificateur puissant et précis (voir la *figure 7*). Par exemple, Random Forest (RF) définit plusieurs DT chacun formés avec différents sous-ensembles de données indépendants. Les modèles d'ensemble sont plus largement utilisés pour la classification mais peuvent également être appliqués pour la régression. Certains modèles de classification tels que RF, ANN ou kNN sont intrinsèquement capables de traiter plusieurs classes. Quelques modèles tels que SVM et LoR doivent utiliser plusieurs classifications binaires afin de traiter un problème multi classe. Deux approches principales sont:

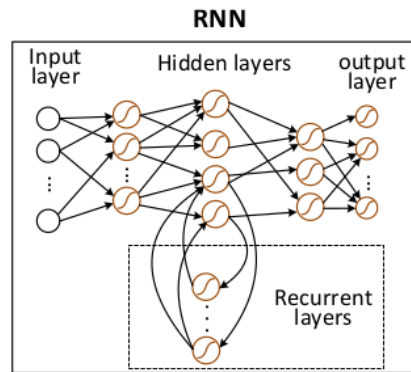


Figure 6: Réseaux de neurones récurrents

Src [9]

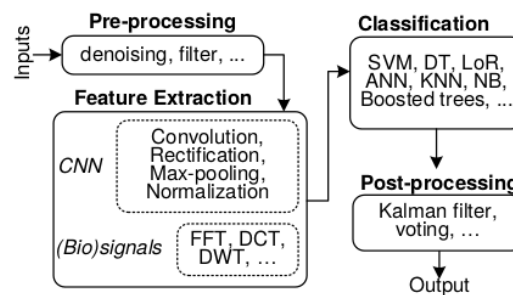


Figure 7: Différentes étapes de la tâche de classification et de l'exécution

Src : [9]

1) un contre tous (OvA One vs All): dans cette approche, un classificateur binaire est utilisé pour séparer chaque étiquette de toutes les autres étiquettes. Par conséquent, cette approche nécessite que le classificateur N gère un problème de classe N.

2) un contre un (OvO One vs One): dans cette approche, un classificateur binaire est utilisé pour séparer chaque paire de classes. Par conséquent, il en résulte  $\frac{N*(N-1)}{2}$  (3) classificateurs dans un problème de classe N. Le pipeline typique pour la classification de l'apprentissage automatique comprend plusieurs étapes, comme illustré à la figure 7 :

Prétraitement (filtrage, réduction du bruit, etc.), extraction des caractéristiques (transformation rapide de Fourier, transformation en ondelettes discrète, transformation en cosinus discrète, regroupement maximal, rectification, etc.). Convolution, normalisation, etc..., classification et enfin post-traitement pour lisser la sortie (par exemple, filtre de Kalman, vote à la majorité, etc.).

Chaque étape contient différentes opérations de calcul, chacune d'entre elles pouvant bénéficier d'une mise en œuvre matérielle efficace ou d'une optimisation logicielle.

### III. 4. 2 Régression

La régression modélise la relation entre les variables d'entrée et la sortie pour fournir un modèle prédictif. La sortie de la régression appartient à une plage continue. Par exemple, dans une application de réseau intelligent, un modèle de régression peut prédire la consommation d'énergie en fonction d'informations antérieures. La régression linéaire (LiR) est le modèle de régression le plus utilisé. P formule une équation linéaire des variables d'entrée (par exemple,

$$p = \sum f_i \cdot w_i \quad (4)$$

Où  $f_i$  est la valeur de la  $i$ -ème caractéristique,  $w_i$  le poids de cette caractéristique et  $p$  la valeur prédite).

Certains modèles tels que DT et KNN peuvent également être utilisés pour la régression même si leur usage courant est la classification. La régression polynomiale est un modèle similaire, à la différence que la puissance des caractéristiques (par exemple,  $f_i$ ) peut être supérieure à 1. Ce modèle présente un risque de surajustement qui rend le modèle très performant pour les données de test, mais généralisé pour les nouvelles données, le modèle de régression linéaire peut souffrir de multi colinéarité (c'est-à-dire une forte corrélation entre les caractéristiques). Cela entraîne une grande variance dans la valeur du poids des fonctionnalités lors de l'exécution de la phase d'entraînement à plusieurs reprises. Pour éviter ce problème, deux modèles de régression similaires sont généralement utilisés, par exemple: régression de la dorsale et régression de l'opérateur de sélection du rétrécissement le plus absolu (LASSO). Bien que ces modèles utilisent également des prédicteurs linéaires, ils limitent la valeur des pondérations des entités au cours de la phase de formation. Par conséquent, les poids importants et la variance entre eux sont évités. Un avantage supplémentaire de LASSO réside dans le fait qu'il

sélectionne les caractéristiques en supprimant les caractéristiques sans importance, car les légères pondérations deviennent 0. Par conséquent, la différence entre Ridge, LASSO et la régression linéaire correspond aux pondérations des entités dans leur modèle linéaire. Lorsque le nombre d'entités est très grand, la régression de LASSO est plus appropriée.

### III. 4. 3 Clustering

Le clustering est une méthode d'apprentissage non supervisée qui extrait la structure et les modèles cachés des données. Il regroupe les données en quelques groupes ayant des caractéristiques similaires et une structure commune et les points de données de différents clusters sont dissemblables. Les principales applications du clustering incluent le marketing (par exemple, découvrir des segments de clientèle pour cibler des clients similaires avec des publicités similaires), les systèmes de recommandation (par exemple, la vente au détail, la musique, la vidéo à la demande), la détection des valeurs aberrantes (par exemple, la détection de tendances frauduleuses dans les domaines de l'assurance, du réseau intelligent, etc...), gestion du contenu (par exemple, regroupement d'articles, de livres et d'autres contenus), etc...

K-Means est l'un des modèles les plus populaires pour cette tâche, qui regroupe les données d'entrée en  $k$  grappes. Compte tenu du nombre de grappes,  $k$ , l'algorithme k-means suit ces étapes:

- On sélectionne  $k$  centroïdes (par exemple, de manière aléatoire) pour les grappes.
- On trouve le centre de gravité le plus proche pour chaque point de données et on assigne les données à ce cluster. La métrique permettant de trouver le centroïde le plus proche est souvent la distance euclidienne.
- on met à jour les centroïdes en calculant la position moyenne des points de données dans les grappes.
- On répète les étapes 2 et 3 jusqu'à ce que les centroïdes convergent.

La complexité de k-Mean clustering est  $O(n)$ , mais l'inconvénient est qu'elle nécessite une connaissance préalable de  $k$ .

La classification hiérarchique est une autre technique qui offre plus de flexibilité pour le nombre de grappes créées. La classification hiérarchique suit plusieurs étapes:

- On commence avec N groupes (par exemple, un pour chaque point de données s'il est petit).
- On fusionne les deux clusters les plus proches.
- On met à jour la distance calculée entre les clusters.
- Puis on répète les étapes 2 et 3 jusqu'à ce qu'il n'y ait qu'un seul cluster.

Les figures 8-a et 8-b montrent des exemples de k-moyennes et de classification hiérarchique, respectivement.

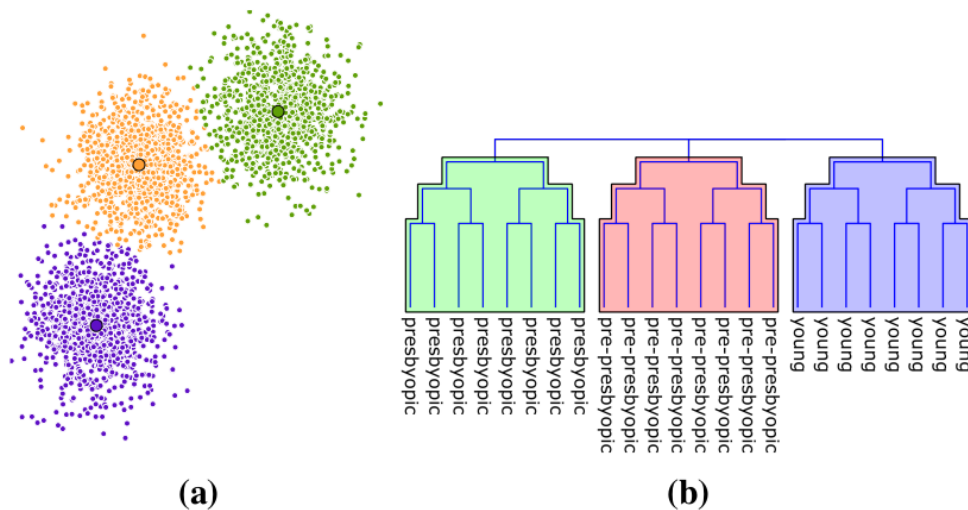


Figure 8: Exemples de (a) K-means Clustering avec  $K=3$ , (b) hierarchical Clustering

*Src : [9]*

### III. 4. 4 Apprentissage par renforcement

L'apprentissage par renforcement est principalement utilisé pour la prise de décision. Le résultat de chaque décision sera considéré comme la récompense ou la pénalité pour évaluer la décision et mettre à jour la fonction de décision. L'apprentissage par renforcement ne nécessite pas de formation hors ligne et convient donc aux systèmes IoT n'ayant aucune connaissance préalable de la relation entre leur entrée et leur sortie correcte (par exemple, une décision).

L'apprenant (appelé agent) apprend à partir d'expériences et d'essais et d'erreurs. Il prend des actions, observe la récompense de l'action plus le changement d'état l'environnement, puis améliore sa prise de décision en conséquence (voir Figure 9).

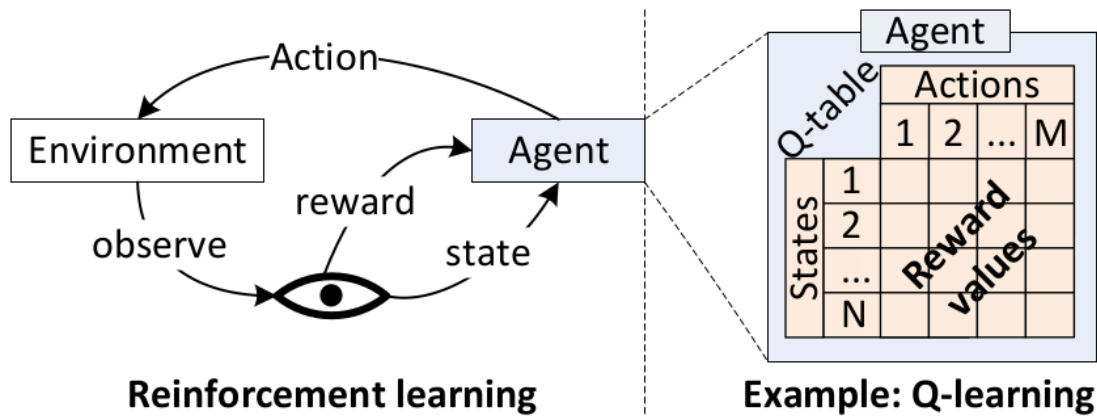


Figure 9: Comportement de l'apprentissage par Renforcement et de leurs interactions

*Src [9]*

Les problèmes d'apprentissage par renforcement peuvent être formalisés par le processus de décision de Markov décrit par:

- un ensemble fini d'états,
- un ensemble fini d'actions possibles dans chaque état,
- les transitions entre les états,
- les récompenses associées à chaque transition.

L'apprenant en renforcement a pour objectif de maximiser la somme des avantages à long terme:

$$\sum_{t=0}^{t \rightarrow \infty} \gamma^t \times r(s(t), a(t)) \quad (5)$$

Où  $r(s(t), a(t))$  désigne la récompense associée à une action  $a$  dans un état  $s$ . Le  $\gamma$  est un facteur de réduction compris entre 0 et 1 qui détermine l'importance des récompenses futures. Pour résoudre ce problème du processus de décision de Markov, nous devons trouver une

politique permettant de prendre les meilleures mesures dans chaque État, de manière à maximiser la récompense accumulée [9].

Q-Learning est l'algorithme d'apprentissage par renforcement le plus répandu, car il est facile à mettre en œuvre et sa convergence vers la solution optimale a été prouvée. Il crée une fonction de valeur d'action (appelée fonction Q) qui estime la récompense escomptée lorsque vous exécutez une action donnée tout en étant dans un certain état. La valeur initiale de la fonction Q de toutes les entrées est une valeur arbitraire fixe. À chaque horodatage, la fonction est mise à jour en fonction de l'action entreprise et de la récompense reçue.

$$Q(s_t, a_t) \leftarrow Q(s_{t-1}, a_{t-1}) + \alpha \times (r_t + \gamma \cdot \max_a Q(s_{t+1}, a) - Q(s_{t-1}, a_{t-1})) \quad (6)$$

où  $\alpha$ , le taux d'apprentissage indique le poids de la valeur apprise. Lorsque  $\alpha = 1$ , l'ancienne valeur est complètement remplacée avec la valeur apprise. La fonction Q de mise à jour est fondamentalement une moyenne mobile exponentielle des valeurs apprises. Notez que Q-Learning considère également la récompense possible pour les actions ultérieures. La fonction Q est généralement implémentée sous la forme d'une table (appelée Q-table) dont la taille est délimitée par le nombre d'états multiplié par le nombre maximal d'actions possibles pour un état (voir la *figure 9*).

L'apprentissage par renforcement, contrairement à d'autres modèles d'apprentissage automatique, doit faire l'objet d'un compromis entre exploitation et exploration au moment de l'exécution: qu'il s'agisse d'exploiter des actions déjà testées et efficaces, ou d'explorer de nouvelles actions et d'en découvrir d'autres plus efficaces. Par exemple, dans Q-Learning, nous pouvons choisir l'action avec la récompense attendue la plus élevée ou effectuer une action aléatoire un certain pourcentage du temps.



### III. 5 Avantages et Controverses de l'AA

#### III. 5. 1 Avantages de L'Apprentissage Automatique

- La prise de décision est plus rapide - L'apprentissage automatique fournit les meilleurs résultats possibles en hiérarchisant les processus de prise de décision de routine.
- Adaptabilité - L'apprentissage automatique offre la possibilité de s'adapter rapidement à un nouvel environnement en mutation. L'environnement change rapidement en raison du fait que les données sont constamment mises à jour.
- Innovation - L'apprentissage automatique utilise des algorithmes avancés qui améliorent la capacité globale de prise de décision. Cela aide à développer des services et des modèles d'entreprise innovants.
- Insight - L'apprentissage automatique permet de comprendre des modèles de données uniques et de prendre des mesures spécifiques.
- Croissance des entreprises - Grâce à l'apprentissage automatique, les processus métier et le flux de travail seront plus rapides, ce qui contribuera à la croissance et à l'accélération globale de l'entreprise.
- Le résultat sera bon - Grâce à l'apprentissage automatique, la qualité du résultat sera améliorée avec moins de risques d'erreur.

#### III. 5. 2 Controverses

- L'apprentissage automatique demande de grandes quantités de données pour fonctionner correctement. Il peut s'avérer difficile de contrôler l'intégrité des jeux de données, notamment dans le cas de données générées par les réseaux sociaux.
- La qualité des "décisions" prises par un algorithme d'apprentissage automatique dépend non seulement de la qualité (donc de leur homogénéité, fiabilité, etc.) des données

utilisées pour l'entraînement mais surtout de leur quantité. Donc, pour un jeu de données sociales collecté sans attention particulière à la représentation des minorités, l'AA est statistiquement injuste vis-à-vis de celles-ci. En effet, la capacité à prendre de “ bonnes ” décisions dépend de la taille des données, or celle-ci sera proportionnellement inférieure pour les minorités.

- L'apprentissage automatique ne distingue actuellement pas cause et corrélation de par sa construction mathématique, et est incapable d'aller au-delà du cadre imposé par ses données, il n'a donc pas de capacité d'extrapolation.

## Chapitre II : Etat de l'Art

### I. Machine Learning appliquée dans le domaine des IdO (Exécution des données du Domaine Agricole)

L'apprentissage automatique joue un rôle important en fournissant des services d'application aux utilisateurs. La plupart des applications IoT ont besoin de classifier les données d'entrées comme celles de sorties de leurs services. De nombreux domaines d'application bénéficient des avantages de l'apprentissage automatique, comme la santé, le smart home, l'industrie et les réseaux intelligents etc.

Dans le domaine de l'agriculture, diverses solutions IdO ont été proposées pour améliorer la productivité et ainsi réduire les coûts de maintenance des systèmes agricoles et de l'agriculture.

Dans [10], la classification d'images basée sur le réseau de neurones convolutifs est utilisée pour détecter des parasites et des maladies de plantes dans une ferme. Cette application d'Apprentissage Automatique à l'agriculture résout un problème particulier de diagnostic des maladies des cultures basé sur des images de plantes prises avec un smartphone. Deux informations sont importantes ici, l'incidence et la gravité de la maladie. Ils ont présenté un système de classification qui forme un modèle de classification à 5 classes pour déterminer l'état de maladie d'une plante. Les 5 classes représentent une classe de santé et 4 classes de maladies. Ils ont étendu en outre le modèle de classification pour classer différents niveaux de sévérité pour l'une des 4 maladies. Les niveaux de gravité sont attribués aux classes 1 à 5, 1 étant une plante en bonne santé, 5 étant une plante gravement malade. Ils présentent différentes manières d'extraire différentes caractéristiques des images de feuilles et montrent comment différentes méthodes d'extraction entraînent des performances différentes du classificateur. Ce système est basé sur un smartphone qui utilise le modèle de classification appris pour faire une prédiction en temps réel de l'état de santé d'un jardin de paysans. Cela consiste pour l'agriculteur à télécharger l'image d'une plante dans son jardin et à obtenir un score de maladie d'un serveur distant.

Dans une application similaire, réf. [11] utilise un classificateur linéaire de vecteur de support, des modèles kNN et ExtATree pour détecter les maladies des cultures à partir d'images de plantes. Les images peuvent être prises avec un smartphone, puis téléchargées sur un serveur sur lequel la classification est effectuée. L'application implique une classification multi-classe où quatre maladies et l'état de santé sont détectés.

En effet, les technologies de vision par ordinateur ont considérablement amélioré l'efficacité de la reconnaissance et du contrôle des parasites agricoles. Toutefois, la densité des caméras déployées sur les terres agricoles est généralement faible et les images ou vidéos de parasites agricoles collectés sont souvent obscures. Cela se traduit toujours par une faible résolution des parasites sur les images, ce qui les rend difficiles à observer et à surveiller. De plus, le procédé de détection d'objet existant n'est pas satisfaisant pour la détection de petites cibles avec une faible résolution en pixels. Par conséquent, il est nécessaire de restaurer et de sur-échantillonner les images collectées afin d'améliorer le taux de rappel de la détection. Ce travail a proposé un nouveau modèle de super-résolution basé sur le réseau résiduel récurrent profond. Comparée aux méthodes d'interpolation traditionnelles et aux modèles avec réseaux de neurones convolutifs peu profonds, la méthode proposée est plus puissante pour la reconstruction d'image et permet d'obtenir des performances de pointe. Les résultats expérimentaux montrent que la méthode a considérablement amélioré le taux de rappel de la détection des ravageurs de 202,06%. En outre, par rapport aux méthodes d'agrandissement d'image, telles que l'interpolation bi cubique et le réseau de neurones convolutionnels à super-résolution (SRCNN), cette méthode a été améliorée de 111,31% en moyenne et de 41,89% respectivement. Le modèle qui a été proposé pourrait réduire la densité de la disposition des caméras des systèmes de surveillance de l'Internet des objets agricole (IdO) et réduire le coût des infrastructures, qui a une grande valeur pratique.

Dans [12] Patil et Thorat surveillent les paramètres environnementaux d'un plant de raisin, notamment la température, l'humidité, la moiteur des feuilles, etc. Les données saisies sont transmises à un serveur où une application identifie les risques de maladies du raisin à un stade précoce et en informe le producteur.

La culture de la vigne a une importance sociale et économique en Inde. Au cours des dernières années, la qualité du raisin s'est dégradée pour de nombreuses raisons. Les maladies

des raisins sont l'une des principales causes. Pour prévenir les maladies, les agriculteurs pulvérisent une quantité énorme de pesticides, ce qui augmente les coûts de production. En outre, les agriculteurs sont incapables d'identifier les maladies manuellement. Les maladies ne sont identifiées qu'après l'infection, mais cela prend beaucoup de temps et a des effets néfastes sur le vignoble. Le travail effectué consiste à développer un système de surveillance qui identifiera les risques de maladies du raisin à un stade précoce en utilisant le modèle de Markov caché qui fournit des alertes par SMS au producteur et à l'expert. Le système comprend un capteur de température, d'humidité relative, d'humidité, d'humidité des feuilles et Zigbee pour la transmission de données sans fil.

Dans [13], CNN est utilisé pour détecter les fleurs et les gousses à partir des images prises et envoyées depuis la plante de la ferme afin de surveiller le taux de croissance et les paramètres environnementaux.

Récemment, une nouvelle approche TIC de l'agriculture, appelée «agriculture intelligente», a été considérée avec beaucoup d'attention pour aider les agriculteurs à prendre des décisions afin d'obtenir un bon rendement final dans divers types de conditions de terrain. À cette fin, cet article présente deux méthodes de détection d'images qui permettent une observation automatique pour capturer des fleurs et des gousses de soja dans de vrais champs. Les méthodes de détection d'images développées sont considérées comme des capteurs dans un système cyber-physique agricole dans lequel sont analysées des données volumineuses sur l'état de croissance des plantes agricoles et des informations environnementales (par exemple, météo, température, humidité, rayonnement solaire, état du sol, etc.) à exploiter des règles utiles pour une culture appropriée. Les méthodes de détection d'image proposées sont construites en combinant plusieurs techniques de traitement d'image et d'apprentissage automatique. La détection des fleurs est réalisée selon une approche grossière à fine dans laquelle les zones de fleurs candidates sont d'abord détectées par le SLIC et les informations de teinte, et l'acceptation des fleurs est décidée par CNN. Lors de la détection de germe, les candidats des régions de germe sont d'abord détectés par la méthode de détection d'objet de Viola-Jones. Les auteurs utilisent également CNN pour prendre une décision finale concernant l'acceptation des gousses détectées. La performance des méthodes de détection d'images proposées est évaluée pour un

ensemble de données d'images de soja prises à partir d'une foule de soja dans de véritables champs agricoles à Hokkaido, au Japon.

En résumé, les objectifs de ces solutions peuvent être classés en deux objectifs principaux:

La détection des maladies et des ravageurs [10], [11], [12] et le maintien de l'environnement requis pour les plantes, comme la température, l'humidité, l'état du sol, etc. [13].

Des techniques d'apprentissage automatique ont été utilisées pour évaluer l'environnement ou les installations à l'aide de moyens de surveillance tels que des images et vidéos [10], [11], ou de capteurs environnementaux tels que la température, l'humidité, etc. [12]. L'un des inconvénients de ces solutions existantes est la dépendance aux serveurs cloud. La complexité du traitement des données dans ces applications nécessite un matériel efficace et puissant pour activer ML sur des périphériques IdO intégrés.

## II. Etude sur les différentes méthodes de prédiction de l'Humidité du sol

Dans une logique d'appliquer notre solution dans un domaine bien déterminé. Nous nous sommes orientés dans le domaine de l'Agriculture. Ce choix a été motivé de par le fait qu'il soit une activité de secteur primaire au niveau de notre localité. De plus, ce domaine fait partie des sources principales de revenus alimentaires.

Afin de venir en aide à l'optimisation des ressources dont font usage les cultivateurs pour une bonne gestion de l'irrigation, notre étude se fait sur la prédiction de l'Humidité du sol d'un champ donné.

L'humidité du sol est l'un des principaux facteurs de la production agricole et des cycles hydrologiques, et sa prédiction précise est importante pour l'utilisation rationnelle et la gestion de l'eau. Cependant, l'humidité du sol implique des caractéristiques structurelles et des facteurs météorologiques complexes et il est difficile d'établir un modèle mathématique idéal pour la

prévision de l'humidité du sol. Les modèles de prédiction existants ont des problèmes tels que la précision de la prédiction, la généralisation et la capacité de traitement multifonctionnel, et les performances de prédiction doivent s'améliorer.

Se référant des données journalières reçues de part une station météo à coté nous pourrions prédire l'Humidité du sol. Ainsi, avant d'entrer dans le vif du sujet, il serait bien de faire un parcours littéraire sur l'ensemble des modèles qui ont été récemment utilisés pour faire une prédiction de l'humidité du sol. Cet étude nous permettra d'avoir une aperçu sur les algorithmes utilisés en même de constater le modèle le plus adapté à notre cas d'étude.

À l'heure actuelle (2020...), les principales méthodes de prévision de l'humidité du sol utilisent principalement des formules empiriques, une régression linéaire et des réseaux de neurones pour construire des modèles de prévision. Le modèle de formule empirique est le plus ancien.

En analysant la teneur initiale en eau du sol, les précipitations quotidiennes, la température moyenne et la différence de saturation moyenne quotidienne, et sur la base de la relation linéaire multi variée de l'humidité du sol, Chen Xiaofeng [14] et d'autres ont établi une formule pour l'humidité du sol, les précipitations et la sécheresse modèle de prévision de l'évaluation, qui peut fournir une évaluation de la sécheresse pendant 5 à 10 jours à l'avenir. Le modèle fournit des stratégies pour des systèmes d'irrigation résistants à la sécheresse.

Jackson, dans [15], utilise la formule empirique pour estimer le flux d'humidité du sol avec un instrument de réflectométrie dans le domaine temporel (TDR). Les résultats sont similaires, mais la formule est plus simple. Bien que la formule empirique soit simple et facile à comprendre, les paramètres du modèle ont une forte dépendance régionale et doivent être recalculés lors de la transplantation dans d'autres régions, ce qui prend du temps et est inefficace.

Avec le développement rapide de la technologie informatique, divers modèles de prédiction ont vu le jour. J.W. Hummel [16] a utilisé un capteur de réflexion dans le proche infrarouge pour collecter les données d'humidité du sol et a analysé les données en utilisant une régression linéaire multiple. Ce qui en résulte un écart type prévu de 5,31%.

Après l'analyse de corrélation grise des données météorologiques afin de trouver les composantes principales intervenant à la variation de l'humidité du sol, Shu Sufang [17] a établi un modèle de régression linéaire pour prédire ce dernier, qui peut montrer ses tendances saisonnières.

La régression linéaire présente des erreurs relativement importantes et une précision insatisfaisante pour la prédiction de données non linéaires en raison de limitations internes et a du mal à répondre aux exigences de prévision.

Avec l'optimisation des algorithmes de formation, les chercheurs ont progressivement commencé à utiliser des algorithmes de réseau neuronal pour la prédiction de l'humidité du sol. Hou Xiaoli [18] et al. utilisaient un réseau de neurones artificiels pour prédire les valeurs d'humidité du sol à différentes profondeurs avec des données météorologiques à entrées multiples, et les résultats étaient en bon accord avec les données réelles.

Sur cette base, Ji Ronghua [19] a amélioré la fonction d'activation du réseau neuronal. La fonction d'activation traditionnelle a été remplacée par un domaine numérique complexe, et le réseau a été formé selon la structure du perceptron multicouche. La précision de la prédiction s'est améliorée de 9,1% par rapport au réseau neuronal traditionnel de rétropropagation (BP). Fournissant une base théorique plus précise pour la prédiction de l'humidité du sol, M. Kashif Gill [20] a évité la malédiction du problème de dimensionnalité dans les réseaux de neurones en utilisant un support machine vectorielle pour prédire l'humidité du sol et augmenter la précision à 89%.

Ainsi Wei Xu et al. [21] fera appel à la méthode d'approximation alpha proposée en 2011 tout en faisant abstraction aux questions de contraintes et de système d'équation sous-déterminé pour augmenter les données mesurées. Ensuite, ces données sont appliquées dans une machine de régression à vecteur de support pour l'estimation de l'humidité du sol. En comparaison avec la méthode alpha, les résultats sont plus satisfaisants.

Li Ning [22] a amélioré l'algorithme d'optimisation du réseau neuronal basé sur les caractéristiques des données de l'humidité du sol. L'algorithme BP a une vitesse d'apprentissage lente et tombe facilement dans les optima locaux car les paramètres initiaux du réseau sont attribués de manière aléatoire. Par conséquent, l'algorithme génétique a été introduit pour



trouver les paramètres initiaux optimaux globaux avant la formation, ce qui accélère efficacement la formation et améliore la précision des prédictions du modèle.

Caojun Huang et al. [23] mettent en place un modèle de prévision de l'humidité du sol basé sur un réseau neuronal génétique car la prévision de l'humidité du sol est non linéaire et complexe. Les poids et la valeur seuil du réseau BP sont optimisés en fonction de la capacité d'optimisation de la situation totale de l'algorithme génétique, ce qui peut éviter efficacement que le réseau BP soit vulnérable à la valeur minimale locale comme sa faible capacité d'optimisation totale. Le résultat de la prévision montre que le modèle a une précision de prévision favorable, ce qui indique que le modèle de réseau neuronal génétique est faisable et efficace pour prédire l'humidité du sol.

Olutobi Ademeyi [24] présente une approche Dynamic Neural Network pour la modélisation des flux temporels d'humidité du sol. Les modèles sont formés pour générer une prévision à un jour de la teneur volumétrique en humidité du sol en fonction de l'humidité du sol, des précipitations et des mesures climatiques passées. En utilisant les données de terrain de trois sites, une valeur  $R^2$  supérieure à 0,94 a été obtenue lors de l'évaluation du modèle dans tous les sites. Les modèles ont également été en mesure de générer de solides prévisions d'humidité du sol pour des sites indépendants qui n'ont pas été utilisés pour la formation des modèles. Le système de planification prévisionnelle de l'irrigation a été évalué par rapport à un système basé sur des règles qui applique l'irrigation sur la base de seuils prédéfinis. Les résultats indiquent que le système prédictif permet une économie d'eau comprise entre 20 et 46% tout en réalisant un rendement et une efficacité d'utilisation de l'eau similaires à ceux du système basé sur des règles.

Cependant Ramendra et al. [25] ont utilisé deux utilitaires multi-résolutions polyvalents, efficaces en termes de calcul et auto-adaptatifs (formules empiriques) pour résoudre les problèmes de stationnarité des données avant d'appliquer des modèles hybrides intelligents de données d'apprentissage automatique extrême (ELM).

Toutefois, l'humidité du sol implique des effets structurels et des facteurs météorologiques complexes, et il est difficile d'établir un modèle mathématique idéal pour la prévision de

l'humidité du sol. Les caractéristiques de la structure et les algorithmes du réseau neuronal traditionnel sont faibles pour le traitement des méga données, la précision de la prédiction est difficile à améliorer davantage, et la capacité de généralisation et l'évolutivité sont limitées.

Avec le développement rapide de l'intelligence artificielle ces dernières années, Hinton [30] a proposé en 2006 le Deep Learning (DL), qui utilise une structure à plusieurs couches cachées pour augmenter la classification et la capacité d'ajustement aux données volumineuses et multifonctionnelles. Comparé aux réseaux de neurones traditionnels, il montre une forte puissance de calcul et a été appliqué avec succès dans plusieurs domaines. Selon la nature non linéaire et extrêmement complexe du sol, certains chercheurs ont introduit la DL dans l'analyse de la taille des particules du sol et de la texture du sol [26, 27] ces dernières années, surmontant les problèmes de faible précision des prévisions.

Sur cette base, Cai et al. [28] ont construit et optimisé un modèle de prédiction de l'humidité du sol grâce à un apprentissage en profondeur (DNNR) et ses puissantes capacités de traitement des données pour réaliser une prédiction de haute précision de l'humidité du sol à Pékin. Ces derniers ont effectué au préalable un prétraitement sur les données météorologiques afin de pouvoir trouver des relations entre les caractéristiques et les variables prédictives.

## Chapitre III : Traitement en périphérie et Évaluation de performances

Après avoir présenter l'ensemble des thématiques sur lequel porte notre sujet. Nous allons, se basant sur quelques méthodes proposées par les scientifiques, proposer des algorithmes de traitement à la périphérie du réseau et faire leur expérimentations.

Le Machine Learning a traditionnellement été réalisé uniquement sur des serveurs et des machines hautes performances. Cependant, les progrès de la technologie des puces nous ont permis de disposer de bibliothèques miniatures qui tiennent dans nos poches et les capacités des processeurs mobiles ont considérablement augmenté, réduisant ainsi le fossé qui sépare les simples processeurs intégrés à ces éléments et leurs cousins plus complexes sur des ordinateurs personnels. Ainsi, avec l'avancement actuel de ces périphériques, en termes de puissance de traitement, de stockage d'énergie et de capacité de mémoire, il est devenu possible de tirer un avantage précieux de l'apprentissage automatique en périphéries pour les appareils dédiés à l'Internet des objets (IdO). La mise en œuvre de l'inférence d'apprentissage machine sur des périphériques de pointe présente un potentiel énorme et en est encore à ses débuts [29].

Il a été montré que dans cadre de l'IdO, au sein d'une carte raspberry, les algorithmes de Machine Learning qui sont classique notamment les SVM, les Random Forest, et les perceptrons multicouches peuvent y être déployé avec une précision dépassant les 80% [29].

Les réseaux de neurones (Deep Learning) étant une perspective pour les travaux de Yazici et al. [29] nous ont incités à relever ce challenge en plus de vérifier la capacité d'exécution pour les appareils dédiés à l'IdO. Notre travail consistera à faire du traitement en périphérie du réseau, faisant abstraction du Cloud et usant des algorithmes statistiques afin de faire des prédictions sur les conditions agricoles de différents champs. Ainsi, pour cette solution nous

ferons une description générale du système dans lequel nous allons évoluer partant de la phase d'acquisition au traitement. Ainsi présentant ce système de manière globale nous pourrons parler de l'Architecture Gateway que l'on a utilisé pour implémenter notre solution. Cette description sera suivie de la présentation des jeux de données qu'on utilisera pour déployer un modèle sur la carte raspberry. Il s'en suivra la création du modèle d'apprentissage et des technologies utilisées pour enfin évaluer les performances du modèle et de la carte.

## I. Architectures (Système et Gateway)

### I. 1 Architecture Système

Toujours dans le cadre de L'IdO nous nous mettons dans un contexte où nous disposerons d'un système de Réseaux de capteurs déployés dans une zone (un champ). Ces dispositifs sont connectés sur un IdO Gateway jouant le rôle de pont entre les données mesurées et les serveurs distants où ils sont définitivement stockés. La technologie dont fera usage les capteurs pour transmettre les données au Gateway est LoRa.

LoRa est une technologie Radio propriété du fabricant de composants Semtech. Le groupement LoRa Alliance a défini un protocole appelé LoRawan qui permet à toute entreprise de déployer son propre réseau. LoRawan est un protocole de télécommunication permettant la communication à bas débit, par radio, d'objets à faible consommation électrique communiquant selon la technologie LoRa et connectés à l'Internet via des passerelles, participant ainsi à l'Internet des objets [33]. Ce protocole est utilisé dans le cadre des villes intelligentes, le monitoring industriel ou encore l'agriculture. Le signal radio est émis sur une grande largeur spectrale, pour limiter au maximum le risque d'interférence avec des signaux parasites. Cette technique de modulation est utilisée pour les communications spatiales et militaires. Ce protocole de communication permet d'envoyer des données en intérieur (indoor), en sous-sol (deep indoor) et en extérieur (outdoor). Cette technologie de modulation de fréquence ne peut faire circuler que de petits paquets de données, émis par des capteurs de température ou d'humidité par exemple. Elle peut faire transiter entre 0,3 et 50 kilobits par seconde (le débit du réseau s'adapte à chaque objet pour ne pas grignoter trop de bande passante) [33].

Ces informations peuvent transiter sur des distances plus longues que sur les réseaux télécoms traditionnels. Un objet connecté en LoRa peut envoyer un message à une borne située à une distance d'environ 1 kilomètre en zone urbaine et à 20 kilomètres dans une zone rurale plane [33]. Ces informations peuvent transiter sur des distances plus longues que sur les réseaux télécoms traditionnels. Un objet connecté en LoRa peut envoyer un message à une borne située à une distance d'environ 1 kilomètre en zone urbaine et à 20 kilomètres dans une zone rurale plane.

Une fois les données arrivées au Gateway elles seront temporairement stockées et répliquées sur le serveur à distance / Cloud lorsqu'il aura à disposition une connexion. C'est au niveau de ce Gateway que nous concentrerons notre traitement. Ainsi des interfaces seront déployées au niveau local et global afin de pouvoir faire la visualisation et l'exploitation des données.

Les données se trouvant sur le Gateway feront usage du protocole MQTT pour transmettre les données sur le serveur.

Ainsi l'ensemble de l'architecture est décrite ici (*voir figure 10*) et nous allons nous orienter sur le Gateway. Là où tout se fera.

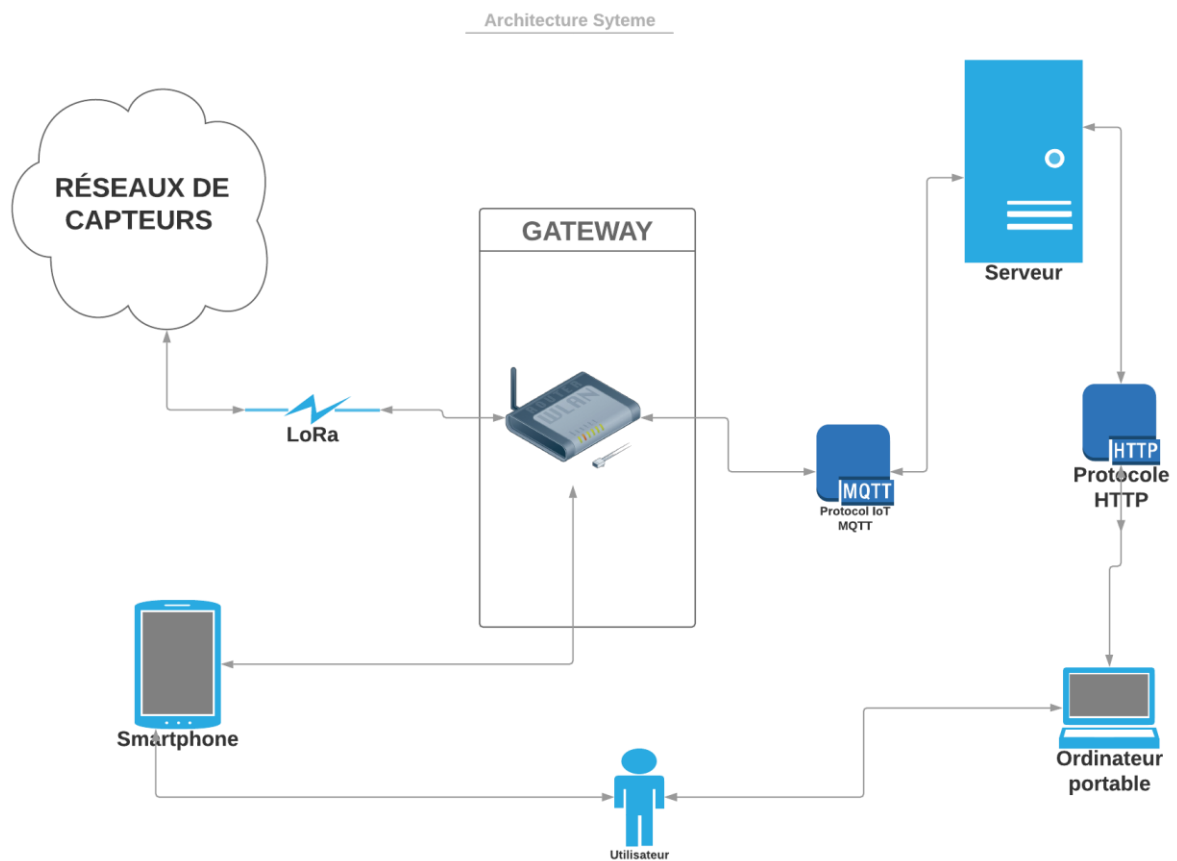


Figure 10: Architecture système

## I. 2 Architecture déployée sur le Gateway

Une passerelle Internet des Objets (IdO) est un périphérique physique ou un logiciel servant de point de connexion entre le Cloud et les contrôleurs, les capteurs et les périphériques intelligents. Toutes les données transférées vers le Cloud, ou inversement, passent par la passerelle, chose qui peut être un équipement physique dédiée ou un programme logiciel. Une passerelle IdO peut également être appelée passerelle intelligente ou niveau de contrôle.

Certains capteurs génèrent des dizaines de milliers de données par seconde. Une passerelle permet de prétraiter ces données localement sur le bord avant de les envoyer au Cloud. La passerelle IdO, comprenant le matériel hôte, le boîtier de connexion bus et les périphériques, collecte les données des capteurs et des contrôleurs, les filtre selon des règles préétablies et les transmet à un point central (Cloud, Cloudlet). Lorsque les données sont agrégées, résumées et analysées de manière interprétative à la périphérie, le volume de données à transférer sur le Cloud devient minime, ce qui peut avoir un impact important sur les temps de réponse et les coûts de transmission du réseau.

Un autre avantage du Gateway IdO est qu'elle peut renforcer la sécurité du réseau et des données qu'il transporte. Dans la mesure où la passerelle gère les informations circulant dans les deux sens, elle peut protéger les données transférées vers le Cloud des fuites et empêcher que les périphériques IdO ne soient compromis par des attaques malveillantes de l'extérieur, grâce à des fonctionnalités telles que la détection de sabotage, le chiffrement, les générateurs de nombres aléatoires et les moteurs de chiffrement.

L'architecture de la passerelle IdO que nous utilisons à l'occasion de cette étude est un système assez complexe et dont l'usage est d'une importance qui pourra découler sur des performances notoires. Ainsi, il sera doté d'une API Broker (en supposant qu'il y a plusieurs entités) censée échanger avec tous les éléments interagissant avec le Gateway de la phase d'acquisition de données ainsi que les serveurs cloudlets à distance en plus des interfaces utilisateurs (*figure 11*). La passerelle IdO communique avec le réseau de capteurs grâce à la technologie LoRa pour récupérer des données ou exécuter des actions. Les données émises par les capteurs sous forme de topiques passent par le biais du Broker pour être stockées dans la base de données locale. Ces mêmes données sont répliquées sur le serveur à distance (cloud) par le protocole MQTT. Divers types de Base de données pourraient être créés sur la passerelle pour stocker les données de capteurs, historier les données déjà stockées et gérer les utilisateurs qui se connectent au Gateway.

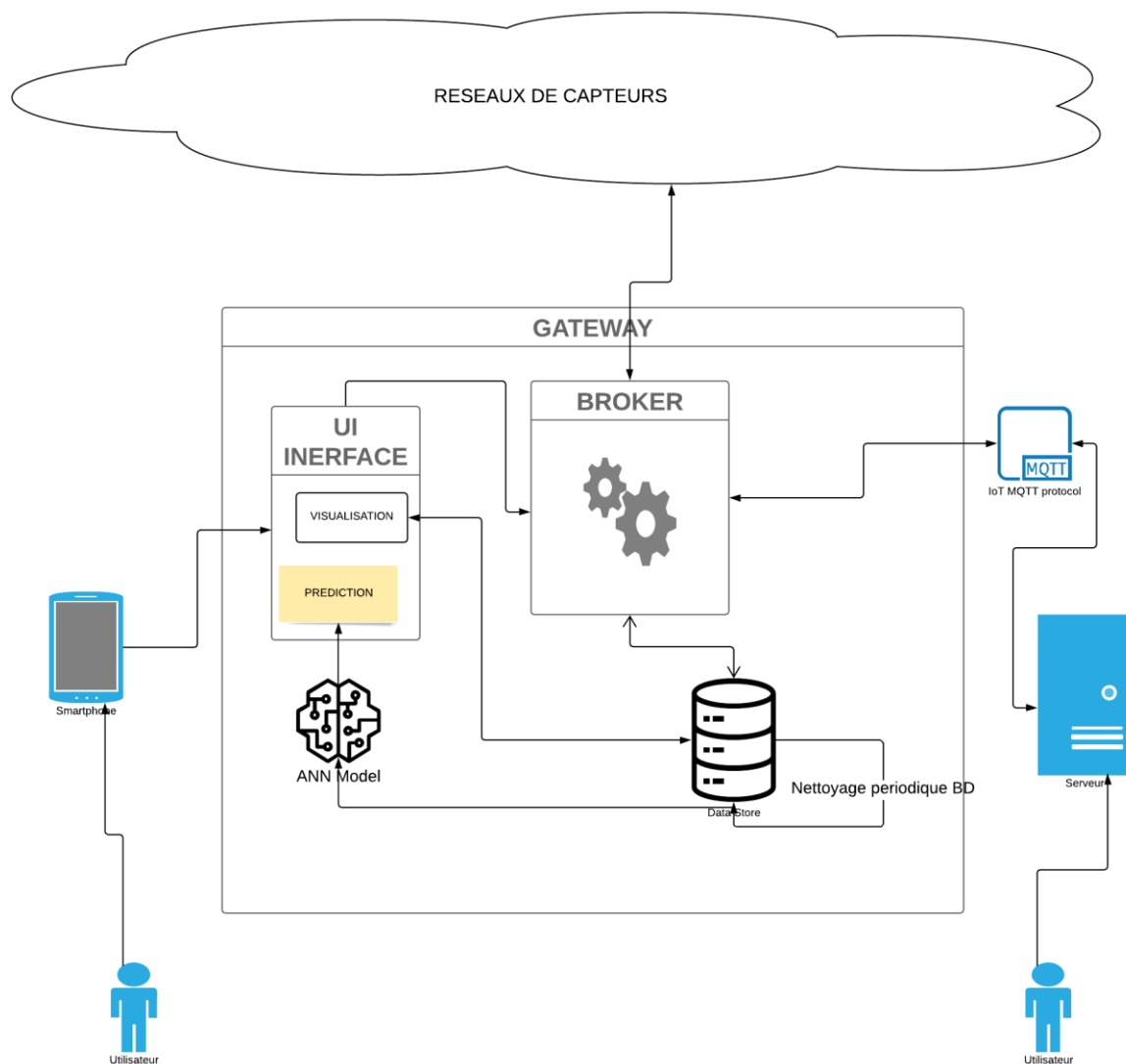


Figure 11: Architecture Gateway

A cela s'ajoute une Api permettant de visualiser l'évolution des données. Cette interface pourra être utilisée par les clients qui se connectent à la passerelle et qui disposent d'identifiants et de mots de passe.

Ceci est similaire à l'architecture proposée par Waziup [31] sur leur Gateway dénommée Wazigate. Le wazigate est une passerelle IdO avec une architecture dédiée qui disposent des composants lui permettant une fois configuré de recevoir des données de capteur et de les transmettre automatiquement à un serveur distant. Nous ferons usage de cette passerelle mais en y apportant quelques modifications suivant nos besoins.



Notre but principal est de prédire l'Humidité du sol grâce aux données d'une station météo se basant sur un ensemble de jeux de données pour construire un modèle que nous déploierons sur la carte. Une fois les données d'une station météo envoyées sur le Wazigate, notre modèle fera usage des données envoyées pour prédire l'humidité du sol. Ce modèle sera au préalable entraîné sur cette passerelle. Cet apprentissage nous permettra d'évaluer la robustesse et la capacité du dispositif à entraîner des modèles. La valeur obtenue sera stockée sur la base de données locale de même que sur le Cloud pour un stockage permanent.

## II. Jeux de Données et Prévisualisation

### II. 1 Les Données Environnementales

Le fait de réaliser le système dans sa globalité et de mettre en pratique les concepts expliquées afin d'avoir assez de données est une tâche qui prend un peu de temps. C'est la raison pour laquelle nous avons fait recours à un jeu de données sur l'agriculture portant sur les informations environnementales de ces derniers notamment l'humidité du sol, la température entre autres pour évaluer la performance de l'architecture du Gateway et la mise en pratique des réseaux de neurones sur une carte Raspberry pi.

Ce jeu de données a été collecté sur la plateforme Zindi dans le cadre d'une expérience menée par Waziup à l'aide de capteurs Internet à bas coût (IdO) sur une période de 4 mois dans 4 champs de maïs et d'arachides au Sénégal [32].

Un capteur IoT a été placé dans quatre parcelles distinctes plantées de maïs ou d'arachides. Les parcelles sont côte à côte, séparées par un périmètre d'un mètre (*figure ci-dessous*).

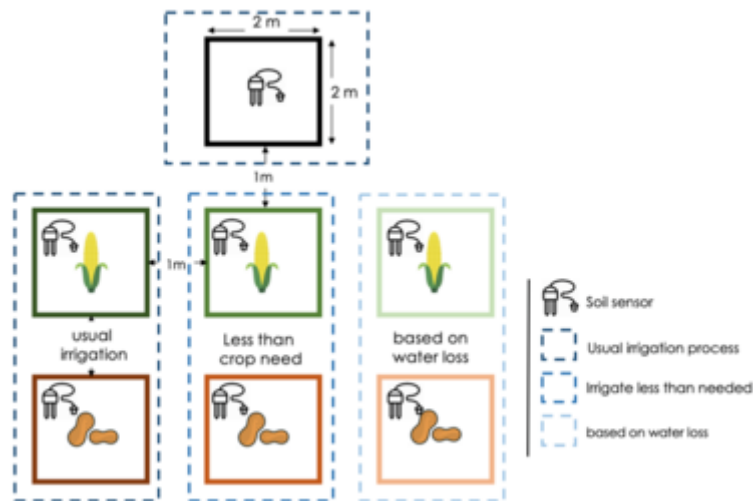


Figure 12: Organisation Parcelles et disposition des capteurs

Src : <https://zindi.africa/competitions/wazihub-soil-moisture-prediction-challenge/data>

Chaque parcelle a été irriguée de manières différentes Il existe trois types de calendriers d'irrigation:

L'irrigation habituelle consistant à arroser le champ tous les deux jours. La deuxième irrigation est faite moins que les besoins des cultures: moins d'eau tous les deux jours, c'est-à-dire que les cultures ont été irriguées à des intervalles irréguliers, ce qui leur a donné moins que ce qui était nécessaire. Le dernier type d'irrigation sera fait suivant la perte en eau: L'eau donnée est basée sur une perte d'eau estimée. Les pertes d'eau estimées ont été calculées à l'aide de différents paramètres, notamment l'évapotranspiration et le niveau d'humidité du sol, collectés à partir des capteurs IdO.

L'objectif de ce défi était de prévoir avec précision le niveau d'humidité du sol. Cette solution aidera les agriculteurs à préparer leurs calendriers d'irrigation de manière plus efficace.

Les données recensées sur lesquelles porteront notre modèle sont issues de quatre champs chacun ayant un type d'irrigation particulière. Ces données sont émises toutes les 5 minutes.

Notre solution sera de mettre en place un modèle de prédiction de l'humidité du sol. Et pour cela un jeu de donnée a été mis à disposition. Ainsi par peur d'incohérence dû à une différence du type d'irrigation utilisé nous ne traiterons que les données de la première parcelle sur lequel

est planté du maïs avec un type d'irrigation moins que les besoins en eau des cultures. Ce choix a été fait aléatoirement.

Les capteurs d'humidité du sol ont été installés dans chacun des champs et une station météorologique a été installée près des champs. Ces appareils IdO ont transmis les données suivantes par intervalles de cinq minutes: l'Humidité du sol en (%), la Température de l'air (C), L'humidité de l'air (%), la Pression (KPa), la Vitesse du vent (km / h), le Rafale de vent (Km / h), la Direction du vent (Deg).

Une variable dénommée «irrigation» a été associée au champ. La variable d'irrigation est définie sur 1 lorsque l'irrigation est activée et que l'humidité du sol est en hausse et sur 0 lorsque l'irrigation est désactivée.

## II. 2 Prévisualisation des données

L'humidité du sol implique des caractéristiques structurelles et des facteurs météorologiques complexes et il est difficile d'établir un modèle mathématique idéal pour la prévision de l'humidité du sol. Les modèles de prédiction existants ont des problèmes tels que la précision de la prédiction, la généralisation et la capacité de traitement multifonctionnel, et les performances de prédiction doivent s'améliorer. Ce qui nécessite une connaissance des données et des paramètres en jeu. La compréhension de ces dernières apportera des informations sur les dépendances entre variables et les composants intervenant sur la variation de l'humidité du sol.

Dans notre contexte afin de pouvoir choisir les bons composants intervenant sur la variation de notre variable cible, une visualisation de données sur graphe a été faite pour suivre leur évolution au cours du temps. Ces graphes nous ont permis d'en déduire une dépendance entre différentes variables des facteurs météorologiques. Cette visualisation a été possible grâce au langage de programmation python.

Les données auxquelles nous faisons usage comportent les variables suivantes (*voir figure 12*): le temps par intervalle de cinq (5) minutes, la température de l'air en degré Celsius (C) (AirTemperatureC), l'humidité de l'air en pourcentage (AirHumidityPercent), la Pression atmosphérique en Kpa (PressureKPa), la vitesse du vent en Kilomètre par heure

(WindSpeedKmh), le coup de vent en Kilomètre par heure (WindGustKmh), la direction de l'air en degré (WindDirectionDeg), l'humidité du sol en pourcentage (soilHumidity), et l'irrigation du terrain qui ne contiennent que deux valeurs (0-1) (irrigationField).

timestamp	AirTemperatureC	AirHumidityPercent	PressureKPa	WindSpeedKmh	WindGustKmh	WindDirectionDeg	soilHumidity	irrigationField
2019-02-23 00:00:00	19.52	55.04	101.50	2.13	6.30	225.00	67.92	0.0
2019-02-23 00:05:00	19.49	55.17	101.50	2.01	10.46	123.75	67.89	0.0
2019-02-23 00:10:00	19.47	55.30	101.51	1.90	14.63	22.50	67.86	0.0
2019-02-23 00:15:00	19.54	54.20	101.51	2.28	16.08	123.75	67.84	0.0
2019-02-23 00:20:00	19.61	53.09	101.51	2.66	17.52	225.00	67.81	0.0

Figure 13: Entête des données de la station météorologique

Le champ contient 8917 lignes datant du 23 Février 2019 à minuit au 29 Mars à 19 heures 50 minutes. Ainsi la variation en une journée de l'humidité du sol associé à la température de l'air et de l'indicateur d'irrigation du champ a cette allure (*figure 14*) :

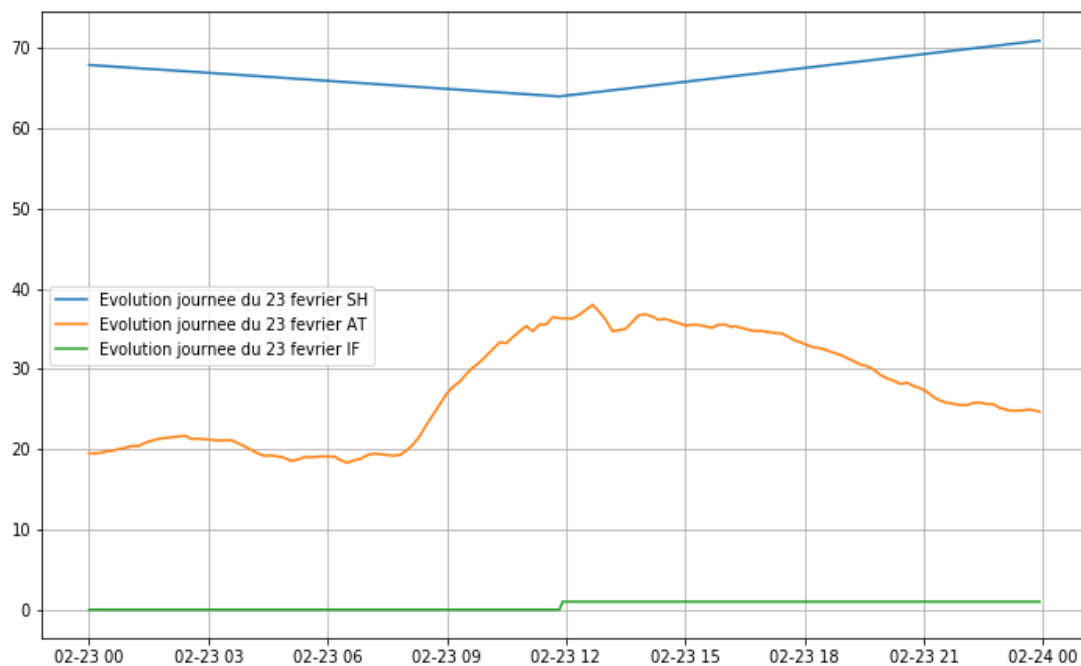


Figure 14: Évolution de l'humidité du sol, de la Température de L'Air et de l'irrigation du champ à la date du 23 Février

D'emblée nous pouvons dire que l'augmentation de la température de l'air de même que la variation des autres variables influent de manière négligeable sur l'humidité du sol lorsque le champ est irrigué. De ce fait, leur impact ne peut être visible qu'à la période où le champ ne l'est pas.

Ce qui permet d'émettre la supposition à savoir que si le champ n'est pas irrigué la température de l'air agit sur la diminution de l'humidité du sol hormis les besoins en eau de la plante.

Afin de vérifier notre théorie nous avons visualisé la variation de ces composants ci-dessus sur tout le mois de février ce qui donne ce graphe ci:

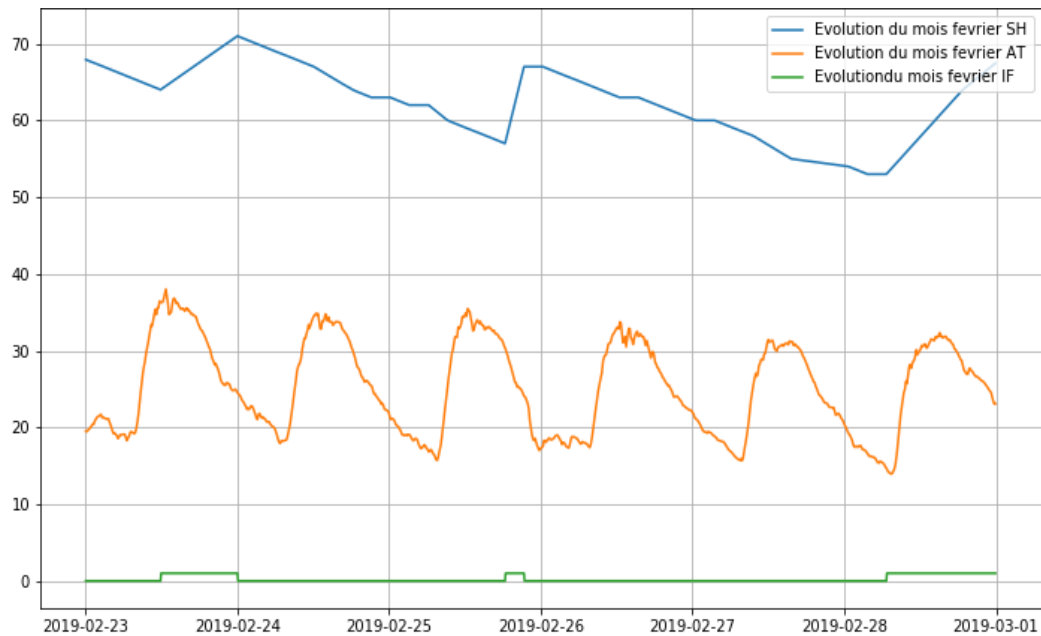


Figure 15: Évolution de l'humidité du sol, de la Température de L'Air et de l'irrigation du champ au mois de Février.

Ce graphique prouve que si le champ est irrigué peu importe la température de l'air l'humidité du sol augmente. Reste à voir à quel point la température de l'air impact sur l'humidité du sol en temps non irrigué. Cette vérification nous incita à tracer un graphe sur lequel figurent tous les paramètres que l'on a à notre disposition dans le but d'analyser leur variation comparé à la température de l'air. Le tout est affiché en fonction du temps.

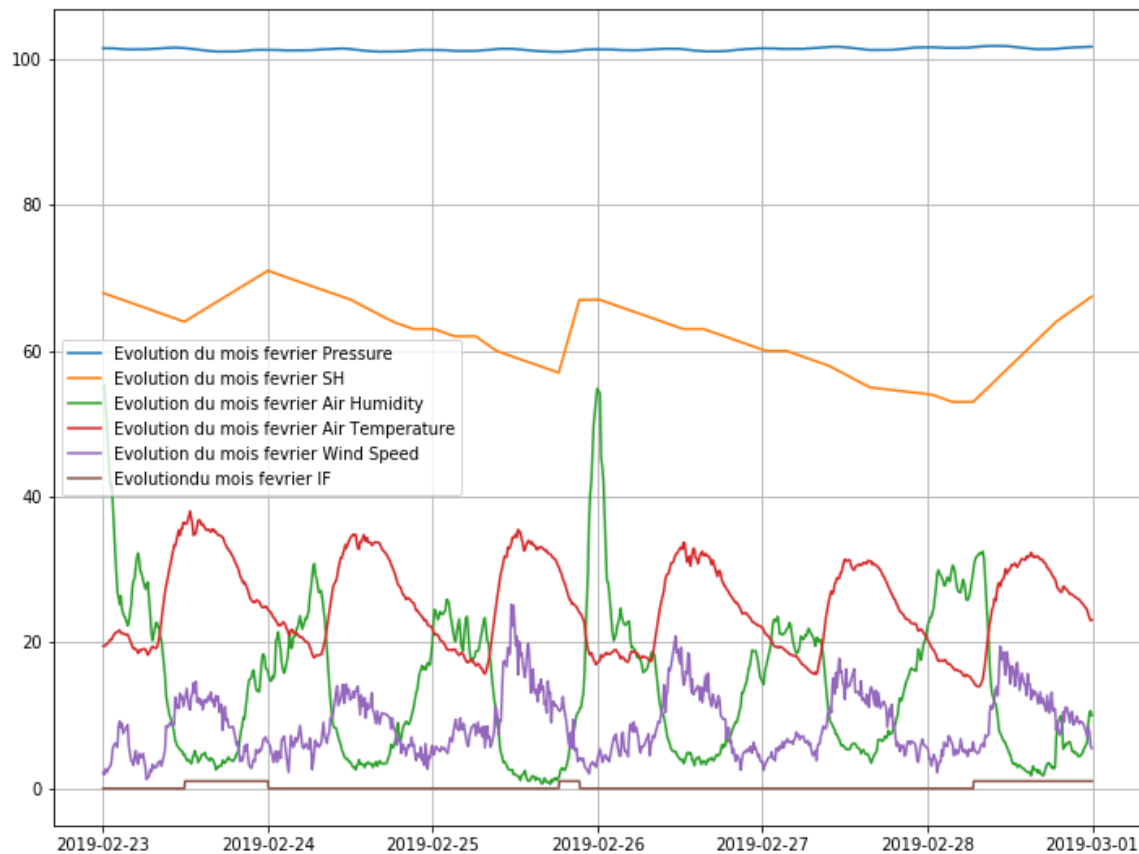


Figure 16: Évolution de la pression, de la Température de L'Air, de l'humidité de l'air, de l'irrigation du champ et de la vitesse du vent au mois de Février.

La variation de la pression n'influe aucunement sur la variation de l'humidité du sol (Soil Humidity) car elle garde une même allure tout au long du mois. L'humidité de l'air diminue au fur et à mesure que la température de cette dernière augmente. La température de l'air influe sur l'évapotranspiration (obtenu par calcul manuel) et la diminution de l'humidité du sol. On peut en déduire un même sens d'évolution de l'humidité de l'air et celui du sol. C'est à dire que dans une période où le champ n'est pas irrigué si l'humidité de l'air diminue alors l'humidité du sol en fait de même. On peut dire aussi que la vitesse du vent est faible en température minimum et forte en température maximum.

Le tout constitue l'ensemble des observations qui sont effectuées sur l'évolution des données.

### III. Modèles d'apprentissage

La visualisation des données suivant le temps nous a permis d'avoir un aperçu sur les différents paramètres intervenant sur la variation de l'humidité du sol. Ainsi dans le choix des variables qui constitueront les entrées de notre modèle nous avons choisi 4 principaux paramètres à savoir: La température de l'air, l'humidité de l'air, la direction et la vitesse du vent. Afin de pouvoir bien évaluer les performances, les données d'entrées ont été standardisées en parallèle et stockées sur une autre variable.

Les modèles ont été entraînés au niveau d'une carte Raspberry pi 3 model B+ RAM 4 Go et une mémoire de 32Go permettant de stocker les données.

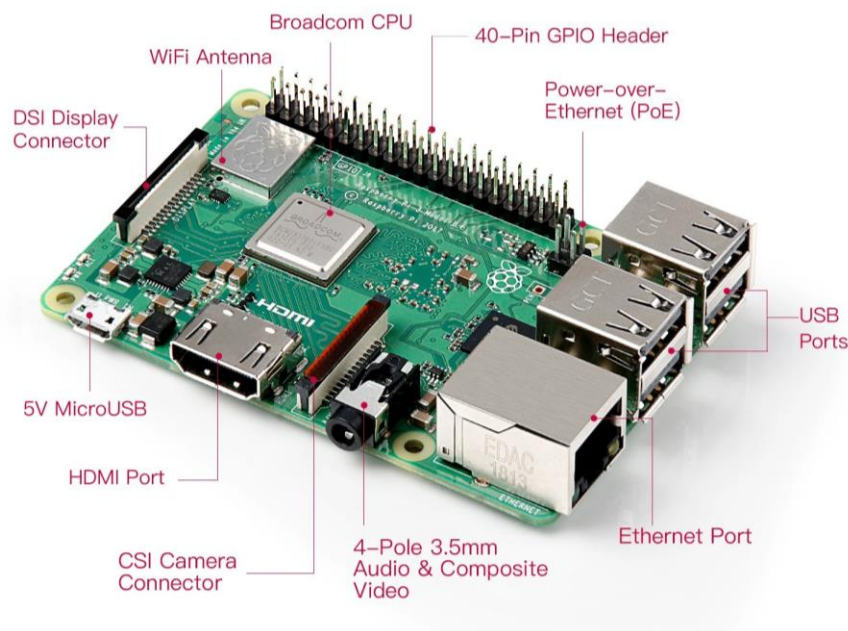


Figure 17: Raspberry Pi 3



Ce dispositif avec des capacités assez réduites comparées aux ordinateurs de nos jours nous servira de passerelle et d'instrument pour stocker les données issus des capteurs. Ainsi en entraînant les modèles sur cette dernière nous pourrons prendre en compte de la capacité de ces types d'équipement et songer à tester ses limites.

Du fait que la variable prédite est de type continu, nous choisirons une technique de régression. C'est en ce sens que diverses méthodes ont été ciblées pour créer notre modèle.

Les données sont prétraitées au préalable permettant d'isoler ceux qui sont émises en période où il n'y a pas d'irrigation. Après extraction nous nous retrouvons avec une base de 8914 lignes. C'est en ce sens que des modèles ont été ciblés, nous avons utilisé des techniques de Machine Learning pour mettre en place notre modèle. Les méthodes utilisées sont: Le Support Vector Machine, le perceptron multicouche, Un modèle de Régression avec une couche cachée et un modèle composé de plusieurs couches. Nous nous baserons sur l'erreur quadratique moyenne pour évaluer les modèles.

### III. 1 Perceptron Multi Couche

Le perceptron multicouche pour la régression nous a semblé opportun à l'occasion de la création de modèle. Ce dernier a été réalisé sur python avec la bibliothèque scikit-learn qui constitue un ensemble de fonctions utiles pour faire du Machine Learning. Le perceptron multicouche, introduit par Rosenblatt en 1957, est une méthode qui apprend par expérience. Le perceptron prend en entrée un vecteur à plusieurs dimensions (1 par neurone) et opère une séparation entre ces données pour fournir une sortie. Grâce à cette séparation qu'il a construite entre les données, il sait, pour un nouvel exemple, quelle doit être la réponse.

Le modèle ainsi conçu est composé de cents couches cachées et d'un Learning rate initialisé à 0.1. Le modèle prend en entrée quatre variables d'entrées: La température de l'air, l'humidité de l'air, la vitesse du vent et la direction du vent. L'entraînement du modèle sur les deux types de données donna:

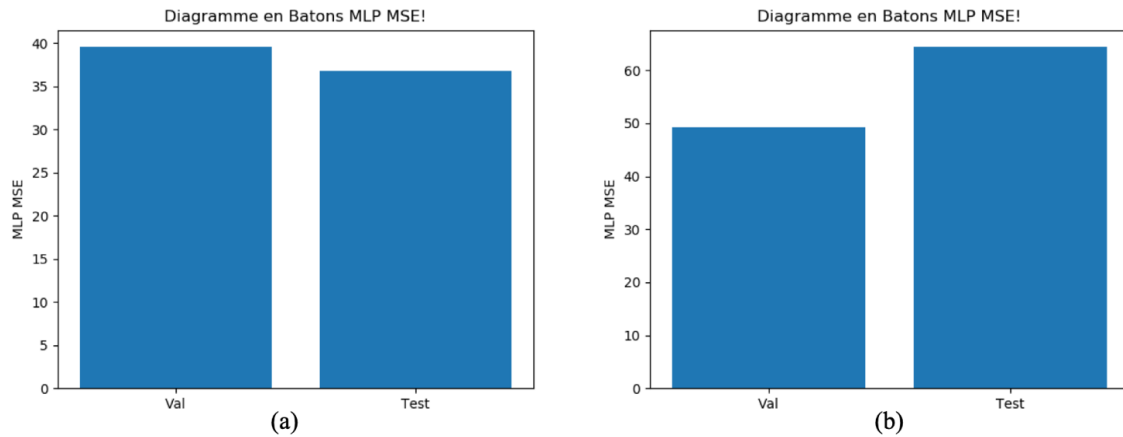


Figure 18: Diagramme en Barre MSE Scaling Data(a) & No Scaling Data (b) Multi layer Perceptron

S'agissant des données standardisées, nous obtenons comme MSE pour les données de Validation une valeur égale à 39 et une valeur de 37 pour les données tests. Ce modèle avec les mêmes paramètres de configuration donnera pour les données brutes une erreur de 58 et de 64 respectivement pour les données de validation et de Test.

### III. 2 Support Vector Machine pour la Régression SVR

Le Support Vecteur Machine est plutôt connu pour ses techniques de classification. Toutefois, il existe une version spécifique pour faire de la régression linéaire. SVR nous donne la flexibilité de définir le niveau d'erreur acceptable dans notre modèle et trouvera une ligne appropriée (ou un hyperplan dans des dimensions plus élevées) pour ajuster les données [34].

Le Scoring du modèle a permis d'obtenir les diagrammes suivants:

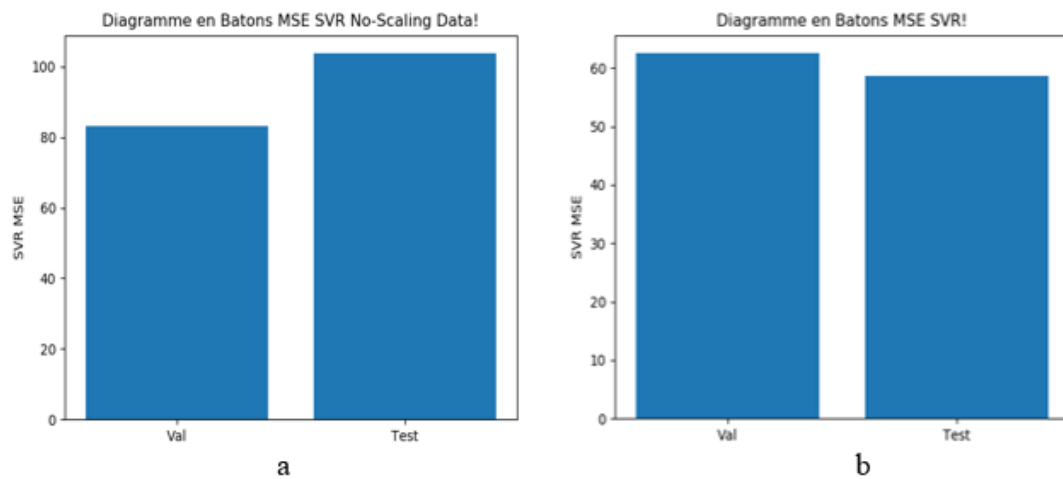


Figure 19: Diagramme en Barre MSE No Scaling Data (a) et Scaling Data (b) SVR

La prédiction des données nous a donné une erreur quadratique moyenne qui varie entre 85 et 108 pour les données brutes comparé aux données standardisées dont l'erreur quadratique moyenne est passée de 63 à 58 pour les données de validation et de test.

### III. 3 Modèle de Régression (Réseaux de Neurones) avec une couche cachée

Dans cette situation le choix des données standardisées n'étaient pas approprié car même après 4000 époques d'apprentissage sur les données la courbe de perte continue toujours à décroître.

Le modèle est Séquentiel n'a fait usage que d'une seule couche cachée pour entraîner le modèle.

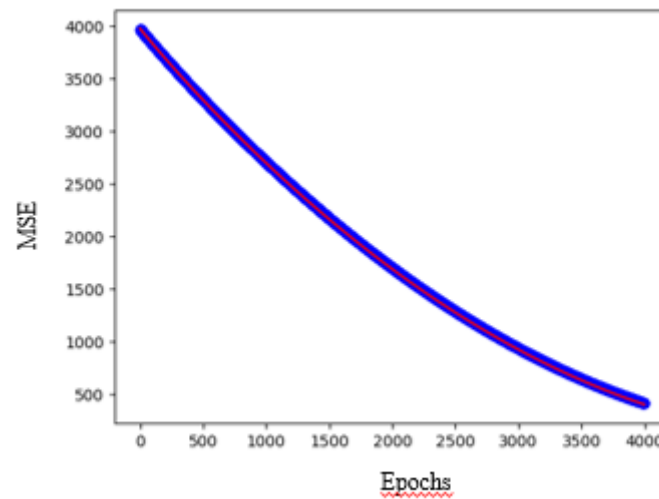


Figure 20: Courbe d'Évolution MSE Validation Data (Scale Data)

Faisant usage du même modèle pour les données brutes nous avons pu atteindre une constance au niveau de la courbe d'évolution de la perte pour pouvoir comparer le score entre les données test et celle de validation. (Voir figure 21)

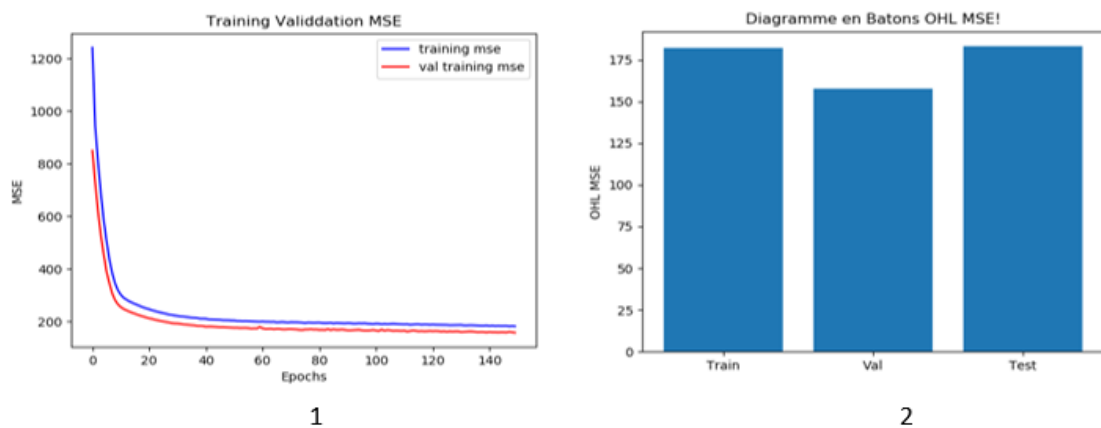


Figure 21: Courbe d'Évolution MSE Validation Data(1) & Diagramme en Barre MSE No Scaling Data

### III. 4 Modèle de Régression (Réseaux de Neurones) avec plusieurs couches cachées

Grâce à la bibliothèque de KERAS nous avons mis en place un modèle de cinq couches dont trois couches cachées avec 16 neurones décroissant par puissance de 2 ( $16 \rightarrow 8 \rightarrow 4$ ).

Ce modèle prend comme fonction d'activation la fonction ReLU (Rectified linear Unit) qui retourne le maximum entre une variable et 0. Comme Back propagation le modèle fait usage du Mean absolute Error comme fonction de perte de même que rmsprop (technique d'optimisation basée sur le gradient utilisée dans l'entraînement des réseaux de neurones) pour l'optimiseur. C'est ainsi que l'on a entraîné le modèle avec les données et de leur variable cible. L'évaluation du modèle effectué sur les données tests donnera les graphes suivants:

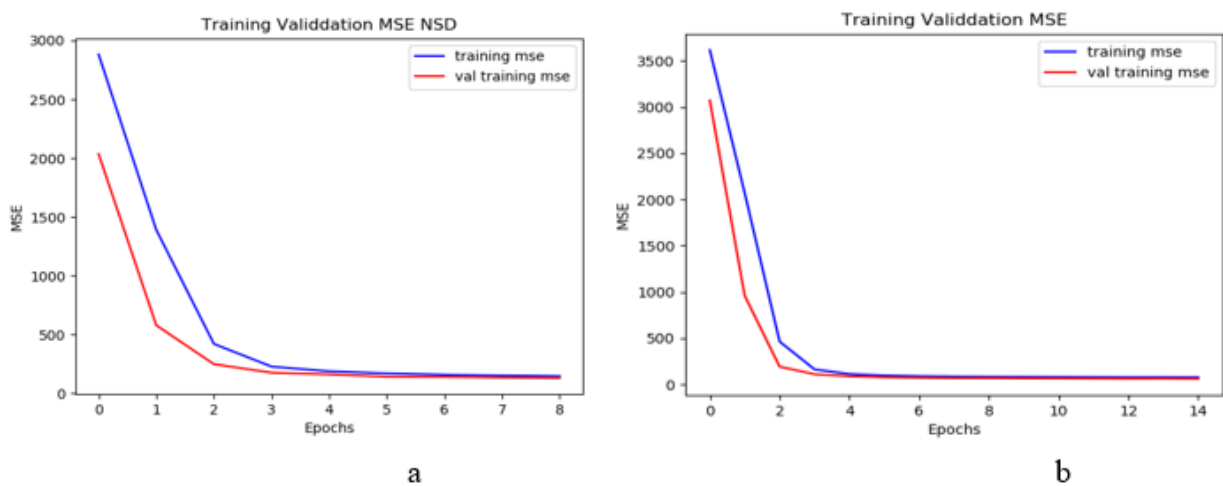


Figure 22: Courbe d'évolution de décroissance du MSE Données Brutes(a) et Données Standardisées (b)

Se basant sur les dernières valeurs de ces graphes associées au score obtenu lors de la prédiction des données de Tests nous avons tracé le diagramme comparatif des scores pour les deux types de données.

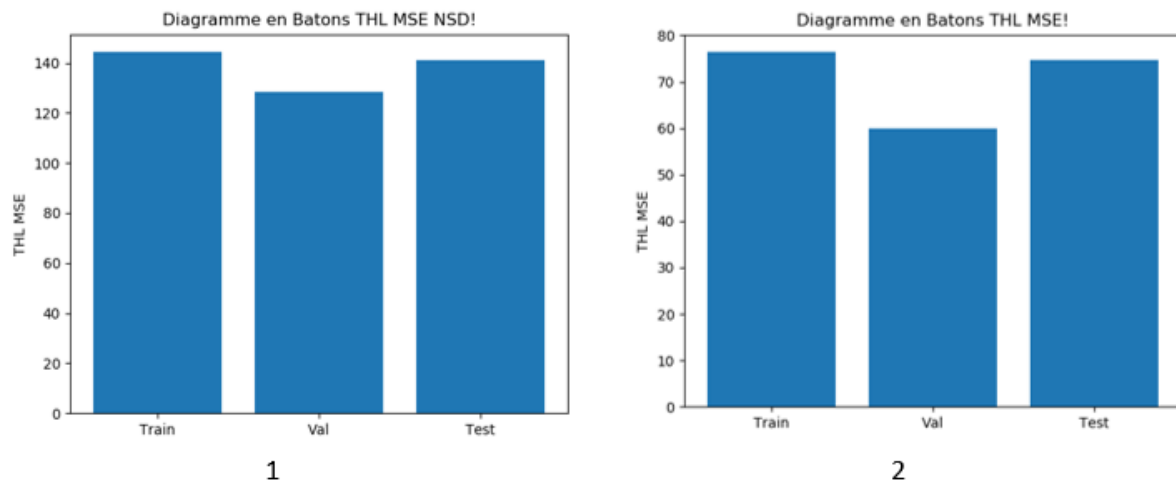


Figure 23: Diagramme en Barre MSE modèle de régression à plusieurs Couches Données Brutes (1) et Données Standardisées (2)

L'évaluation des deux graphes ci-dessous nous permet d'interpréter de la sorte: Le modèle est plus performant lorsque les données sont standardisées.

Afin de pouvoir faire un choix sur deux des modèles entraînés, un diagramme prenant en compte l'ensemble des modèles a été créé. Ces modèles seront déployer et analyser sur le Gateway,

## IV. Comparaison des Modèles:

La comparaison de ces modèles s'est faite à partir de l'erreur quadratique moyenne de l'ensemble des modèles qui ont été vus antérieurement et le choix des modèles se feront suivant le modèle qui présente les scores les plus faibles.

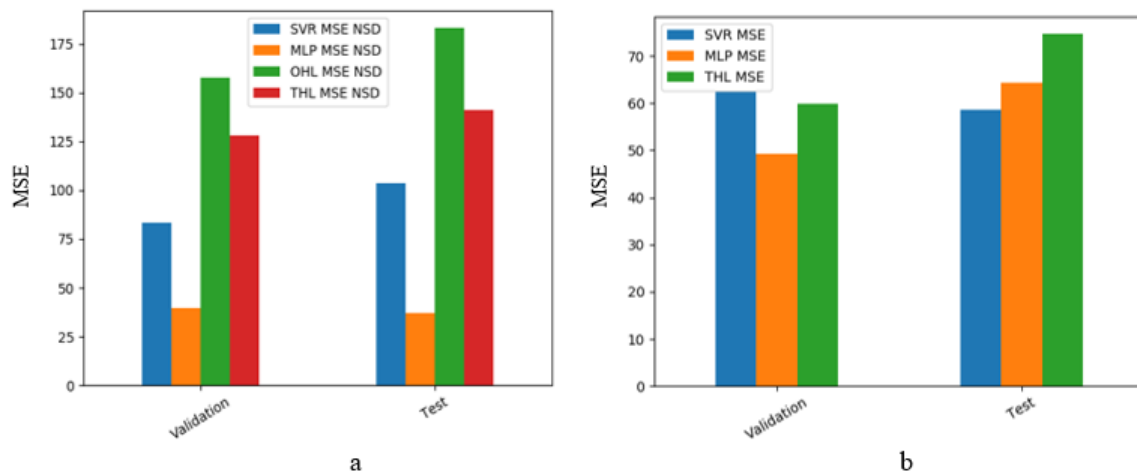


Figure 24: Diagramme de Comparaison (MSE) des différents modèles: Données Brutes(a) et Données Standardisées(b)

Dans le premier cas de figure avec les données brutes, le modèle du perceptron est le plus efficace avec la plus faible erreur suivi du modèle de SVR.

Dans le second cas de figure les modèles les plus performants se basant sur les données test sont ceux du perceptron et du SVR.

En conclusion nous utiliserons le perceptron et celui de la machine à support vecteur pour la régression comme modèle de prédiction pour nos tests sur le Gateway.

Ces modèles ont été stockés après entraînement et validation sur un fichier pickle qui pourra être utilisé pour faire de la prédiction.

#### IV. 1 Mise en pratique du Système / Expérimentations Gateway

L'expérimentation aux niveaux IdO a demandé beaucoup plus de tâches que prévu: le déploiement d'un script au niveau du Gateway permettant après récupération de données d'y appliquer le modèle pour ensuite les stocker. Et une configuration au niveau périphérique afin de simuler l'envoi de données issues de capteurs.

S'agissant du Gateway nous avons choisi une carte Raspberry dans lequel nous avons installé l'architecture waziup pour effectuer nos tests afin d'apporter des réponses à nos questions.

Dans un contexte de simulation de la station météo, nous allons paramétrer une carte Arduino de sorte qu'ils émettent des données climatologiques composées de l'ensemble des paramètres dont l'on aura besoin avec des valeurs semblables à celle de la station. Ainsi ces derniers, par le biais du réseau LoRa, seront envoyés au Gateway où on procédera à la prédiction. La carte sera munie dans ce cas d'un module radio grâce à laquelle l'émission se fera.

Le script déployé au niveau du Gateway sera exécuté et aura pour rôle après envoi des données issues des capteurs de récupérer les paramètres dont il aura besoin et de leur valeur pour par la suite les stocker sur un dataframe. Cette variable servira d'input pour le modèle. En lui appliquant ce modèle nous aurons ainsi la valeur de l'humidité du sol prédite. Ainsi nous pourrions stocker l'ensemble de ces informations sur un fichier Excel que nous étudierons plus tard.



# Conclusions

## Interprétation des Résultats

Après avoir construit les deux modèles nous avons constaté que le score pourrait être meilleur car l'erreur est assez grande. Selon les observations faites nous pouvons en déduire le peu de jeu de données que l'on a eu pour créer ces modèles (8914 lignes). Car plus il y a de données cohérentes, plus il y a de chances de créer un modèle plus performant en comprenant les relations entre variables d'entrées et la variable cible. De plus, en parlant de données météorologiques, leurs variations se font sentir en trois points de la journée: le matin, à midi et le soir. Néanmoins nous ne disposons pas assez de données nous permettant de relever ces trois mesures le jour.

Au déploiement sur le Gateway avec l'architecture de Waziup nous avons réalisé des simulations grâce à une carte arduino qui est un microcontrôleur qui peut émettre de manière aléatoire des données semblables à ceux de la station météo. Ces données envoyées au Gateway par le réseau Lora permettent de prédire l'humidité du sol suivant les deux cas avec les modèles déployés et les stocker sur un fichier CSV local. Cette dernière nous a permis de prouver que les modèles Shallow Learning pourraient être entraînés et déployés sur une périphérique dépourvu de certaines caractéristiques tel que la puissance de calcul et d'une certaine mémoire d'exécution.

Le périphérique est doté d'une mémoire de stockage de 32 giga dont 8 gigas sont utilisés pour le système. Ce qui nous permet de réserver une quinzaine de gigas pour le stockage des données. Est-ce suffisant ? Non pour un stockage indéfini car à un moment l'équipement atteindra ses limites. Oui pour un stockage temporaire d'une durée mensuelle permettant un suivi local des différents composants environnementaux par un agriculteur. Dans ce cas de figure les données datant d'un mois seront répliquées sur un cloud ou un serveur à distance et enlever de la passerelle ou bien la récupération peut se faire en local avec une machine.

Enfin la prédiction de l'humidité du sol de par les paramètres météorologiques nous permettra de pouvoir automatiser l'irrigation des champs. Mais de plus faire abstraction du déploiement de certains capteurs en les diminuant et en introduisant la notion de capteur virtuel permettant de à partir de certaines variables, estimer la valeur d'un autre paramètre sans pour autant faire l'usage de capteur.

En résumé, ce système pourrait être utilisé dans des endroits dépourvus d'internet et d'électricité et qui pourrait marcher avec des panneaux solaires et une station météo permettant ainsi aux acteurs agricoles de pouvoir augmenter leur production tout en optimisant leur ressources le tout à faible coût.

### Perspectives

Dans la mesure de mettre en place un modèle beaucoup plus performant, il serait bien de continuer les expérimentations dans un espace beaucoup plus élargi en déployant des capteurs et des stations météo sur une grande étendue et recenser beaucoup de données sur une longue durée. Ceci mettra à disposition un jeu de données assez riche pour faire une étude approfondie sur la relation entre variables de la station météo et l'humidité du sol.

Nos modèles nous permettent à partir de données de la station météo de prédire l'humidité du sol lorsque le sol n'est pas irrigué. Il serait bien de suivre l'évolution de l'humidité du sol en temps d'irrigation grâce à un modèle Machine Learning. Ainsi nous pourrions mettre en place un système complexe de modèles qui gèrera la prédiction de l'humidité du sol en temps d'irrigation et en temps de non irrigation en fixant des seuils.

Il serait intéressant de pouvoir évaluer les limites de la raspberry en y entraînant des modèles beaucoup plus robustes. De plus, le Machine Learning progressif est une méthode d'apprentissage continu qui fait usage de nouvelles données en entrées pour rendre le modèle plus performant en découvrant d'autres composants influant sur la variation de la variable. Nous pourrions étudier la faisabilité de ce dernier dans notre contexte car les données possèdent un caractère saisonnier (données météorologiques d'été différents de celui d'hiver)

Enfin la passerelle est un périphérique limité en ressources et en puissance d'exécution, d'où vient l'idée d'optimiser les ressources avec différentes techniques permettant une longue durée de vie et une conservation énergétique durable.

## Référence:

- [1] Les enjeux et les défis de l'Internet des Objets (IdO), [https://www.researchgate.net/publication/316469870\\_Les\\_enjeux\\_et\\_les\\_defis\\_de\\_l'Internet\\_des\\_Objets\\_IdO](https://www.researchgate.net/publication/316469870_Les_enjeux_et_les_defis_de_l'Internet_des_Objets_IdO).
- [2] Edge Computing : l'informatique en périphérie, une révolution pour les années 2020, [lien](#)
- [3] Internet of Things(IoT), Techsparks, [lien](#)
- [4] F. Samie, V. Tsoutsouras, S. Xydis, L. Bauer, D. Soudris, and J. Henkel, "Computation Offloading and Resource Allocation for Low-power IoT Edge Devices," in IEEE WF-IoT, 2016.
- [5] L'edge computing, un outil incontournable dans l'industrie 4.0, Célia Garcia-Montero [lien](#)
- [6] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, "A Survey on the Edge Computing for the Internet of Things," IEEE Access, vol. 6, pp. 6900–6919, 2018.
- [7] Intelligence Artificielle, Yann LeCun, [lien](#) : [https://www.college-de-france.fr/media/yann-lecun/UPL4485925235409209505\\_Intelligence\\_Artificielle\\_\\_\\_\\_Y.\\_LeCun.pdf](https://www.college-de-france.fr/media/yann-lecun/UPL4485925235409209505_Intelligence_Artificielle____Y._LeCun.pdf).
- [8] Machine Learning, Observatoire de Paris – LESIA – Service d'Informatique Scientifique, Emmanuel Grolleau,, [https://sites.lesia.obspm.fr/emmanuel-grolleau/files/2017/08/SIS\\_Presentation\\_MachineLearning\\_Grolleau-1.pdf](https://sites.lesia.obspm.fr/emmanuel-grolleau/files/2017/08/SIS_Presentation_MachineLearning_Grolleau-1.pdf)
- [9] From Cloud Down to Things: An Overview of Machine Learning in Internet of Things, Farzad Samie, Lars Bauer, and Jörg Henkel, Fellow, IEEE, [lien](#) : [https://www.researchgate.net/publication/330458771\\_From\\_Cloud\\_Down\\_to\\_Things\\_An\\_Overview\\_of\\_Machine\\_Learning\\_in\\_Internet\\_of\\_Things](https://www.researchgate.net/publication/330458771_From_Cloud_Down_to_Things_An_Overview_of_Machine_Learning_in_Internet_of_Things).
- [10] Y. Yue, X. Cheng, D. Zhang, Y. Wu, Y. Zhao, Y. Chen, G. Fan, and Y. Zhang, "Deep recursive super resolution network with laplacian pyramid for better agricultural pest surveillance and detection," Computers and Electronics in Agriculture, 2018.
- [11] E. Mwebaze and G. Owomugisha, "Machine learning for plant disease incidence and severity measurements from leaf images," in International Conference on Machine Learning and Applications (ICMLA), 2016.

- [12] S. S. Patil and S. A. Thorat, "Early detection of grapes diseases using machine learning and IoT," in the International Conference on Cognitive Computing and Information Processing (CCIP), 2016.
- [13] S. Yahata, T. Onishi, K. Yamaguchi, S. Ozawa, J. Kitazono, T. Ohkawa, T. Yoshida, N. Murakami, and H. Tsuji, "A hybrid machine learning approach to automatic plant phenotyping for smart agriculture," in International Joint Conference on Neural Networks (IJCNN), 2017.
- [14]. Chen XF, Wang ZM, Wang ZL, Li R. Drought evaluation and forecast model based on soil moisture simulation. *China Rural Water and Hydropower*, 2014(05): 165–169.
- [15]. Jackson SH. Comparison of calculated and measured volumetric water content at four field sites. *Agricultural water management*, 2003, 58(3): 209–222.
- [16]. Hummel JW, Sudduth KA, Hollinger SE. "Soil moisture and organic matter prediction of surface and subsurface soils using an NIR soil sensor." *Computers and electronics in agriculture* 32.2 (2001): 149–165.
- [17]. Shu SF, Qian HF, Qiu XW. Soil moisture forecast model based on meteorological factors in Jinhua City. *Chinese Journal of Agrometeorology*, 2009, 30(02):180–184.
- [18]. Hou XL, Feng YH, Wu GH, He YX, Chang DM. Application research on artificial neural network dynamic prediction model of soil moisture. *Water Saving Irrigation*, 2016(07):70-72+76.
- [19]. Ji RH, Zhang SL, Zheng LH, Liu QX. Prediction of soil moisture based on multilayer neural network with multi-valued neurons. *Transactions of the Chinese Society of Agricultural Engineering*, 2017, 33(S1): 126–131.
- [20]. Gill MK, Asefa T, Kemblowski MW, Mckee M. "Soil moisture prediction using support vector machines 1." *JAWRA Journal of the American Water Resources Association* 42.4 (2006): 1033–1046.
- [21]. Soil Moisture Estimation With SVR and Data Augmentation Based on Alpha Approximation Method.

- [22]. Li N, Zhang Q, Yang FX, Deng ZL. Research of adaptive genetic neural network algorithm in soil moisture prediction. *Computer Engineering and Applications*, 2018, 54(01): 54–59+69.
- [23]. Research of Soil Moisture Content Forecast Model Based on Genetic Algorithm BP Neural Network, Caojun Huang, Lin Li, Souhua Ren, and Zhisheng Zhou.
- [24]. Dynamic Neural Network Modelling of Soil Moisture Content for Predictive Irrigation Scheduling, Olutobi Adeyemi, Ivan Grove, Sven Peets, Yuvraj Domun and Tomas Norton.
- [25] Soil moisture forecasting by a hybrid machine learning technique: ELM integrated with ensemble empirical mode decomposition Ramendra Prasad, Ravinesh C. Deo, Yan Li, Tek Maraseni.
- [26]. Veres M, Lacey G, Taylor G W. Deep learning architectures for soil property prediction. *Computer and Robot Vision (CRV)*, 2015 12th Conference on. IEEE, 2015: 8–15.
- [27]. Wang JR, Chen TJ, Wang YB, Wang LS, Xie CJ. Soil near-infrared spectroscopy prediction model based on deep sparse learning. *Chinese Journal of Luminescence*, 2017, 38(01): 109–116.
- [28]. Research on soil moisture prediction model based on deep learning, Yu Cai, Wengang Zheng, Xin Zhang, Lili Zhangzhong, Xuzhang Xue, <https://doi.org/10.1371/journal.pone.0214508>.
- [29] Edge Machine Learning: Enabling Smart Internet of Things Applications  
Mahmut Taha Yazici ID , Shadi Basurra ID and Mohamed Medhat Gaber
- [30]. Hinton GE, Osindero S, Teh YW. A fast learning algorithm for deep belief nets. *Neural computation*, 2006, 18(7): 1527–1554. <https://doi.org/10.1162/neco.2006.18.7.1527> PMID: 16764513.
- [31]. <https://www.wazihub.com/>
- [32] <https://zindi.africa/competitions/wazihub-soil-moisture-prediction-challenge/data>
- [33] LoRa, Comment fonctionne le réseau, <https://www.journaldunet.fr/web-tech/dictionnaire-de-l-iot/1197635-lora-comment-fonctionne-le-reseau-queelles-differences-avec-sigfox-20201211/>

[34] An Introduction to Support Vector Regression (SVR), <https://towardsdatascience.com/an-introduction-to-support-vector-regression-svr-a3ebc1672c2>