République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

pour l'obtention du diplôme de Master en Informatique

Option : *Réseaux et Systèmes Distribués (R.S.D)*

Thème

Détection et identification des activités quotidiennes dans les maisons intelligentes

Réalisé par :

BEDRANE Bouchra

Présenté le 04 Juillet 2022 devant le jury composé de :

- *Mr BENMOUNA Youcef* *(Président)*
- *Mr LEHSAINI Mohamed* *(Encadrant)*
- *Mme BENMAHDI Meryem Bochra* *(Examinateur)*
- *Mr KADDOUR Sidi Mohammed* *(co-Encadrant)*

Année universitaire : 2021-2022

Remerciements

Avant tout, je rends grâce à DIEU tout puissant de m'avoir accordé la volonté et le courage pour réaliser ce mémoire.

Au terme de ce travail je tiens tout d'abord à exprimer ma profonde gratitude à mon professeur et encadrant Monsieur LEHSAINI Mohamed Directeur de Laboratoire de Recherche STIC pour son suivi et pour son énorme soutien qu'il n'a cessé de nous orienter tout au long de la période du projet.
Nous tenons à remercier également mon co-encadrant Monsieur KADDOUR Sidi Mohamed doctorant à l'université Abou Baker Belkaid pour le temps qu'il a consacré et pour les précieuses informations qu'il nous a accordé avec intérêt et compréhension.

Mes remerciements s'adressent à tous les membres du jury
pour l'honneur qu'ils m'ont fait en acceptant
de juger ce travail.

Dédicaces

Je dédie ce mémoire

À mes très chers parents

Pour tous leurs sacrifices, leur tendresse, leur soutien tout au long de mes études

A mon adorable sœur Ahlem, je ne pourrais jamais exprimer l'attachement et l'affection que j'ai pour vous. Aucun mot ne pourrait exprimer la gratitude et l'amour que je vous porte

À toute ma famille et mes amis.

Table des matières

Liste des figures

Liste des Tableaux

Liste des abréviations

AVQ	**A**ctivité **V**ie **Q**uotidienne
EnR	**En**ergies **R**enouvelables
DEG	**D**istributed **E**nergy **G**eneration
TIC	**T**echnologies **I**nformation **C**omunication
IA	**I**ntelligence **A**rtificial
BEMS	**B**uilding **E**nergy **M**anagement **S**ystem
NLP	**N**aturel **L**anguage **P**rocessing
DBSCAN	**D**ensity **B**ased **S**patial **C**lustering **A**pplication **N**oise
ETL	**E**xtraction, **T**ransformation, **L**oading

Résumé

Dans le cadre de ce projet de fin d'études, nous avons proposé d'utiliser deux méthodes de classification pour détecter les activités quotidiennes au sein des maisons intelligentes. Ceci dans le but est de connaitre la quantité d'électricité nécessaire pour répondre aux besoins des occupants de ces maisons intelligentes et éviter ainsi une production d'électricité qui dépasse leurs besoins c'est à dire créer une adéquation entre l'offre et la demande en terme de consommation d'électricité. Il s'agit de deux méthodes qui font parties des méthodes d'apprentissage automatique non supervisé : K-Means et DBSCAN. Les résultats obtenus sur un ensemble de données ont montré que DBSCAN fournit de bonnes performances comparée à K-Means sur une variété de distributions différentes.

Mots clés : Plateforme KNIME, Python, Réseaux de capteurs, K-Means, DBSCAN, Apprentissage non supervisé

Abstract

In this final year study projects, we have proposed to use two classification methods to detect the daily activities in smart homes. The goal is to know the amount of electricity needed to meet the needs of the occupants of these smart homes and thus avoid a production of electricity that exceeds their needs, i.e. to create a match between supply and demand in terms of electricity consumption. The two methods are part of the unsupervised machine learning methods: K-Means and DBSCAN. The results obtained on a dataset have shown that DBSCAN provides good performances compared to K-Means on a variety of different distributions.

Key words: KNIME platform, Python, Sensor networks, K-Means, DBSCAN, Unsupervised learning

ملخص

في إطار مشروع التخرج، اقترحنا استخدام طريقتين تصنيف للكشف عن الأنشطة اليومية في المنازل الذكية .وذلك لمعرفة كمية الكهرباء اللازمة لتلبية احتياجات ساكني هذه المنازل الذكية وبالتالي تجنب إنتاج كمية الكهرباء التي تتجاوز احتياجاتهم، أي خلق توازن بين العرض والطلب من حيث استهلاك الكهرباء .هاتان طريقتان تشكلان جزءًا من طرق التعلم الآلي غير الخاضعة للإشراف :K-Means و DBSCAN. أظهرت النتائج التي تم الحصول عليها على مجموعة البيانات أن أداء DBSCAN جيدًا مقارنةً بـ K-Means على مجموعة متنوعة من التوزيعات المختلفة.

الكلمات المفتاحية: منصة KNIME، Python ، شبكات الاستشعار، K-Means, DBSCAN، تعليم غير مشرف عليه

Introduction générale

Introduction générale

Dans le marché d'électricité, la connaissance des consommateurs d'électricité fournit une compréhension de leur comportement de consommation, qui est récemment devenu important dans l'industrie électrique. Avec cette connaissance, les fournisseurs d'électricité sont capables de développer une nouvelle stratégie commerciale et d'offrir des services basés sur la demande des clients.

La méthode la plus efficace actuellement pour réduire les pertes commerciales est d'utiliser des compteurs électroniques intelligents. Ces compteurs peuvent aviser les consommateurs en tout moment si sa consommation dépasse un certain seuil.

Le but de ce projet est de développer une approche pour la classification des activités de la vie quotidienne (AVQ) dans les maisons intelligentes et de connaitre leur consommation en termes d'énergie électrique. Ces informations permettent aux fournisseurs d'électricité de produire des quantités suffisantes pour répondre aux besoins des consommateurs et en même temps éviter de produire des quantités supérieures à leurs besoins pour ne pas gaspiller cette ressource précieuse. Pour ce faire, nous avons proposé deux méthodes de classification : K-Means et DBSCAN pour connaitre les activités quotidiennes des occupants des habitats et ceci pour mettre en adéquation l'offre et la demande en matière d'électricité.

Ce mémoire est structuré en quatre chapitres :

- Le premier chapitre présente des notions générales sur les smart grids (les réseaux intelligents) et le système de la gestion de l'énergie.
- Le deuxième chapitre est une simple description des méthodes de machine learning (l'apprentissage automatique) et de quelques algorithmes de clustering.
- Le troisième chapitre est une présentation des outils logiciels nécessaires pour le développement de notre application tels que la plateforme KNIME, les bibliothèques Python utilisées dans l'apprentissage automatique et la base de données ainsi que les méthodes implémentées (K-Means, Fuzzy c-Means, Random forest, DBSCAN).
- Le quatrième chapitre présente l'application développée dans le cadre de notre projet de fin d'études où une classification des activités quotidiennes a été faite par les deux

méthodes (K-Means et DBSCAN) suivie d'une comparaison entre elles en termes de précision de détection des activités issues d'une base de données (dataset).

Enfin, on clôture par une conclusion dans laquelle nous rappelons l'objectif du projet de fin d'étude et nous décrivons les résultats obtenus ainsi que les quelques perspectives.

Chapitre I

Réseaux électriques intelligents

(Smart Grids)

Chapter I : Les réseaux électriques intelligents (Smart Grids)

I.1 Introduction

"Smart grid" signifie "réseau intelligent" en littérature. Ainsi comme l'intelligence artificielle a envahi presque tous les domaines et les environnements, que ce soit les maisons dites les maisons intelligentes, les véhicules autonomes, l'industrie, il était également normal qu'à une plus grande échelle, le réseau électrique prenne aussi le sujet. En effet, que ce soit sous la contrainte écologique ou pour mieux faire correspondre l'offre et la demande d'énergie, un changement était absolument indispensable dans ce domaine. Par ailleurs, mieux apprendre à analyser les demandes des clients, y satisfaire leurs attentes, et à mettre en place un réseau mieux adapté, tel est le but des réseaux intelligents (smart grids). Par réseaux intelligents, on entend un réseau énergétique intégrant les technologies de l'information et de la communication (TIC), ce qui contribue à améliorer son fonctionnement et à développer de nouveaux usages tels que l'autoconsommation ou le stockage [1]. Ce type de réseaux permet donc de basculer d'un système de production dépendant de la demande à un système de consommation basé sur l'offre, qui devra dans le futur s'adapter aux variations arbitraires de la production d'énergie que ce soit éolienne ou solaire. Combiné à d'autres technologies telles que le pompage-turbinage ou les centrales à gaz à cycle combiné, ce type de réseaux devrait participer à améliorer la sécurité d'approvisionnement, à diminuer les frais liés au réseau de distribution et au contrôle de l'énergie, à incorporer les énergies renouvelables (EnR) dans le réseau et à perfectionner le rendement de tout le système [2].

Selon la commission européenne, les "smart grids" sont des réseaux électriques intelligents pouvant incorporer avec succès les activités de tous les usagers qui y sont connectés: producteurs, consommateurs, afin de créer un système économique et durable avec de faibles pertes et un niveau élevé de fiabilité d'approvisionnement [3].

La figure I-1 illustre un exemple d'un réseau électrique intelligent à base des smart grids dans lequel une éventuelle adéquation est visée entre les producteurs de l'électricité et les consommateurs.

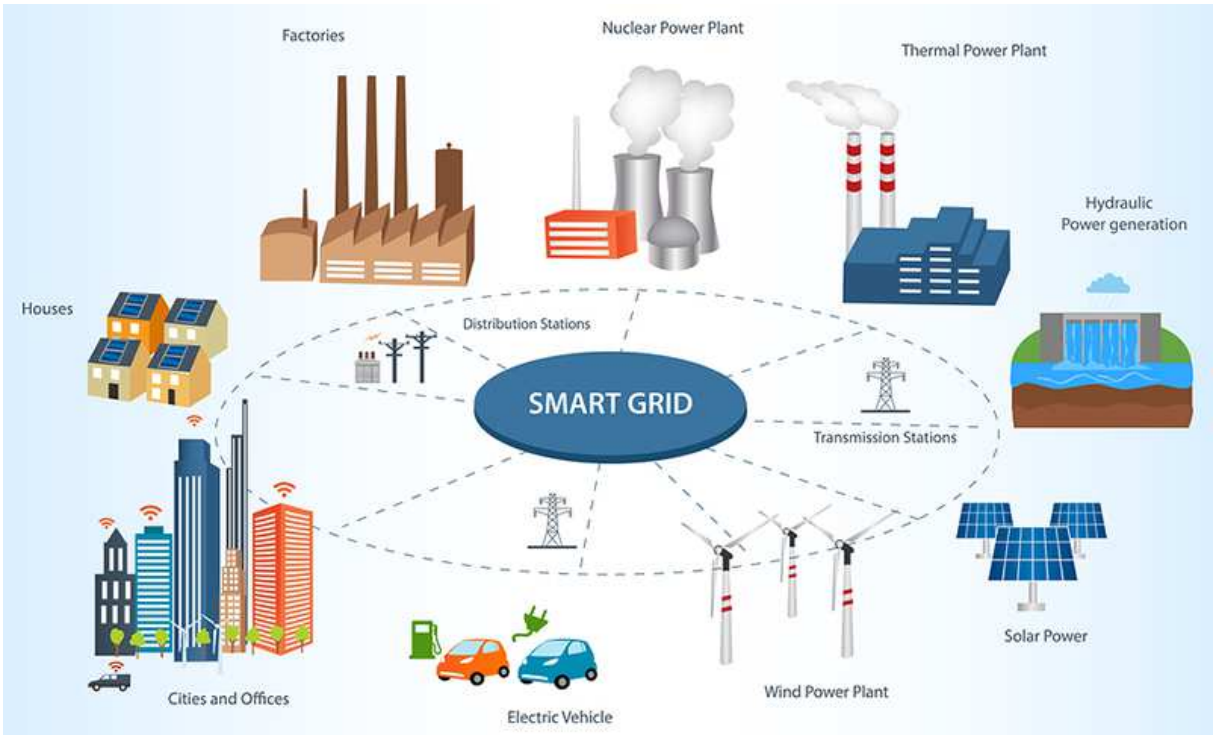

Figure I-1: Smart Grid pour mutualiser les réseaux électriques intelligents [1]

I.2 Le rôle des réseaux intelligents

Les réseaux intelligents permettent de contrôler la consommation d'énergie et de l'optimiser pour le consommateur. Jusqu'à récemment, l'équilibre du système énergétique était essentiellement obtenu en contrôlant l'offre (production) d'énergie en fonction de la demande (consommation), aux meilleures conditions d'offre et de coût. Avec les réseaux intelligents, il devient possible d'adapter la consommation à la production en ciblant une adaptation entre l'offre et la demande, d'où le rôle des "consommateurs" peut intervenir dans ce projet.

Nous désignons par "Smart Grids" un réseau d'énergie qui fait appel aux technologies de l'information et de la communication (TIC), ce qui contribue à améliorer son fonctionnement et à développer de nouveaux usages tels que l'autoconsommation ou le stockage. Les réseaux intelligents apportent des solutions à plusieurs défis posés par les profonds changements du système énergétique, tels que [4]:

- Faciliter la réintégration des énergies renouvelables dans les réseaux électriques. Dans ce contexte, le déploiement des compteurs Linky devrait contribuer à améliorer la connaissance du secteur de la basse tension et permettre l'amélioration de l'optimisation des solutions de raccordement par une meilleure connaissance des activités des usagers (consommateurs).

- Faciliter l'intégration des gaz verts dans les réseaux électriques où les gestionnaires des réseaux de gaz sont censés à développer des rebours pour rendre l'interface entre le réseau de distribution et le réseau de transport bidirectionnelle.
- Encourager le déploiement de la mobilité électrique : il est indispensable de développer la contrôlabilité de la recharge des véhicules électriques afin de fluidifier la demande d'énergie aux heures de pointe, qui créerait des contraintes importantes sur les réseaux. Grâce à Linky, cette contrôlabilité sera possible via le compteur, avec des commandes simples.
- Réaliser des actions de maîtrise de l'énergie et de productivité énergétique : les nouveaux compteurs perfectionnés comme Linky, mais aussi les objets connectés domotiques offrent la possibilité aux consommateurs de contrôler la consommation de leurs appareils énergivores, c'est-à-dire les appareils qui consomment le plus d'énergie comme le fer à repasser, le poste à souder, etc.
- Améliorer le fonctionnement des réseaux : le déploiement de nouvelles sous-stations intelligentes, de compteurs perfectionnés, de capteurs et même d'actionneurs fournit aux gestionnaires de réseaux des informations sur l'état des réseaux et leur permet de les contrôler à distance, ce qui améliore la qualité du service pour les usagers.

I.2.1 Comparaison entre les réseaux intelligents et les réseaux électriques classiques

Les réseaux électriques classiques et les réseaux intelligents se diffèrent sur plusieurs points. Nous pouvons les résumer comme suit :

- Système traditionnel
- Plusieurs décisions de mise en œuvre ont été faites il y a plus de 120 ans.
- Structure hiérarchisée.
- Centrales généralement de grande taille.
- Faible nombre de grandes installations centrales de stockage d'énergie (centrales de pompage).
- Une production d'énergie centralisée et une consommation passive.
- Un système n'est pas performant : la perte de transmission est de près de 20%.
- Contrôle n'est pas efficace du réseau de distribution.
- Utilisation réduite des technologies de l'information et de la communication, notamment dans les réseaux électriques traditionnels.

- Nombreux composants de différentes tailles.
- Intégration d'installations de production décentralisées.
- Intégration de nombreuses petites installations de stockage décentralisées (véhicules électriques).
- Composants plus intelligents.
- Utilisation constante des TIC jusqu'aux consommateurs.

Le tableau I-1 illustre les principales différences entres les réseaux électriques traditionnels et les réseaux électriques intelligents.

Tableau I-1 : Comparaison entre les réseaux électriques actuels et les smart grids [2]

Réseaux électriques actuels	**Réseaux électriques intelligents**
Analogique	Numérique
Unidirectionnel	Bidirectionnel
Production centralisée	Production décentralisée
Communication sur une partie des réseaux	Communication sur l'ensemble des réseaux
Gestion de l'équilibre du système électrique par l'offre/production	Gestion de l'équilibre du système électrique par la demande/consommation

I.3 L'architecture des réseaux intelligents

L'architecture des smart grids est constituée de trois niveaux [4] :

a) Le premier niveau sert à livrer l'électricité et le gaz naturel par le biais d'une infrastructure conventionnelle de structures électriques et de gaz naturel (lignes, transformateurs, etc.).

b) Le deuxième niveau est constitué par un réseau de communication utilisant différents supports et technologies de communication (fibre optique, GPRS, 4/G5G, etc.).

c) Le troisième niveau est composé d'applications et de services, tels que des systèmes de télédépannage ou des dispositifs qui réagissent systématiquement à la demande d'électricité en utilisant des informations en temps réel.

La figure I-2 présente l'architecture d'un réseau électrique intelligent et les trois niveaux de cette architecture.

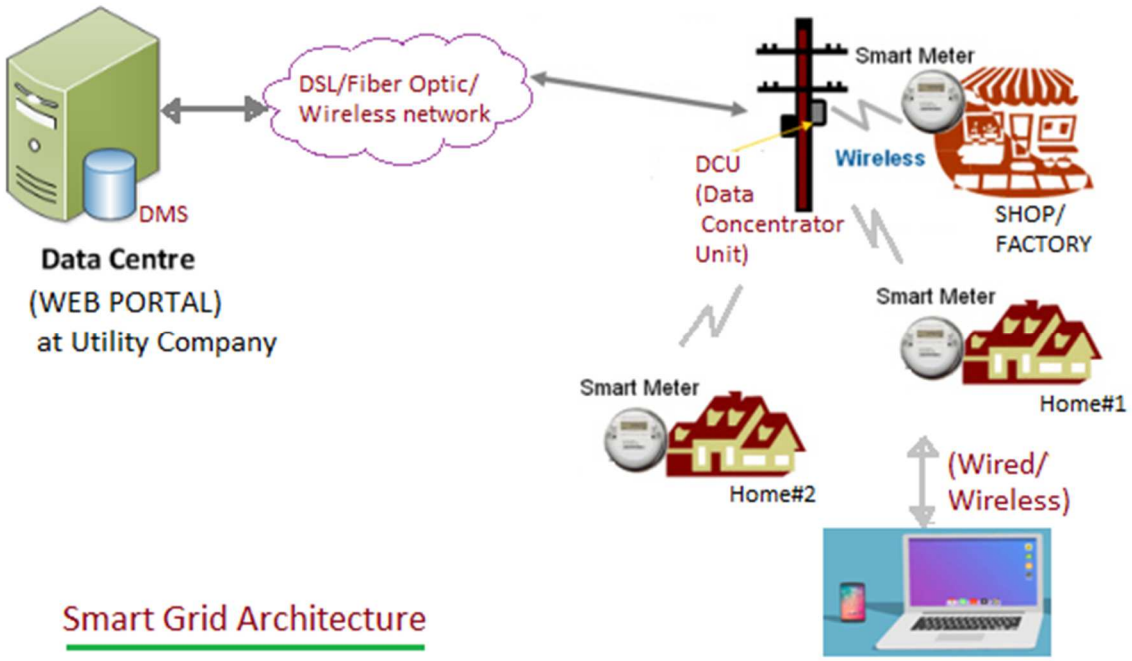

Figure I-2 : L'architecture des réseaux électriques intelligents [5]

I.4 Le fonctionnement des réseaux intelligents

Un réseau intelligent combine l'infrastructure électrique avec les technologies de l'information et de la communication (TIC) pour analyser et communiquer les informations reçues. Cette technologie de communication est impliquée à tous les niveaux du réseau : production, transmission, distribution et consommation, et permet de prendre les bonnes décisions pour démontrer une véritable concordance entre l'offre et la demande [6].

Les objectifs des réseaux intelligents sont nombreux et satisfont à différentes exigences pour faire face au monde d'aujourd'hui et, plus encore, à celui de demain avec de nouvelles contraintes :

- Les exigences d'exploitation : pour que les réseaux puissent répondre à tout moment à la demande des consommateurs sans craindre une panne. Il s'agit d'établir la sûreté de fonctionnement en prévoyant tous les scénarios possibles, même chaotiques dans certains cas.
- Flexibilité : pour que ces mêmes réseaux puissent également s'adapter aux variations de cette demande, selon le jour, l'heure et la période (hiver, été). Il s'agit de mettre en place des dispositifs intelligents qui permettent de passer d'un scénario à un autre avec une grande souplesse.

- Besoin en puissance : pour que les nouvelles productions énergétiques puissent être facilement intégrées au réseau existant. Il s'agit de permettre un passage à l'échelle en facilitant le procédé d'intégration de nouvelles productions.
- Exigence de sobriété économique : afin de réduire les frais de production, les déperditions d'énergie et la consommation. Il s'agit de mettre en place des systèmes qui permettent de produire de l'électricité à un plus faible coût et qui établissent la concordance entre l'offre et la demande de telle sorte que nous n'ayons pas de gaspillage d'énergie et que nous couvrions les besoins des consommateurs.

I.5 Avantages et inconvénients des smart grids

Dans cette section, nous présentons brièvement les avantages et les inconvénients des réseaux électriques intelligents.

a) Avantages

- Amélioration de la fiabilité et de la qualité de l'électricité : Les réseaux intelligents garantissent un approvisionnement en électricité fiable, réduisent le nombre et la durée des pannes, fournissent une électricité plus propre et des systèmes dotés de capacités d'autoréparation.
- Sécurité et cybersécurité accrues : Les réseaux intelligents garantissent une auto-surveillance continue qui leur permet de reconnaître toute situation d'insécurité susceptible d'avoir un impact sur leur sécurité intrinsèque et sur la sécurité de leur activité. Un niveau élevé de sécurité est incorporé dans tous les systèmes et opérations, y inclus la surveillance des installations physiques, la cybersécurité et la protection des données personnelles de tous les utilisateurs.
- Une meilleure efficacité énergétique : Les réseaux intelligents peuvent réduire la consommation totale d'énergie, contrôler la demande pendant les périodes de pointe, diminuer les pertes et encourager les utilisateurs finaux à réduire leur consommation d'électricité plutôt que de compter de manière systématique sur une production plus importante.
- Avantages financiers directs (réduction des coûts d'exploitation, élargissement de l'offre tarifaire) : Les réseaux intelligents fournissent aux opérateurs de réseaux des avantages économiques. Les coûts d'exploitation sont considérablement réduits et le client final pourra bénéficier d'une offre tarifaire étendue dans certaines circonstances.

b) Limitations

- Le coût de réalisation demeure encore élevé à ce jour.
- Les données collectées sont délicates à gérer et à stocker. Dans certains cas, nous avons des données aberrantes qui ne représentent pas vraiment le comportement des occupants d'un habitat ou des données incomplètes dans certains cas.
- Les compteurs communicants utilisés peuvent être cibles à des attaques par des hackers (piratés).
- Problème de normalisation des composants utilisés car les producteurs et fournisseurs d'électricité du monde entier veulent protéger leur identité pour des motifs commerciaux.

I.6 Intérêts des réseaux électrique intelligents

L'électricité ne peut pas être stockée facilement et économiquement en grandes quantités. A cet égard, les technologies utilisées dans les réseaux intelligents visent à ajuster en temps réel la production et la distribution (offre et demande) d'électricité en priorisant les demandes de consommation (quantité et localisation) en fonction de leur urgence afin de [7] :

- Optimiser le rendement des unités de production de l'électricité.
- Éviter de devoir construire constamment de nouvelles lignes de production.
- Réduire le gaspillage d'électricité.
- Favoriser l'insertion de la production décentralisée et réduire ou éradiquer les problèmes causés par la volatilité de certaines sources (énergie solaire, éolienne, marémotrice et, dans une moindre mesure, hydroélectrique) tout en privilégiant des sources d'énergie propres.

I.7 Maison Intelligente

Afin d'être alimenté en énergie par des réseaux intelligents et de pouvoir communiquer avec les îlots de la "ville intelligente" tout en respectant le droit à la vie privée des occupants, le bâtiment ou la maison doit être communicant pour mériter d'être une "maison intelligente". Une première application de bâtiment intelligent a vu le jour aux Etats-Unis dans les années en 1970 sous le nom de "Système de gestion de l'énergie du bâtiment (BEMS)". L'idée de "Smart Home" s'est concrétisée dans les années 1980, avec le déploiement des technologies de l'information et de la communication et leur incorporation dans les réseaux électriques.

D'après F-X. Jeuland [8], une maison intelligente est une maison dotée de fonctionnalités permettant de faciliter la vie quotidienne de ses occupants, d'économiser de l'énergie et de

fournir un certain niveau de commodité et de sécurité. Elle est bien prête pour les évolutions futures par la nature même de son infrastructure de câblage et par son ouverture au monde numérique.

I.8 Conclusion

Dans ce chapitre introductif, nous avons présenté les réseaux électriques intelligents, leurs caractéristiques et les enjeux entre offre et demande dans ce type de réseaux. Nous avons aussi présenté les avantages et les limitations des réseaux électriques intelligents dans les maisons d'aujourd'hui et comment détecter les activités de leurs occupants. Ce type d'informations permet d'instaurer une adéquation entre l'offre et la demande et permet par conséquent d'économiser la production de l'électricité en évitant une production qui dépasse les besoins des consommateurs.

Chapitre II

Concepts fondamentaux de l'Apprentissage automatique

Chapter II : Concepts fondamentaux de l’apprentissage automatique

II.1 Introduction

L'Intelligence Artificielle (IA) est la science dont le but est de faire par une machine des tâches que l'homme accomplit en utilisant son intelligence. Elle consiste à mettre en œuvre un certain nombre de techniques visant à permettre aux machines d'imiter une forme d'intelligence réelle. L'IA se retrouve implémentée dans un nombre grandissant de domaines d'application.

Dans ce chapitre, nous présentons les concepts généraux de l’intelligence artificielle et leur intégration dans différents domaines à l’instar les réseaux électriques intelligents. Puis, nous décrivons les bénéfices associés à l’application de l’intelligence artificielle et sa valeur ajoutée.

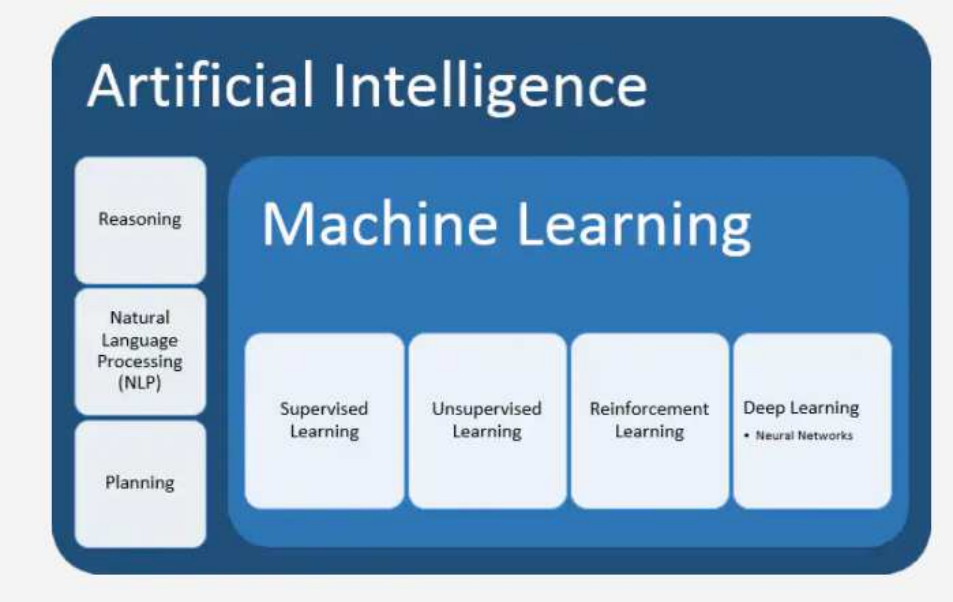

Figure II-1: L'IA engendre l'apprentissage automatique et le traitement du langage naturel [9]

II.2 L’apprentissage automatique (machine Learning)

L'apprentissage automatique, également appelé apprentissage machine ou apprentissage artificiel et en anglais "Machine Learning", est une forme d'intelligence artificielle (IA) qui permet à un système d'apprendre à partir des données et non à l'aide d'une programmation explicite. Cependant, l'apprentissage automatique n'est pas un processus simple. Au fur et à mesure que les algorithmes accusent les données de formation, il devient possible de créer des modèles plus précis basés sur ces données. Un modèle de machine Learning est le résultat

généré lorsque nous entraînons notre algorithme d'apprentissage automatique avec des données. Après la formation, lorsque nous fournissons des données en entrée à un modèle, nous recevons un résultat en sortie. Par exemple, un algorithme prédictif crée un modèle prédictif. Ensuite, lorsque nous fournissons des données au modèle prédictif, nous recevons une prévision qui est déterminée par les données qui ont servi à former le modèle.

L'apprentissage automatique (ML) est une catégorie d'algorithmes qui permet aux applications logicielles de prédire plus précisément les résultats sans être explicitement programmées. Le principe de base de l'apprentissage automatique est de créer des algorithmes capables de recevoir des données d'entrée et d'utiliser une analyse statistique pour prédire une sortie tout en les mettant à jour à mesure que de nouvelles données deviennent disponibles.

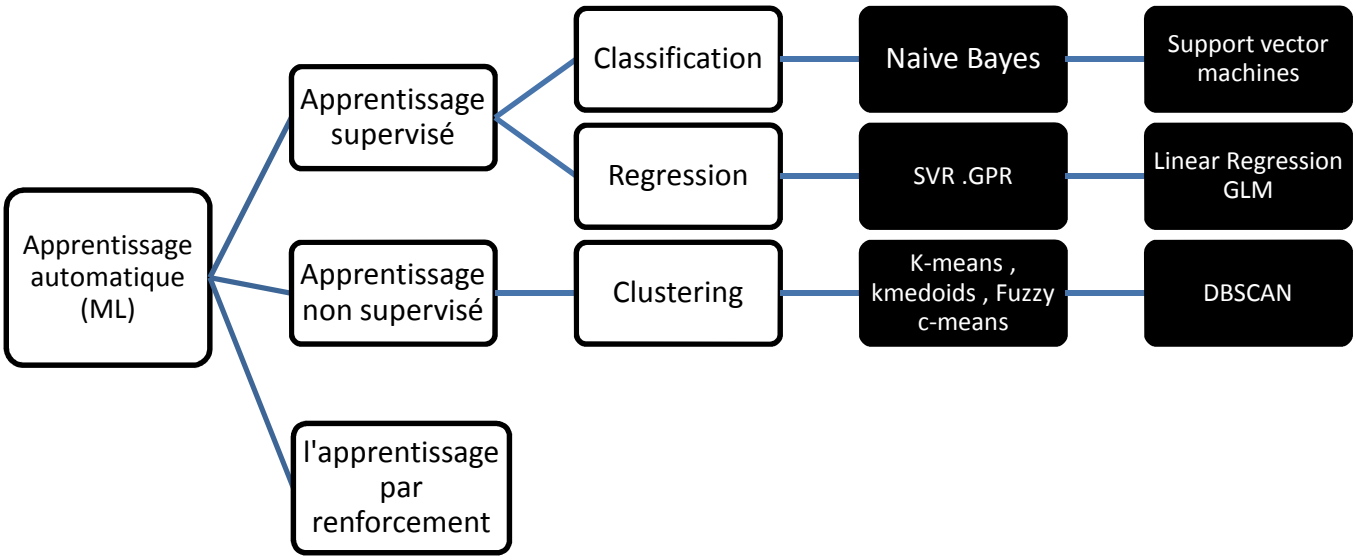

Figure II-2 : Schéma des algorithmes de l'apprentissage automatique

II.3 Intérêts de l'utilisation de l'apprentissage automatique

L'apprentissage automatique permet de solutionner plusieurs types de problèmes. Dans ce qui suit, nous présentons quelques problèmes qui ont été ciblé par l'apprentissage automatique:

- Les problèmes que nous ne les savons pas résoudre tels que le problème de la prédiction des achats.
- Les problèmes que nous savons les résoudre mais nous trouvons des difficultés pour les formaliser algorithmiquement pour les solutionner tels que les problèmes de la reconnaissance d'images et de la voix.

- Les problèmes que nous savons les résoudre mais ils exigent des ressources informatiques très conséquentes à savoir le problème de la prédiction d'interactions entre molécules de grande taille.

L'apprentissage automatique est subséquemment appliqué lorsque les données sont nombreuses, mais que les concepts ne sont pas très accessibles ou pointus. De ce fait, l'apprentissage automatique peut également accompagner les humains à acquérir des connaissances : les modèles générés par les algorithmes d'apprentissage peuvent dévoiler l'importance de certaines informations ou la manière dont elles interférassent les unes avec les autres pour trouver une solution adéquate pour un problème donné. Dans le cas de la prédiction d'achat, la connaissance du modèle peut nous permettre d'analyser quelles caractéristiques des achats passés du client permettent de prédire les achats futurs. Ce concept de l'apprentissage automatique est également appliqué dans d'autres domaines à savoir le domaine médical. Dans ce qui suit, nous présentons quelques exemples de préoccupations auxquelles pourra répondre l'apprentissage automatique [10] :

- Quels sont les gènes qui causent certaines tumeurs ?
- Quelles parties du cerveau sont responsables d'un tel comportement ?
- Quelles sont les propriétés d'une molécule qui en font un bon remède pour une telle maladie telle que le covid ?

L'apprentissage automatique s'appuie sur deux éléments essentiels :

- Les données : qui sont utilisées pour que l'algorithme apprenne.
- L'algorithme d'apprentissage : la démarche qui est exécutée sur ces données pour générer un modèle. Nous appelons "entraînement" le fait d'exécuter un algorithme d'apprentissage sur un ensemble d'exemples ou de données. Ces deux éléments sont très importants car aucun algorithme d'apprentissage ne pourra générer un modèle performant à partir de données qui ne sont pas très bien adaptées.

II.4 Types d'apprentissage automatique

L'apprentissage automatique est un domaine assez vaste. Dans cette section, nous présentons les plus grandes catégories de problèmes auxquels s'intéresse l'apprentissage automatique. Il existe plusieurs techniques d'apprentissage automatique qui sont nécessaires pour améliorer l'exactitude des modèles prédictifs. Ces techniques se distinguent selon la nature

du problème traité, le type et le volume des données. Dans ce qui suit, nous discutons les principales catégories de l'apprentissage automatique.

II.4.1 L'Apprentissage par renforcement

L'apprentissage par renforcement [10] est un modèle basé sur l'apprentissage comportemental. Dans ce modèle, l'algorithme recueille des informations en retour de l'analyse des données et oriente l'utilisateur vers le résultat le plus performant. L'apprentissage par renforcement se distingue des autres types d'apprentissage supervisé en ce sens que le système n'est pas entraîné avec un ensemble de données. En outre, le système apprend par une méthode de tests et d'erreurs. En conséquence, une suite de décisions réussies entraîne le renforcement du processus qui résout le plus efficacement le problème à résoudre.

II.4.2 L'Apprentissage semi-supervisé

L'apprentissage semi-supervisé [10] permet d'apprendre des étiquettes à partir d'un ensemble de données partiellement étiquetées. Le premier de ses avantages est qu'il épargne l'étiquetage de la totalité des exemples d'apprentissage, ce qui est intéressant lorsqu'il est facile de rassembler des données mais que leur étiquetage nécessite un certain effort humain. Par exemple, en classification d'images, il est facile de constituer une base de données comprenant des milliers d'images, mais avoir l'étiquette pour chaque image peut demander beaucoup de travail. En outre, les étiquettes données par des humains sont censées à générer des biais humains, qu'un algorithme totalement supervisé les génère. L'apprentissage semi-supervisé permet dans certains cas de pallier cette limitation.

II.4.3 L'Apprentissage supervisé

L'apprentissage supervisé [10] se déroule typiquement à partir d'un ensemble de données bien déterminé et d'une certaine connaissance de la manière dont ces données sont classées.

L'objectif de l'apprentissage supervisé est de détecter des modèles dans les données et de les associer à un processus d'analyse. Ces données possèdent des propriétés associées à des étiquettes qui permettent de définir leur nature. Par exemple, nous pouvons réaliser une application d'apprentissage automatique qui peut distinguer plusieurs millions d'animaux sur la base d'images et de descriptions écrites.

L'apprentissage supervisé se compose de variables d'entrée (X) et d'une variable de sortie (Y) ; un algorithme pour apprendre la fonction de correspondance (mapping) entre l'entrée et la sortie.

$$Y = f(X)$$

L'objectif est de comprendre la fonction de correspondance si bien que, lorsque nous disposons de nouvelles données d'entrée (X), nous pouvons prédire les variables de sortie (Y) pour ces données. C'est ce qu'on désigne par apprentissage supervisé, car le processus d'un algorithme tiré de l'ensemble des données d'apprentissage peut être considéré comme un enseignant supervisant le processus d'apprentissage. Nous disposons des réponses appropriées, l'algorithme effectue des prédictions itératives sur les données d'apprentissage et est corrigé par l'enseignant. L'apprentissage prend fin lorsque l'algorithme atteint un niveau de performance satisfaisant [11].

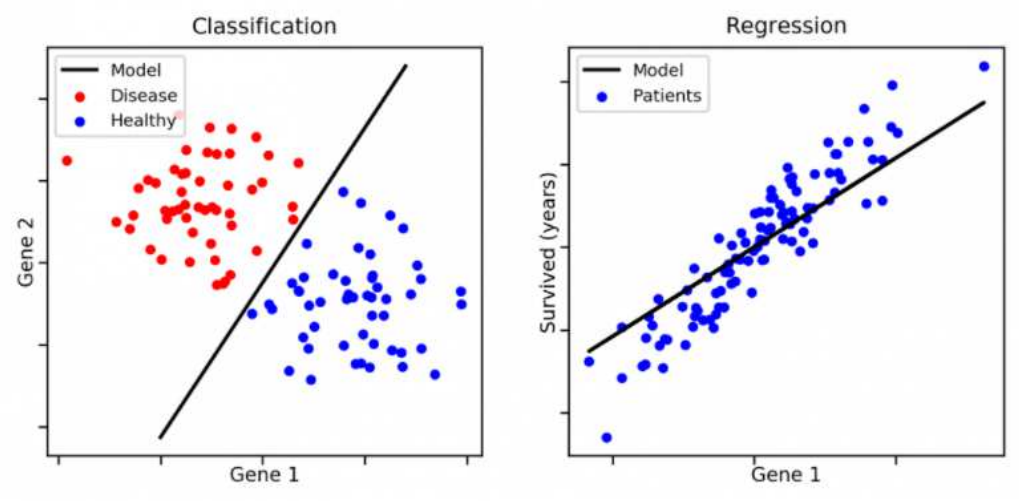

Figure II-3 : La classification et de la régression [11]

Dans l'apprentissage supervisé nous distinguons deux approches comme montre la figure II-3 : la classification et la regression.

a) Classification : Un problème de classification se pose lorsque la variable de sortie est une catégorie, telle que "vert", "blanc" ou "malade" et "n'est pas malade". Dans ce qui suit, nous donnons quelques exemples en relation avec l'approche de classification :

- Dans le secteur bancaire pour détecter s'il y a fraude ou non d'une carte de crédit, nous utilisons la catégorie "fraude" ou "pas fraude".
- La différentiation entre les courriels et les mails indésirables : "spam" ou "pas spam".

- En marketing, nous pouvons analyser les sentiments des clients : "satisfait" ou "pas satisfait".
- Dans le domaine médical, prédire si un patient a une telle maladie ou non par exemple le Covid.

b) Régression : On parle de problème de régression lorsque la variable de sortie est une valeur réelle, telle que "Euros ou Dollars" ou "Taille". Nous citons quelques exemples de cette approche :

- Prédiction du prix d'une maison dans une telle ville et dans un tel endroit.
- Prédiction du prix de baril durant une certaine période.

II.4.4 L'Apprentissage non supervisé

L'apprentissage non supervisé (Unsupervised Learning) [10] est utilisé lorsque le problème requiert une grande quantité de données non étiquetées (unlabeled data) tel que les applications de réseaux sociaux (Twitter, Instagram, …etc). Ainsi, pour donner du sens à ces données, il s'avère nécessaire d'utiliser des algorithmes qui classifient les données en fonction des modèles ou des clusters qu'ils détectent. L'apprentissage non supervisé conduit un processus itératif, analysant les données sans aide humaine. Il est utilisé avec la technologie de détection des pourriels (spams). Les courriels normaux et les spams comportent beaucoup trop de variables pour qu'un analyste puisse étiqueter les spams envoyés en grand nombre. En revanche, des discriminants d'apprentissage automatique, basés sur le regroupement (la mise en clusters) et l'association, sont utilisés pour identifier les courriels indésirables.

L'apprentissage non supervisé consiste à n'avoir que des données d'entrée (X) et aucune variable de sortie correspondante. Son objectif est de modéliser la structure ou la distribution sous-jacente des données afin d'en apprendre davantage sur ces dernières. On parle d'apprentissage non supervisé car, contrairement à l'apprentissage supervisé décrit dans la sous-section précédente, il n'y a pas de réponse correcte ou d'enseignant. Les algorithmes sont laissés à eux-mêmes pour découvrir et présenter la structure intéressante des données. L'apprentissage non supervisé comporte deux catégories d'algorithmes : les algorithmes de regroupement et d'association. La figure II-4 illustre un exemple de l'apprentissage non supervisé.

La mise en clusters consiste à séparer ou à diviser un ensemble de données en un certain nombre de groupes (clusters), de sorte que les données appartenant aux mêmes clusters se ressemblent (ces données ont les mêmes propriétés) davantage que ceux d'autres groupes.

Autrement dit, l'objectif est de séparer les groupes présentant des caractéristiques similaires et de les affecter à des clusters.

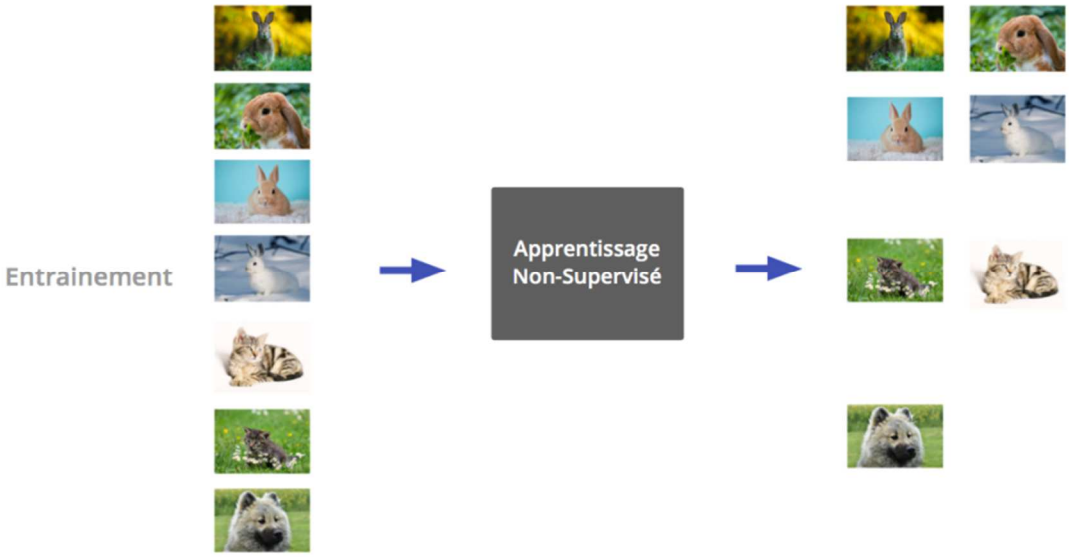

Figure II-4: Regroupement ou Clustering

Pour illustrer la mise en clusters, nous présentons l'exemple suivant : Supposons que nous gérons un magasin de location et que nous souhaitions comprendre les préférences de nos clients pour développer notre activité. Nous proposons de regrouper tous nos clients en huit groupes en fonction de leurs modes d'achat et utiliser une stratégie différente pour les clients de chacun de ces huit groupes. Cette méthode d'analyse et de classification est appelée le "**Clustering"**.

Il existe plusieurs algorithmes d'apprentissage automatique non supervisés qui sont décrit dans la littérature à savoir :

- K-means clustering
- Neural networks (les réseaux de neurones) / Deep Learning (l'apprentissage approfondi)
- Principal Component Analysis (Analyse des composants principaux)
- Distribution models (Modèles de distribution)
- Hierarchical clustering (Classification hiérarchique)

II.5 Apprentissage itératif

L'apprentissage automatique permet aux modèles de s'entraîner sur des données avant d'être utilisés. Il existe des modèles d'apprentissage automatique qui sont en ligne et fonctionnent en continu. Ce processus itératif de modèles en ligne améliore les types d'associations faites entre les éléments de données.

Néanmoins, à cause de leur complexité et de leur taille, ces modèles et ces associations peuvent ne pas être détectés par un superviseur humain. Une fois qu'un modèle a été conçu, il peut être exploité en temps réel pour apprendre à partir des données. L'amélioration de la précision résulte du processus de formation et d'automatisation qui fait partie de l'apprentissage automatique.

II.6 Conclusion

Dans ce chapitre, nous avons présenté l’apprentissage automatique "Machine Learning", ses différentes catégories ainsi que leurs domaines d’application. Cette présentation nous a permis d’avoir une idée sur la catégorie sous-jacente à notre problématique traitée dans le cadre de ce projet de fin d’études.

Dans le chapitre suivant, nous présentons les outils logiciels nécessaires pour aborder notre problématique et nous détaillons certaines notions en relation avec ces outils utilisés.

Chapitre III

Outils logiciels et le dataset utilisés pour le développement de l'application

Chapter III : Outils logiciels et le dataset utilisés pour le développement de l'application

III.1 Introduction

L'apprentissage automatique pourra être programmé sur ordinateurs en utilisant des plateformes bien spécifiques pour alléger la tâche aux chercheurs. Par exemple, *Anaconda* [12] est une distribution qui inclut tous les paquets Python les plus courants, ainsi que de nombreux paquets liés à l'analyse de données et au Big Data. Python possède de nombreuses librairies, qui sont utilisées dans tous les domaines telles que Pandas**,** Matplotlib ou encore Numpy pour tester, explorer les données et les analyser.

Pour développer l'application dans le cadre de notre projet, nous avons utilisé la plateforme analytique KNIME [13] et le langage Python. Nous décrivons dans ce chapitre les différents outils logiciels utilisés pour le développement de notre application.

KNIME (Konstanz Information Miner) est une plate-forme open-source d'analyse, de rapport et d'intégration de données gratuite. KNIME réunit plusieurs composants pour l'apprentissage automatique et la prospection de données par le biais de son concept de pipeline de données modulaire "Building Blocks of Analytics". Une interface utilisateur graphique et l'utilisation de JDBC (Java DataBase Connectivity) permettent de rassembler des nœuds de différentes sources de données, y compris le prétraitement pour la modélisation, l'analyse et la visualisation des données avec une programmation minimale.

KNIME est utilisé dans plusieurs domaines tels que l'exploration de texte, l'économie intelligente, et l'analyse des données financières. Récemment, des contributions ont été faites pour utiliser KNIME comme outil d'automatisation des processus robotiques (RPA) et pour la résolution des problèmes qui impliquent l'intelligence artificielle.

III.2 Outils logiciels utilisés

III.2.1 La plateforme KNIME

KNIME a été développé en Java. Il permet aux usagers de représenter visuellement des flux de données, d'exécuter sélectivement tout ou une partie des étapes d'analyse, puis de contrôler ultérieurement les résultats, les modèles, à l'aide de widgets et de vues interactives. Il permet

également d'ajouter des plugins pour des fonctionnalités additionnelles. La première version de la plateforme KNIME comprend plusieurs modules d'intégration de données (E/S de fichiers, nœuds de bases de données fonctionnant avec tous les systèmes de gestion de bases de données courants par le biais JDBC ou de connecteurs natifs tels que MySQL, Oracle, MS-Access, etc ..), la transformation de données en utilisant des opérateurs qui agissent comme des filtres, convertisseurs, séparateurs, ou combinateurs ainsi que des méthodes couramment utilisées pour les statistiques et l'exploration de données.

KNIME fournit un support de visualisation avec l'extension gratuite "Report Designer". Les flux de travail KNIME peuvent être utilisés comme des ensembles de données pour générer des modèles de rapports qui peuvent être exportés dans des formats de documents tels que doc, ppt, xls, pdf et autres. Dans ce qui suit, nous présentons les principales caractéristiques de la plateforme KNIME :

- L'architecture centrale de KNIME permet le traitement de grands volumes de données qui ne sont limités que par l'espace disque disponible (non limité à la RAM disponible). Par exemple, KNIME permet l'analyse de 300 millions d'adresses clients, 20 millions d'images cellulaires et 10 millions de structures moléculaires.
- KNIME permet le traitement d'un grand nombre de données indépendamment de l'espace disque disponible. Il peut analyser plus de 400 millions informations sur les clients et plus de 30 millions d'images de cellules.
- Des plugins ajoutés à la plateforme KNIME permettent de faire appel à des méthodes d'exploration d'images et d'analyse de réseaux ainsi que d'autres méthodes.
- KNIME permet également d'intégrer d'autres projets open-source tels que les algorithmes d'apprentissage automatique de Weka, le projet R et LIBSVM; ainsi que JFreeChart, ImageJ et Chemistry Development Kit.
- KNIME a été développé et cela n'empêche pas les wrappers de faire appel à d'autres codes et de fournir des nœuds permettant d'exécuter Java, Python et d'autres parties du code.

La figure III-1 illustre l'interface de la plateforme KNIME et les différents éléments manipulés par cette plateforme.

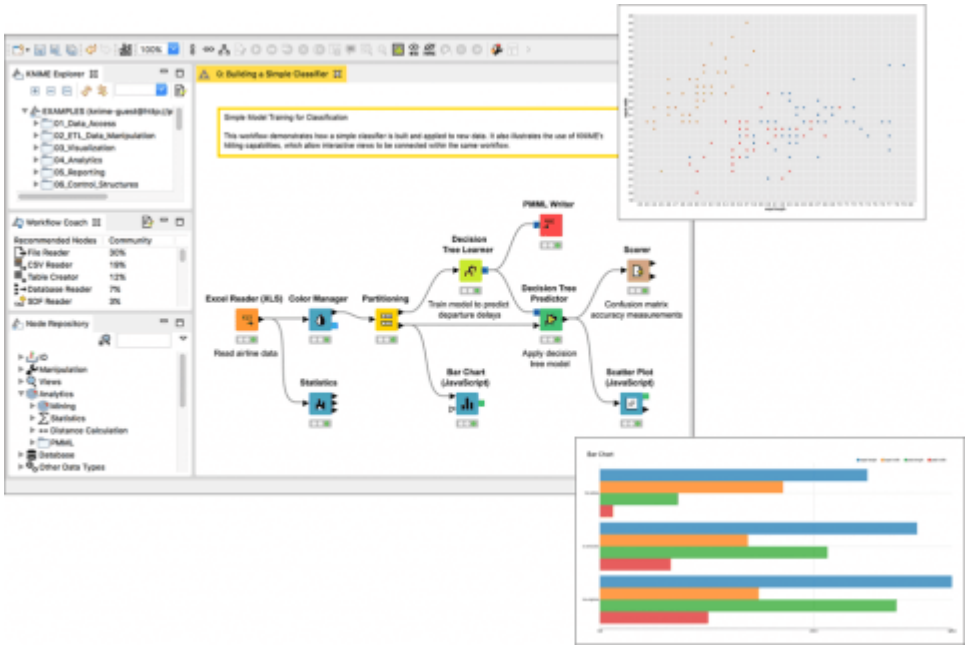

Figure III-1: Interface de la plateforme KNIME [13]

III.3 Les bibliothèques Python pour l'apprentissage automatique

III.3.1 La bibliothèque Skikit-learn

Scikit-learn est une bibliothèque libre de Python destinée à l'apprentissage automatique. Elle est développée par de nombreux contributeurs notamment dans le monde académique par des instituts français d'enseignement supérieur et de recherche comme INRIA et Télécom ParisThec. Elle comprend des fonctions pour estimer des forêts aléatoires, des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support. Elle est conçue pour s'harmoniser avec d'autres bibliothèques libres Python, notamment Numpy et SciPy.

```
from sklearn import datasets
```

Scikit-learn présente une interface concise et cohérente avec les algorithmes d'apprentissage automatique courants, facilitant ainsi l'introduction du ML dans les systèmes de production. La bibliothèque combine un code de qualité et une bonne documentation, une facilité d'utilisation et des hautes performances. Elle constitue de facto le standard de l'industrie pour l'apprentissage automatique avec Python.

III.3.2 La bibliothèque Pandas

Pandas est une bibliothèque Python populaire pour l'analyse de données. Cette bibliothèque n'est pas directement liée à l'apprentissage automatique. Comme nous savons que le jeu de données doit être préparé avant la formation. Dans ce cas, les pandas sont pratiques car ils ont été développés spécifiquement pour l'extraction et la préparation de données. Le package Pandas

fournit des structures de données de haut niveau et de nombreux outils pour l'analyse des données. Il fournit également de nombreuses méthodes intégrées pour tâtonner, combiner et filtrer les données.

```
import pandas as pd
```

Pandas est un package Python conçu pour fonctionner avec des données "étiquetées" et "relationnelles" simples et intuitives. Les modules de pandas constituent un bon outil pour la gestion de données. Il a été conçu pour une manipulation, une agrégation et une visualisation rapides et faciles des données.

III.4 La base de données (Dataset)

III.4.1 Collection des données

Cet ensemble de données concerne les enregistrements par capteurs des activités effectuées par un seul utilisateur dans une maison dite intelligente. Les capteurs utilisés comprennent l'infrarouge passif, les résistances de détection de force, les interrupteurs Reed, les mini capteurs de lumière à cellule photoélectrique, la température et l'humidité et les prises intelligentes. Les données capturées incluent les interactions de l'utilisateur avec l'environnement, telles que les mouvements à l'intérieur, la pression exercée sur le lit ou le canapé, l'utilisation de la cuisinière, de la télévision et du réfrigérateur, ou l'utilisation d'appareils électriques, tels que la cafetière, le lave-vaisselle, la machine à laver, la machine à sandwich ou le micro-ondes. L'ensemble de données peut être utile dans plusieurs domaines, notamment l'analyse de différentes méthodes, par exemple, des algorithmes basés sur les données pour la reconnaissance d'activités ou la reconnaissance d'habitudes. L'ensemble de données contient trois fichiers "csv", à savoir *sensor*, *sensor_sample_int* et *sensor_sample_float*. Le fichier *sensor* contient des informations, tels que le type de mesure du capteur ou le nom du capteur. Le fichier *sensor_sample_int* contient des mesures de capteur de type de données entier tandis que le fichier *sensor_sample_float* contient des mesures de capteur de type de données *float* [16].

III.4.2 Environnement de l'étude de cas

L'appartement est divisé en un salon, une cuisine, une chambre, une salle de bain, un couloir et un balcon. Il contient plusieurs instruments et objets utilisés par le résident, qui pourraient fournir des informations utiles sur l'autonomie fonctionnelle de l'occupant. Ceux-ci comprennent un lit, un canapé, une télévision, un réfrigérateur et plusieurs appareils électriques tels qu'une

cafetière, une machine à sandwich, un lave-vaisselle et autres. Une vue picturale de la disposition de l'appartement est montrée sur la figure III-2.

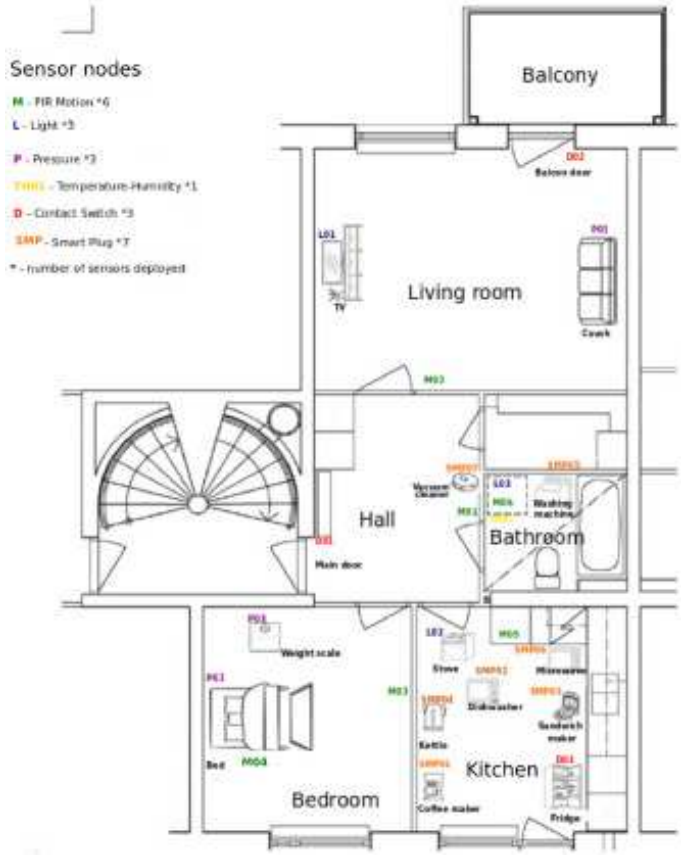

Figure III-2: Répartition des capteurs dans l'environnement domestique expérimental [14]

III.5 Description des méthodes de l'apprentissage automatique utilisées

III.5.1 K-Means

Le Clustering est l'une des techniques d'analyse de données exploratoires les plus courantes utilisées pour obtenir une intuition sur la structure des données. Cela peut être défini comme la tâche d'identifier les sous-groupes dans les données, de sorte que les points de données d'un même sous-groupe appelé aussi cluster soient très similaires, alors que les points de données de différents clusters sont très différents. Par ailleurs, la mise en clusters est considérée comme une méthode d'apprentissage non supervisée, car nous n'avons pas la vérité sur le terrain pour comparer le résultat de l'algorithme de mise en clusters aux libellés réels pour évaluer ses performances. Nous voulons seulement essayer d'étudier la structure des données en regroupant les points de données en sous-groupes distincts.

L'algorithme K-Means est un algorithme itératif qui tente de partitionner les données en *K* sous-groupes distincts prédéfinis, ne se chevauchant pas, dans lesquels chaque point de données

appartient à un seul groupe. Il essaie de rendre les points de données inter-cluster aussi semblables que possible tout en gardant les clusters aussi différents (aussi loin que possible). Il attribue des points de données à un cluster de sorte que la somme de la distance au carré entre les points de données et le centre de gravité du cluster (moyenne arithmétique de tous les points de données appartenant à ce cluster) soit minimale. Moins il y a de variations dans les clusters, plus les points de données sont homogènes (similaires) dans le même cluster. Cependant, l'inconvénient de cette approche est que nous devons dès le début fixer le nombre de clusters et les centres de ces clusters sont choisis aléatoirement [18].

D'après la bibliothèque scikit-learn la complexité moyenne de la méthode K-Means est donnée par ***O(k*n*T),*** où ***n*** est le nombre d'échantillons, ***T*** est le nombre d'itérations, et ***K*** est le nombre des clusters.

III.5.2 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

DBSCAN est un algorithme d'apprentissage automatique non supervisé. Des algorithmes d'apprentissage automatique non supervisés sont utilisés pour classer les données non étiquetées. En d'autres termes, les échantillons utilisés pour former notre modèle ne sont pas accompagnés de catégories prédéfinies comparé à d'autres algorithmes de clustering.

L'algorithme DBSCAN est synonyme de regroupement spatial basé sur la densité d'applications avec bruit. Il est capable de trouver des clusters de forme arbitraire et des clusters avec bruit (c'est-à-dire des valeurs aberrantes) [17].

Dans la bibliothèque scikit-learn, l'implémentation calcule en blocs toutes les requêtes de voisinage, ce qui augmente la complexité de la mémoire à O(n*d) où d est le nombre moyen de voisins, tandis que DBSCAN d'origine avait une complexité de mémoire O(n). Cela peut causer une complexité de mémoire plus élevée lors de l'interrogation de ses voisins les plus proches, selon l'algorithme.

La figure III-3 présente le fonctionnement de l'approche DBSCAN.

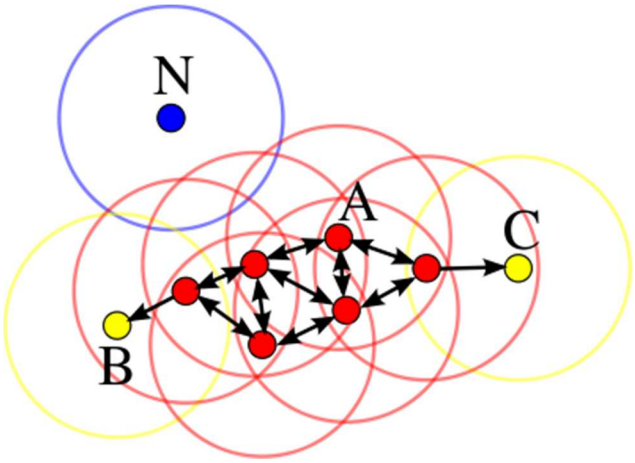

Figure III-3 : Fonctionnement de DBSCAN [15]

L'idée principale derrière DBSCAN est qu'un point appartient à un cluster s'il est proche de nombreux points de ce cluster. Il existe deux paramètres clés dans DBSCAN :

- **eps** : La distance qui spécifie les voisinages. Deux points sont considérés comme voisins si la distance qui les sépare est inférieure ou égale à eps.
- **minPts** : nombre minimal de points de données pour définir un cluster.

Sur la base de ces deux paramètres, les points sont classés comme point central, point frontière ou point aberrant :

- **Point central (Core point)** : un point est considéré comme un point central s'il y a au moins un nombre minPts de points (y compris le point lui-même) dans sa zone environnante avec un rayon eps.
- **Point frontière (Border point)** : un point est un point frontière s'il est accessible à partir d'un point central et s'il y a moins de minPts de points dans sa zone environnante.
- **Valeur aberrante (Noise point)** : un point est une valeur aberrante s'il ne s'agit pas d'un point central et n'est accessible à partir d'aucun point central [19].

La figure III-4 présente les trois types de points dans DBSCAN en illustrant la différence entre ces points.

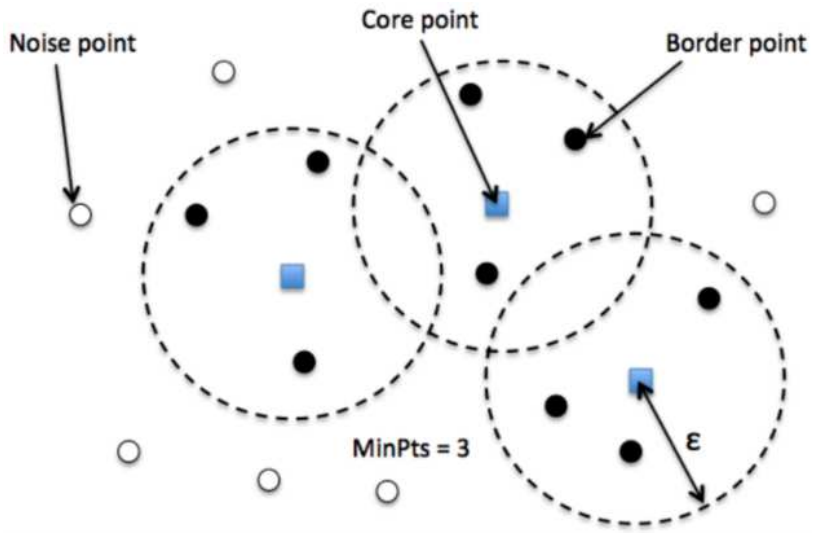

Figure III-4 : Trois types de points dans l'algorithme DBSCAN

III.6 Conclusion

Dans ce chapitre, nous avons présenté les différents outils logiciels nécessaires pour le développement de notre application qui consiste à détecter les activités dans la consommation de l'électricité au sein des maisons telles les bibliothèques de Python et la plateforme KNIME.

Le chapitre suivant fait l'objet de l'application développée en montrant comment ces outils ont été impliqués pour détecter les activités quotidiennes des occupants d'une maison intelligente.

Chapitre IV

Détection des activités à domicile avec une approche non supervisée

Chapter IV : Détection des activités à domicile avec des approches non supervisées

IV.1 Introduction

La détection des activités quotidiennes des occupants d'une maison ou d'un appartement permet de produire l'électricité selon les besoins des consommateurs. Ceci permet de créer une adéquation entre l'offre et la demande et éviter de produire plus que la demande par les producteurs d'électricité d'une part et d'autre part éviter tout manquement dans l'approvisionnement des clients en électricité. Néanmoins, cette opération de détection des activités des occupants d'une maison doit se faire d'une manière très précise pour éviter toute anomalie entre l'offre et la demande. Dans cette optique, nous proposons d'utiliser deux approches pour réaliser l'opération de détection des activités quotidiennes des occupants d'une maison qui est équipée de certains appareils. Ces deux techniques sont : K-Means [16] et DBSCAN [15].

Dans ce chapitre, nous décrivons la démarche suivie pour atteindre cet objectif et les différents éléments nécessaires pour réaliser une application qui répond au cahier de charges de ce projet de fin d'études.

IV.2 Environnement du développement

Pour réaliser notre application, nous présentons tout d'abord le partitionnement de données de consommation énergétique dans les maisons concernées par l'étude. Ce type de maisons contient un ensemble d'équipements tels que la machine à laver, le lave-vaisselle, la TV, etc … Pour cela, nous avons déployé un ensemble de données pour décrire la consommation d'énergie pour chaque équipement électrique et deux méthodes d'apprentissage automatique non supervisé pour classifier les activités à domicile. La figure IV-1 représente le fonctionnement de l'application que nous avons développée pour modifier la description des données

File Table - 3:9 - File Reader (deprecated)

File Edit Hilite Navigation View

Table "res.csv" - Rows: 35603974 | Spec - Columns: 3 | Properties | Flow Variables

Row ID	sensor_id	timestamp	value
18730542	5887	2020-02-26 00:00:00.107997	1024
18730548	6127	2020-02-26 00:00:00.485394	1024
18730560	5887	2020-02-26 00:00:01.109460	1024
18730566	6127	2020-02-26 00:00:01.479707	1024
18730579	5887	2020-02-26 00:00:02.111798	1024
18730584	6127	2020-02-26 00:00:02.488518	1024
18730598	5887	2020-02-26 00:00:03.120853	1024
18730603	6127	2020-02-26 00:00:03.484396	1024
18730616	5887	2020-02-26 00:00:04.128871	1024
18730622	6127	2020-02-26 00:00:04.484097	1024
18730634	5887	2020-02-26 00:00:05.111736	1024
18730640	6127	2020-02-26 00:00:05.485489	1024
18730652	5887	2020-02-26 00:00:06.139620	1024
18730659	6127	2020-02-26 00:00:06.485813	1024
18730671	5887	2020-02-26 00:00:07.125696	1024
18730678	6127	2020-02-26 00:00:07.486244	1024
18730689	5887	2020-02-26 00:00:08.122883	1024
18730697	6127	2020-02-26 00:00:08.487727	1024
18730708	5887	2020-02-26 00:00:09.116760	1024
18730715	6127	2020-02-26 00:00:09.488304	1024
18730726	5887	2020-02-26 00:00:10.118216	1024
18730734	6127	2020-02-26 00:00:10.489230	1024
18730745	5887	2020-02-26 00:00:11.120128	1024

Figure IV-1 : Échantillon d'enregistrements de la table sensor_sample_int

IV.2.1 Prétraitement des données "Pre-Processing"

L'une des premières tâches de l'analyse de données consiste à extraire uniquement certains enregistrements disponibles dans l'ensemble de données brutes. La fréquence de prélèvement est réduite de quelques échantillons par milliseconde à un échantillon par minute. Les enregistrements de tous les équipements sont introduits dans la même colonne. Pour cela, dans cette étape, on modifie la structure de données, où les enregistrements d'une seule colonne vont représenter la consommation d'un équipement spécifique et tout cela en utilisant les fonctionnalités de la plateforme KNIME [13]. La figure IV-2 résume le prétraitement des données pour effectuer l'analyse du nouveau fichier résultant des différentes opérations citées précédemment.

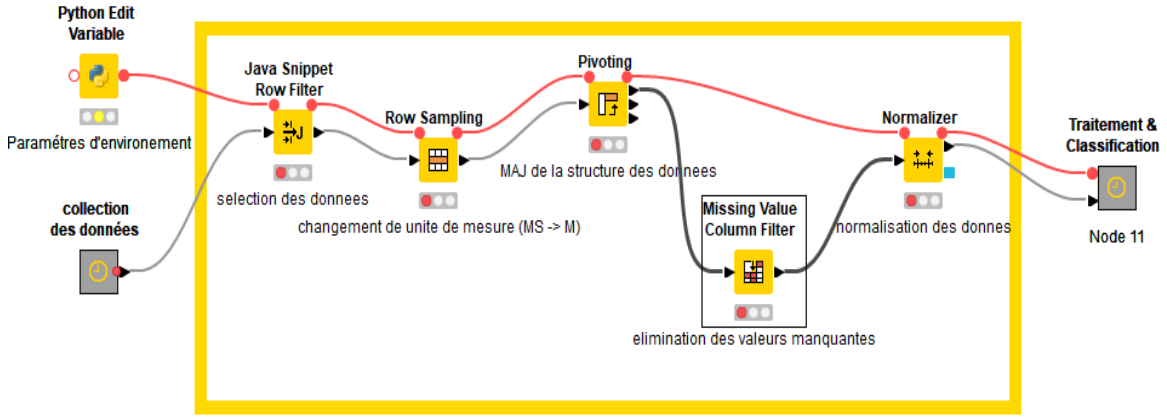

Figure IV-2 : Illustration de l'étape prétraitement sous la Platforme KNIME

IV.2.2 Collection des données

L'ensemble des données brutes utilisées dans cette étude de détection des activités à domicile est la base de données susmentionnée. Cet ensemble de données est de 12.4 Go (247.304.708 échantillons). Cette étape est effectuée en utilisant le nœud File Reader fournit par la plateforme KNIME.

IV.2.3 Sélection des données

L'opération de sélection de données est réalisée par le nœud "Filtre de ligne" qui est basé sur Java Snippet dans la plateforme KNIME. On peut également réaliser l'opération de sélection de données par le biais de l'éditeur Java en saisissant une condition booléenne liée aux identificateurs des capteurs. Cette opération consiste à tester si une ligne d'entrée est incluse et ensuite si la condition est vérifiée elle transmettra la donnée qui répond à cette condition à la sortie.

IV.2.4 Changement d'unité de mesure

Dans cette étape, on a changé la représentation de la colonne "TimeStamps" du millisecondes [yyyy-MM-dd hh:mm:ss:Ssss] aux minutes [yyyy-MM-dd hh:mm]. Cette étape est effectuée en utilisant des nœuds fournis par la plateforme KNIME (String to Date&Time) comment se présente dans la figure IV-4. Cette opération de changement d'unité de mesure nous permet de nous donner une idée très précise sur les activités au sein d'un habitat.

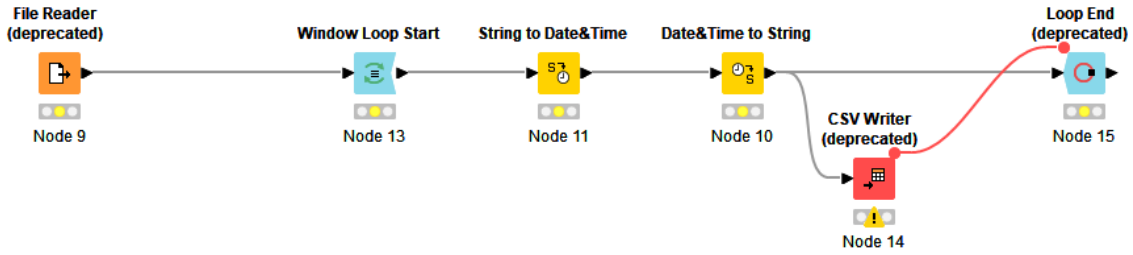

Figure IV-3 : Changement d'unité de mesure du ms en minute

IV.2.5 Restructuration des données et réduction des fréquences de prélèvement

Dans cette phase de prétraitement, on a effectué un pivotement sur la table d'entrée donnée en utilisant un nombre sélectionné de colonnes pour le regroupement et le pivotement. Les colonnes du groupe se traduiront par des lignes uniques, les valeurs de pivot étant transformées en colonnes, et pour chaque ensemble de combinaisons de colonnes la moyenne est calculée.

IV.2.6 Filtre pour les colonnes de valeurs manquantes "Missing value column filter"

Dans cette étape, on teste si les données manquantes vont perturber le modèle d'apprentissage ou non. Pour cela, on élimine les valeurs manquantes en déployant le nœud "Missing value". Ce nœud aide à gérer les valeurs manquantes trouvées dans les cellules de la table d'entrée. Le premier onglet de la boîte de dialogue (labeled "Default") propose des options de gestion par défaut pour toutes les colonnes d'un type donné.

IV.2.7 Normalisateur "Normalizer"

La normalisation des données est une étape obligatoire pour le bon déroulement de la phase traitement des données. Pour cela, on a opté pour une normalisation par mise à l'échelle décimale où les valeurs frontières d'une seule colonne sont être divisées par 10 jusqu'à l'obtention d'un résultat inférieur à 1. Ensuite, les autres valeurs de la colonne sont divisées par 10^j. Cette etape est effectuée en utilisant le nœud "Normlizer". La figure IV-4 illustre un exemple de la normalisation des données par le nœud "Normlizer" dans la plateforme KNIME et la figure IV-5 montre le résultat de l'opération de normalisation effectuée sur les données.

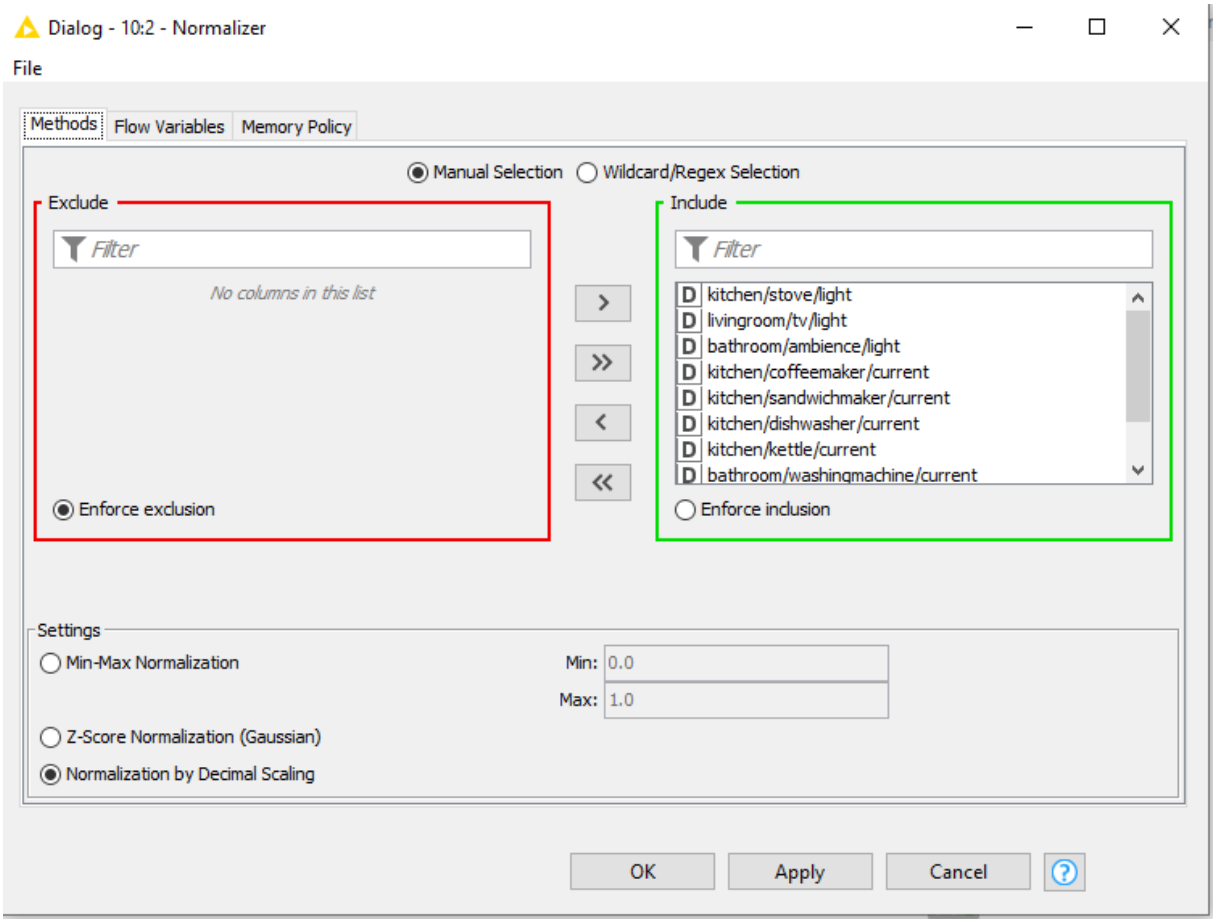

Figure IV-4 : Normalisation des données

File Table - 10:1 - CSV Reader

File Edit Hilite Navigation View

Table "default" - Rows: 187551 Spec - Columns: 11 Properties Flow Variables

Row ID	timestamp	kitchen/stove/light	livingroom/tv/light	bathroom/ambience/...	kitchen/coffee...	kitchen/sandwich...	kitchen/dishw...
Row39	23/03/2020 20:11	1,024	1,024	0	0	0.29	0
Row40	23/03/2020 20:12	1,024	1,024	0	0	0.296	0
Row41	23/03/2020 20:13	407.933	1,024	0	0	0.295	0
Row42	23/03/2020 20:14	289.133	1,024	0	0	0.312	0
Row43	23/03/2020 20:15	289.133	1,024	0	0	0.27	0
Row44	23/03/2020 20:16	289.383	1,024	0	0	0.291	0
Row45	23/03/2020 20:17	290.695	1,024	0	0	0.312	0
Row46	23/03/2020 20:18	293	1,024	0	0	0.295	0
Row47	23/03/2020 20:19	294.55	1,024	0	0	0.247	0
Row48	23/03/2020 20:20	295.217	1,024	0	0	0.287	0
Row49	23/03/2020 20:21	295.183	1,024	0	0	0.296	0
Row50	23/03/2020 20:22	297.717	1,024	0	0	0.304	0
Row51	23/03/2020 20:23	298.083	1,024	0	0	0.279	0
Row52	23/03/2020 20:24	298.867	1,024	0	0	0.307	0
Row53	23/03/2020 20:25	282.433	1,024	0	0	0.256	0
Row54	23/03/2020 20:26	275.083	1,024	0	0	0.294	0
Row55	23/03/2020 20:27	274.367	1,024	0	0	0.218	0
Row56	23/03/2020 20:28	275.517	1,024	0	0	0.306	0
Row57	23/03/2020 20:29	283.633	1,024	0	0	0.273	0
Row58	23/03/2020 20:30	300.867	1,024	0	0	0.296	0
Row59	23/03/2020 20:31	300.467	1,024	0	0	0.286	0
Row60	23/03/2020 20:32	302.639	1,024	0	0	0.272	0
Row61	23/03/2020 20:33	1,024	1,024	0	0	0.299	0
Row62	23/03/2020 20:34	1,024	1,024	0	0	0.299	0
Row63	23/03/2020 20:35	1,024	1,024	0	0	0.254	0

Figure IV-5 : Illustration de modification des données

IV.2.8 Traitement et Classification "Clustering"

Dans ce qui suit, nous allons appliquer des méthodes de classification pour connaitre les activités des occupants d'une maison en connaissant l'état des équipements au sein de cette maison. Ainsi, une fois les données sont filtrées et préparées correctement pour alimenter l'algorithme d'entraînement, il suffit de choisir l'algorithme d'apprentissage automatique à utiliser. Parmi ces méthodes de classification, nous avons proposé d'utiliser deux méthodes qui ont donné de bons résultats dans plusieurs domaines : K-Means et DBSCAN.

a) K-Means [16]

La méthode K-Means est basée sur un algorithme itératif qui minimise la somme des distances entre chaque individu et le centroïde. Le choix initial des centroïdes conditionne le résultat final. Étant donné un ensemble de points dispersés dans l'espace, K-Means déplace les points d'un cluster à un autre jusqu'à ce que la somme ne puisse plus être réduite. Après la fin de l'exécution de K-Means, nous obtenons un ensemble de clusters qui sont séparés. Cet ensemble de clusters est obtenu après avoir fixé le nombre de clusters par une des principales méthodes permettant de déterminer le nombre optimal de clusters [17] telles que Elbow [18], Rule of thumb [19], et Silhouette [20] ; et l'emplacement des centroïdes. La figure IV-6 illustre l'exécution de la méthode K-Means par la plateforme KNIME.

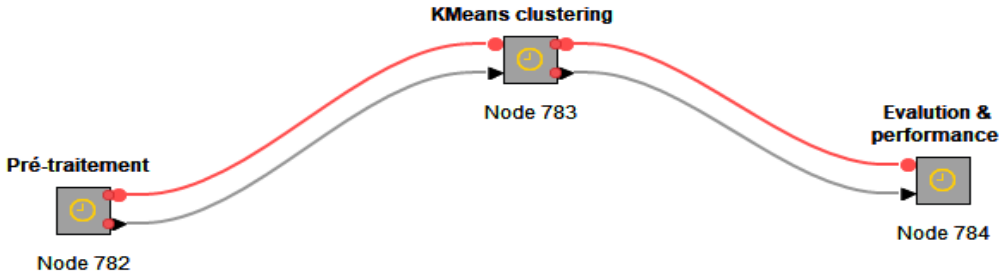

Figure IV-6: La méthode K-Means dans la plateforme KNIME

Dans cette méthode de classification, nous avons proposé de choisir la méthode Elbow [18] pour déterminer le nombre de clusters (k) comme montre la figure IV-7. Le choix de la méthode Elbow est justifié par son efficacité et sa précision car cette dernière détermine le nombre de clusters en fonction du déploiement des nœuds.

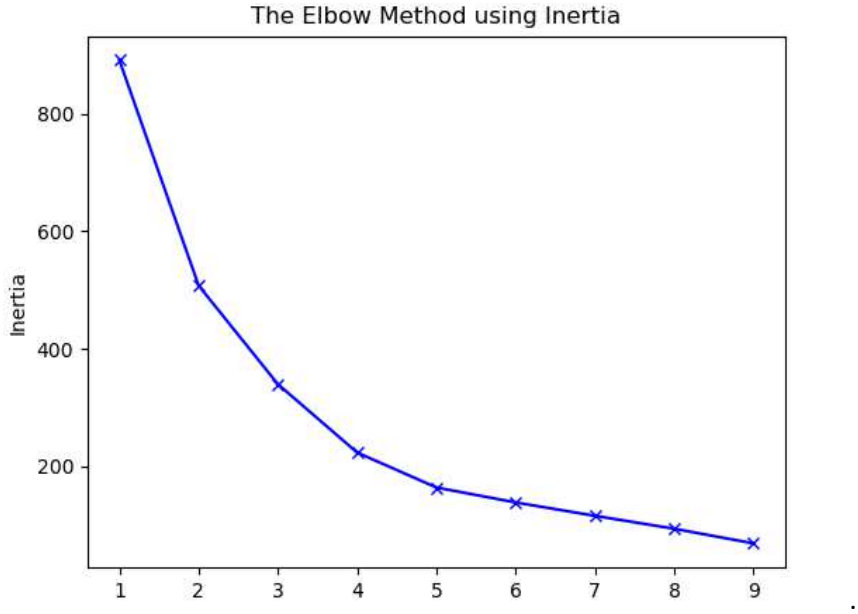

Figure IV-7 : La méthode ELBOW

Le programme suivant (écrit en Python) représente les étapes pour analyser des données en utilisant la méthode "K-Means".

```
output_table_1 = input_table_1.copy()
from sklearn.cluster import KMeans
import numpy as np
X = input_table_1.copy()
kmeanModel = KMeans(n_clusters=5).fit(X)
kmeanModel.fit(X)
output_table_1['labels'] = kmeanModel.labels_
```

b) DBSCAN [15]

La méthode de classification DBSCAN permet l'identification et le regroupement des objets d'une base de données en des classes. L'application de DBSCAN aux bases de données de grande taille (volumineuses) accroit les exigences des algorithmes de clustering tels que :

- Une certaine connaissance du domaine de son application afin de déterminer les paramètres d'entrée car les valeurs adéquates ne sont pas souvent connues au début lorsqu'il s'agit de bases de données de grandes tailles.

- La détermination des clusters de forme non défini (arbitraire) car les formes de clusters dans les bases de données spatiales peuvent prendre différentes formes à savoir linéaire, sphérique, etc...
- Très efficace lorsqu'il s'agit de bases de données volumineuses.

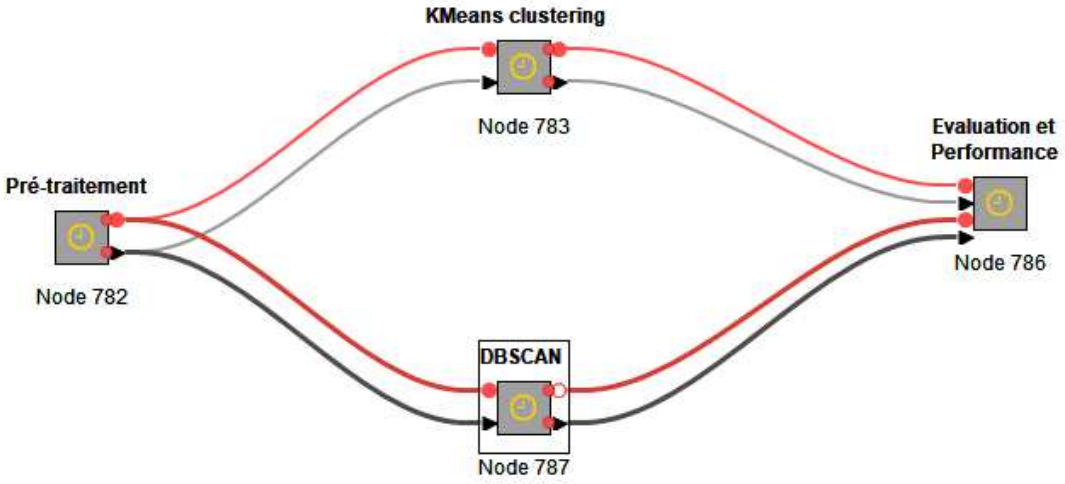

Figure IV-8 : La méthode DBSCAN

Le programme suivant (écrit en Python) représente les étapes pour analyser des données en utilisant la méthode "DBSCAN".

```
output_table_1 = input_table_1.copy()
from sklearn.cluster import DBSCAN
X = input_table_1.copy()
db = DBSCAN(eps=0.01, min_samples=1000)
db.fit(X)
```

Dans la configuration DBSCAN nous définissons deux paramètres Epsilon qui représente le "rayon" d'un voisinage et "points minimum" qui représente la densité d'un voisinage. Le nombre de clusters n'est pas défini au préalable et le calcul des distances se fait par le biais de la formule suivante :

$$\boldsymbol{D(p_1, p_2) = \sqrt{(\sum_{i=0}^{i=9} (x_{2i} - x_{1i})^2)}}$$

IV.2.9 Résultats obtenus

Pour évaluer les performances de chacune de deux méthodes (K-Means et DBSCAN), nous avons utilisé les paramètres mentionnés dans le tableau IV-1.

Tableau IV-1 : Différents paramètres dans les deux méthodes

Méthode	Paramètre	Valeur
K-Means	Nombre de clusters K	5
	Itération	300
DBSCAN	Epsilon	0.01
	Nbr min	1000

a) Les résultats obtenus par l'approche K-Means

Après la détermination du nombre de clusters par la méthode Elbow. Cette méthode nous a retourné cinq clusters (K=5). Ainsi, le nœud K-Means affecte les activités aux cinq clusters créés comme montre le tableau IV-2.

Tableau IV-2 : Les clusters obtenus par K-Means

Numéro du cluster	Condition	Activité
Cluster 0	Ambiance allumée + TV allumée + Stove éteinte	Faire la douche et regarder la TV
Cluster 1	Entre Ambiance +TV+ Stove au moins deux éteints	Faire la douche ou regarder la TV ou cuisinier ou rien
Cluster 2	Entre Ambiance +TV+ Stove au moins deux allumés	Faire la douche et regarder la TV et cuisinier
Cluster 3	Lave-vaisselle en marche	Laver la vaisselle
Cluster 4	TV allumée + ambiance éteinte	Regarder la TV

b) Les résultats obtenus par la méthode DBSCAN

Le premier tableau de sortie du nœud DBSCAN montre l'affectation des clusters de chaque ligne et le deuxième tableau de sortie montre la distribution des fréquences des clusters et des points de bruit comme s'est présenté dans le tableau IV-3.

Tableau IV-3 : Les clusters obtenus par DBSCAN

Numéro de cluster	**Condition**	**Activité**
Cluster 0	Cuisine + TV éteinte + Salle de bain éclairée	Cuisiner et Prendre une douche
Cluster 1	Four éteinte + TV éteinte + Ambiance allumée	Prendre une douche
Cluster 2	Tous éteints	Rien
Cluster 3	Cuisine+ Tv allumée + Ambiance éteinte	Cuisiner et Regarder la TV
Cluster 4	Tv allumée + Ambiance allumée	Regarder la TV et Prendre une douche
Cluster 5	Tv allumée + Four éteint	Regarder la TV

Tableau IV-4 : Comparaison entre les deux méthodes K-Means et DBSCAN

Méthode	Cluster0	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5
K-Means	Ambiance /TV	Ambiance / Stove /TV	Ambiance / Stove /TV	Diswasher	TV	
	83749	9772	30420	749	62861	
DBSCAN	Stove /Bathroom	TV /Ambiance		Stove/TV	TV /Ambiance	TV
	1459	2855	1801	21935	113251	37978

K-Means DBSCAN

Figure IV-9 : Comparaison entre K-Means et DBSCAN

IV.3 Conclusion

Les méthodes de clustering K-Means et DBSCAN ont montré des capacités remarquables dans l'apprentissage automatique plus précisément le partitionnement des données d'une manière non supervisée. En outre, pour augmenter les performances de la méthode K-Means, il est judicieux d'utiliser une méthode efficace qui permet de trouver le nombre de clusters optimal. Dans notre cas, on a fait appel à la méthode Elbow qui est jugée qu'elle est parmi les méthodes les plus performantes dans la littérature. En plus, la méthode K-Means a une complexité temporelle qui n'est pas conséquente; elle est seulement d'ordre $\theta(n)$, où n est la taille des données d'entrée.

Dans le cas de la méthode DBSCAN, il suffit d'utiliser de bonnes valeurs pour les deux paramètres "Radius" et "Nombre des nœuds voisins" pour bénéficier des atouts de cette méthode de clustering. En autre, ce qui est bien avec DBSCAN, le nombre de clusters est adaptatif selon les paramètres prédéfinis. Ainsi, une fonction est établie préalablement pour calculer la distance de n'importe quelle paire d'objets dans l'ensemble de données et quelques indications sur la distance considérée comme "proche". D'où la complexité de calcul de la méthode DBSCAN est d'ordre $\theta(n^2)$ où n est le nombre de degrés dans l'ensemble de données. DBSCAN produit également des résultats plus raisonnables que K-Means sur une variété de distributions différentes.

Conclusion générale

Conclusion générale

Dans le cadre de ce projet de fin d'études qui consiste à détecter les activités quotidiennes au sein des habitats pour connaitre la quantité d'électricité nécessaire pour répondre aux besoins des occupants de ces habitats et éviter une production d'électricité qui dépasse leurs besoins, nous avons proposé d'utiliser deux méthodes de classification pour atteindre cet objectif. Il s'agit de deux méthodes qui font parties des méthodes d'apprentissage automatique non supervisé : K-Means et DBSCAN. Ces deux méthodes ont prouvé leurs performances dans la littérature. Ces deux méthodes sont simples à les mettre en œuvre avec une certaine supériorité à la méthode DBSCAN.

L'utilisation de ces deux méthodes de classification, nous a permis de constater qu'il n'existe aucune approche qui n'est la meilleure pour tous les objectifs. Cela signifie que chacune des méthodes utilisées retourne de bonnes performances dans certaines situations alors que parfois ses performances sont très mauvaises dans d'autres situations. La méthode DBSCAN a montré plus d'efficacité par rapport à K-Means car DBSCAN est basé sur le clustering de densité qui semble correspondre davantage aux intuitions humaines du clustering, plutôt qu'à la distance d'un point central de clustering comme dans K-Means. En outre, les approches basées sur le clustering de densité telle que DBSCAN, utilisent le concept d'accessibilité, c'est-à-dire combien de voisins ont un point dans un rayon. En plus, DBSCAN n'a pas besoin de connaitre le nombre de clusters comme dans le cas de K-Means qui fait recours à une méthode permettant de trouver ce nombre pour illustrer son efficacité. Ainsi, lorsque nous ne connaissons pas le nombre de clusters caché dans un ensemble de données, il est judicieux d'utiliser DBSCAN car cette dernière produit un nombre variable de clusters, en fonction des données d'entrée.

Perspectives

Comme perspectives de notre projet de fin d'études, nous proposons tout d'abord d'exécuter ces méthodes de classification sur plusieurs maisons intelligentes pour mieux voir l'efficacité de la détection des activités par les méthodes proposées. Naturellement dans la vie réelle, un appareil au sein de l'habitat pourra cesser de fonctionner un moment donné (capteur, électroménager, résistance, etc.). C'est pourquoi nous proposons d'appliquer ces approches de

classification sur des bases de données contenant plusieurs types de défauts pour illustrer leur application dans un environnement réel.

Références bibliographiques

Références bibliographiques

[1] B. Reinteau, "Smart Grid,". Available: https://www.xpair.com/lexique/definition/smart-grid-reseau-intelligent.htm. [Accès le 15 April 2022].

[2] A. SOUISSI, "Réseaux électriques intelligents Smart Grid," Master Professionnel, Département de Génie électrique, ENSA de Khouribga, Université Hassan I, Maroc, 2016.

[3] A. Ferretti, "Smart grids" : les réseaux et les compteurs d'électricité intelligents : émergence d'une ère post-carbone, ou avènement de la société de contrôle,» Mémoire de fin d'études, Master 2 Recherche Design, médias technologie : Design & Environnement, Université Paris 1, France, 2014.

[4] "Smart Grids Le site édité par la La commission de régulation de l'énergie (CRE)," [En ligne]. Available: https://www.smartgrids-cre.fr/introduction-aux-smart-grids. [Accès le 20 April 2022].

[5] "Smart Grid Architecture Basics | Smart Grid Architecture Working," [En ligne]. Available: https://www.rfwireless-world.com/Articles/Smart-Grid-Architecture-basics-and-working.html. [Accès le 25 April 2022].

[6] C. De Perthuis, "Réseau intelligent (Smart Grid)," [En ligne]. Available: https://www.connaissancedesenergies.org/fiche-pedagogique/reseau-intelligent-smart-grid. [Accès le 15 May 2022].

[7] S. Bouvier et P. Strubel, Réseaux d'énergie intelligent (Smart grids): Déployer un réseau plus intelligent grâce à des solutions et services de câblage, Livre blanc, Nexans, 2010.

[8] F. X. Jeuland, La maison communicante: Réussir son installation domotique et multimédia, Paris, France: 4eme édition, Eyrolles, 2012.

[9] J. Hurwitz et D. Kirsch, Machine Learning For Dummies, IBM Limited Edition, 2018.

[10] C.-A. Azencott, Introduction au Machine Learning, Deuxième édition, Dunod, 2022.

[11] Z. ISMAILI, "Différence entre apprentissage supervisé et non supervisé," [En ligne]. Available: https://analyticsinsights.io/apprentissage-supervise-vs-non-supervise/. [Accès le 2022 April 15].

[12] "Data science technology for a better world," [En ligne]. Available: https://www.anaconda.com/. [Accès le 2022 April 15].

[13] "Open for Innovation KNIME," [En ligne]. Available: https://www.knime.com/. [Accès le 2022 March 30].

[14] G. Chimamiwa, M. Alirezaie, F. Pecora et A. Loutfi, "Multi-sensor dataset of human activities in a smart home environment," *Data in Brief (Elsevier),* vol. 34, p. 106632, 2021.

[15] M. Ester, H.-P. Kriegel, J. Sander et X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96),* p. 226–231, 1996.

[16] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability,* p. 281–297, 1967.

[17] M. B. Benmahdi et M. Lehsaini, "Performance evaluation of main approaches for determining optimal number of clusters in wireless sensor networks," *International Journal of Ad Hoc and Ubiquitous Computing,* vol. 33, n° %13, pp. 184-195, 2020.

[18] N. Bertagnolli, "Elbow method and finding the right number of clusters," December 2015. [En ligne]. Available: http://www.nbertagnolli.com/jekyll/update/2015/12/10/Elbow.html. [Accès le December 2018].

[19] T. Kodinariya et P. Makwana, "Review on determining number of cluster in K-means clustering," *International Journal of Advance Research in Computer Science and Management Studies,* vol. 1, n° %16, p. 90–95, 2013.

[20] L. Kaufman et P. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis, Hoboken, New Jersey, USA: John Wiley & Sons, Inc., 1990.

Résumé

Dans le cadre de ce projet de fin d'études, nous avons proposé d'utiliser deux méthodes de classification pour détecter les activités quotidiennes au sein des maisons intelligentes. Ceci dans le but est de connaitre la quantité d'électricité nécessaire pour répondre aux besoins des occupants de ces maisons intelligentes et éviter ainsi une production d'électricité qui dépasse leurs besoins c'est à dire créer une adéquation entre l'offre et la demande en terme de consommation d'électricité. Il s'agit de deux méthodes qui font parties des méthodes d'apprentissage automatique non supervisé : K-Means et DBSCAN. Les résultats obtenus sur un ensemble de données ont montré que DBSCAN fournit de bonnes performances comparée à K-Means sur une variété de distributions différentes.

Mots clés : Plateforme KNIME, Python, Réseaux de capteurs, K-Means, DBSCAN, Apprentissage non supervisé

Abstract

In this final year study projects, we have proposed to use two classification methods to detect the daily activities in smart homes. The goal is to know the amount of electricity needed to meet the needs of the occupants of these smart homes and thus avoid a production of electricity that exceeds their needs, i.e. to create a match between supply and demand in terms of electricity consumption. The two methods are part of the unsupervised machine learning methods: K-Means and DBSCAN. The results obtained on a dataset have shown that DBSCAN provides good performances compared to K-Means on a variety of different distributions.

Key words: KNIME platform, Python, Sensor networks, K-Means, DBSCAN, Unsupervised learning

ملخص

في إطار مشروع التخرج، اقترحنا استخدام طريقتين تصنيف للكشف عن الأنشطة اليومية في المنازل الذكية .وذلك لمعرفة كمية الكهرباء اللازمة لتلبية احتياجات ساكني هذه المنازل الذكية وبالتالي تجنب إنتاج كمية الكهرباء التي تتجاوز احتياجاتهم، أي خلق توازن بين العرض والطلب من حيث استهلاك الكهرباء .هاتان طريقتان تشكلان جزءًا من طرق التعلم الآلي غير الخاضعة للإشراف :K-Means و DBSCAN. أظهرت النتائج التي تم الحصول عليها على مجموعة البيانات أن أداء DBSCAN جيدًا مقارنةً بـ K-Means على مجموعة متنوعة من التوزيعات المختلفة.

الكلمات المفتاحية: منصة KNIME، Python ، شبكات الاستشعار، K-Means, DBSCAN، تعليم غير مشرف عليه