كلية العلوم و التقنيات فاس
ⵜⴰⵖⵡⵍⵍⴰⵢⵜ ⵏ ⵜⵓⵙⵙⵔⵜⵉⵏ ⴷ ⵜⵓⵣⵔⴼⵜⵉⵏ
Faculté des Sciences et Techniques de Fès

جامعة سيدي محمد بن عبد الله
ⵜⴰⵙⴷⴰⵡⵉⵜ ⵙⵉⴷⵉ ⵎⵓⵃⴰⵎⴻⴷ ⴱⵏ ⵄⴱⴷⵍⵍⴰⵀ
Université Sidi Mohamed Ben Abdellah

Département d'Informatique

Projet de fin d'études

Master Sciences et Techniques
Systèmes Intelligents & Réseaux

Proposition d'une approche pour estimer la quantité des matières chimiques et organiques dans le sol

Lieu du stage : MASCIR

Réalisé par :

- BENAICHA Moncef

Encadré par :

- Pr. MRABTI Fatiha
- Pr. BEN ABBOU Rachid
- Mr. RABIE Reda
- Mr. SAIDI Ouadi

Soutenu le 20.06.2019 devant le jury composé de :

- Pr. BEGDOURI Ahlame	Faculté des Sciences et Techniques de Fès	(Présidente)
- Pr. BENABBOU Abderrahim	Faculté des Sciences et Techniques de Fès	(Examinateur)
- Pr. MRABTI Fatiha	Faculté des Sciences et Techniques de Fès	(Encadrante)
- Pr. BEN ABBOU Rachid	Faculté des Sciences et Techniques de Fès	(Encadrant)

Année Universitaire 2018 – 2019

Remerciements

Tout d'abord, j'adresse mes remerciements à mes enseignants, et mes encadrants ***Pr MRABTI Fatiha****, et* ***Pr BEN ABBOU Rachid*** *de la Faculté des Sciences et Techniques de Fès Université Sidi Mohamed Ben Abdellah pour leurs aides, leurs encouragements, et leurs soutiens tout en long de mon cursus académique et mon stage.*

Je tiens aussi à remercier mes encadrants à la fondation MAScIR, ***Mr RABIE Reda*** *et* ***Mr SAIDI Ouadi*** *pour leurs conseils, leurs collaborations activent et aussi pour leurs présences.*

Mes remerciements à ***la direction de la fondation MAScIR*** *de nous avoir accueillies. Les membres du personnel qui ont mis tout en œuvre pour que notre stage se déroule dans les meilleures conditions possibles. Durant cette durée, j'ai eu l'occasion d'acquérir de nouvelles connaissances et compétences. Celles-ci me seront fort précieuses pour la réalisation des projets à venir.*

Je remercie ensuite l'ensemble des membres du jury, qui m'ont fait l'honneur de bien vouloir étudier avec attention mon travail. ***Pr BENABBOU Abderrahim*** *pour avoir accepté d'être rapporteur de ce mémoire.* ***Pr BEGDOURI Ahlame*** *pour m'avoir fait l'honneur d'accepter de présider ce jury.*

Mes vifs remerciements à toutes les personnes qui ont contribué de près ou de loin au succès de ce travail.

Résumé

Ce rapport intitulé “Proposition d’une approche pour estimer la quantité des matières chimiques et organiques dans le sol“ récapitule mon projet de fin d’études de mon cycle Master. L’analyse du sol se fait dans un laboratoire. En plus des ressources financières, cette analyse prend environ 72h pour avoir le bilan des quantités des matières chimiques et organiques. Ce projet consiste à faire une étude et conception d’un modèle capable à estimer la quantité des matières chimiques dans le sol et cela à base du spectre proche infrarouge et/ou les images du sol. Le but de la conception de ce modèle c’est d’avoir la possibilité de générer les bilans en temps réel. Nous avons utilisé en premier temps un modèle à base des machines à vecteurs support et un autre à base du perceptron multicouches pour estimer la quantité du N-total dans les échantillons du sol. Finalement, nous avons conçu un modèle hybride entre les algorithmes génétiques et MLP. L’implémentation des algorithmes génétiques et du perceptron multicouche nous a donné des résultats proches aux valeurs observés par rapport à l’implémentation du SVM ou MLP d’une façon indépendante.

Abstract

This report entitled "Suggesting an Approach to Estimate the Quantity of Chemical and Organic Materials in the Soil" summarizes my graduation project from my Master's degree cycle. Soil Analysis is done in a laboratory. In addition to financial resources, this analysis takes about 72 hours to have the quantities report of chemical and organic materials. This project consists of making a study and design a model able to estimate the quantity of the chemical materials in the ground and that based on the near-infrared spectrum and / or the images of the soil. The purpose of this model is to have the ability to generate the report in real time. We first used a model based on support vector machines and another based on multilayer perceptron to estimate the amount of N-total in soil samples. Finally, we designed a hybrid model between genetic algorithms and MLP. The implementation of genetic algorithms and multilayer perceptron gave us results close to the observed values compared to the SVM or MLP implementation in an independent way.

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

AG/GA	Algorithmes Génétiques / Genetics Algorithms
ANN	Artificial Neural Network
BD	Base de données
CPU	Central Processing Unit
DBSCAN	density-based spatial clustering of applications with noise
GPU	Graphics Processing Unit
IA/AI	Intelligence artificielle/Artificial intelligence
MAScIR	Moroccan foundation for advanced science innovation and research
MLP	Multi-Layer Perceptron
NIR	Near infra-red
PMC	Perceptron Multi Couches
RBF	Radial Basis Function
RMSE	Root Mean Square Error
SPIR	Spectroscopie proche infrarouge
SVM	Support Vectors Machine
VPS	Virtual Private Server

Introduction générale

Actuellement, les problèmes informatiques ne sont plus des problèmes dont la résolution se limite sur un algorithme simple, il s'agit des problèmes qui nécessitent de l'intelligence de la part de la machine ou proprement dite de ***l'intelligence artificielle***.

Le principe général de l'intelligence artificielle c'est de créer des machines capables à simuler l'intelligence humaine en utilisant des concepts théoriques et techniques. Ce terme qui est très répondu dans nos jours fut créé en 1956 dans la conférence de Dartmouth [1], et vu la limite des ordinateurs dans ce temps, ce domaine n'a pas connu beaucoup de succès.

Or, aujourd'hui, on voit bien les applications de l'intelligence artificielle dans plusieurs domaines tels que : la vision par ordinateur, traitement de la parole, les jeux vidéo, les systèmes experts, etc. La performance de ces algorithmes et leurs efficacités a poussé les entreprises à s'intéresser de plus en plus à ce domaine et ses applications, et c'est le cas pour la fondation marocaine des sciences avancée de l'innovation et de la recherche (MAScIR). En effet, le département microélectronique et packaging traite beaucoup des sujets dans domaine de l'intelligence artificielle, traitement de l'image, les systèmes décisionnels en temps réel, etc.

Dans ce contexte, la fondation MAScIR a pris l'initiative d'implémenter l'intelligence artificielle dans un domaine qui est un atout pour le Maroc, et c'est l'agriculture.

Le sol est la partie la plus superficielle de l'écorce terrestre. Il a des constituants organiques, minéraux, des gaz qui circule dans les interstices du sol, d'eau et des ions. Pour extraire la quantité des matières organiques et minérales, on effectue des analyses au niveau du laboratoire. Ces analyses prennent environ 72h, et ils demandent des ressources financières importantes. En se basant sur le spectre proche infrarouge et les images du sol, on veut créer un modèle capable à estimer la quantité des matières chimiques et organiques en temps réel. Dans un premier temps le modèle sera chargé à estimer la quantité du N-total puis il sera généralisé sur les autres minéraux.

Ce rapport est rédigé en quatre chapitres, une introduction générale et une conclusion :

Dans le premier chapitre nous présenterons le lieu du stage où la fondation MAScIR son statut et ses activités, suivis par une présentation de la problématique et la solution entamée.

Le deuxième chapitre sera réservé pour l'étude des données dont nous expliquerons la démarche suivie pour la collecte des données et les analyses et prétraitement effectué.

L'état d'art sera sujet du troisième chapitre. Nous détaillerons les différents algorithmes utilisés dans le contexte de notre projet d'une façon théorique.

Le quatrième chapitre sera réservé pour l'implémentation du modèle et les résultats.

Chapitre I: Contexte général du projet

Introduction

Au cours de ce chapitre, nous commencerons par une présentation sur l'organisme d'accueil, son fonctionnement, ses rôles et sa structure. Après nous allons introduire notre projet, son contexte général, ses objectifs, et la démarche suivie dans la conduite du projet. À la fin de ce chapitre, nous allons présenter le cahier des charges qu'on doit respecter.

I.1 Organisme d'accueil

I.1.1 Présentation de MASCIR

Moroccan foundation for **A**dvanced **Sc**ience **I**nnovation and **R**esearch (**MAScIR)**. Il s'agit d'un organisme de recherche à caractère scientifique et technique à but non lucratif. Il a vu le jour en 2007, pour mission principale la promotion de la recherche scientifique et le développement technologique.

MAScIR s'appuie sur la valorisation de la recherche pour mettre son expertise et son savoir-faire au service des industriels.

Outre ses plateformes scientifiques à la pointe de la technologie, MAScIR a investi dans un capital humain, des chercheurs et ingénieurs qui œuvrent dans des domaines aussi innovants que complémentaires. De l'environnement, aux énergies renouvelables, en passant par la santé. La recherche à MAScIR s'adapte aux besoins de la société et de l'industrie.

MAScIR contient, actuellement, 3 Plateformes :

- MAScIR Microélectroniques : créé vers la fin de l'année 2008, a pour objectif de devenir un centre de Recherche et Développement dans le domaine de la microélectronique.
- MAScIR Biotechnologie : deuxième centre inscrit dans MAScIR œuvrant dans le domaine de la biotechnologie : recherche et développement des médicaments ou des biocides.
- Nano Technologie : qui a pour mission de mener des recherches appliquées, innovantes, et à la fine pointe de la technologie pour créer de la propriété intellectuelle et des prototypes dans le domaine des nanomatériaux et des nanotechnologies.

I.1.2 Département Microélectronique et packaging

MAScIR Microélectroniques est un centre d'innovation et de développement des technologies dans le milieu microélectronique. Il se concentre sur le micro packaging,

l'ingénierie, les tests de simulation, le design, la qualification, le prototypage de produits microélectroniques et les systèmes embarqués.

MAScIR Microélectroniques offre ces services aux clients industriels et partenaires tels que OCP, Lear, Lesieur, etc. et cela dans plusieurs domaines (Systèmes embarqués, management du projet …) :

MAScIR Microélectroniques se compose de trois (3) divisions : Packaging & prototyping, Design et Testing et Embedded System (Figure 1).

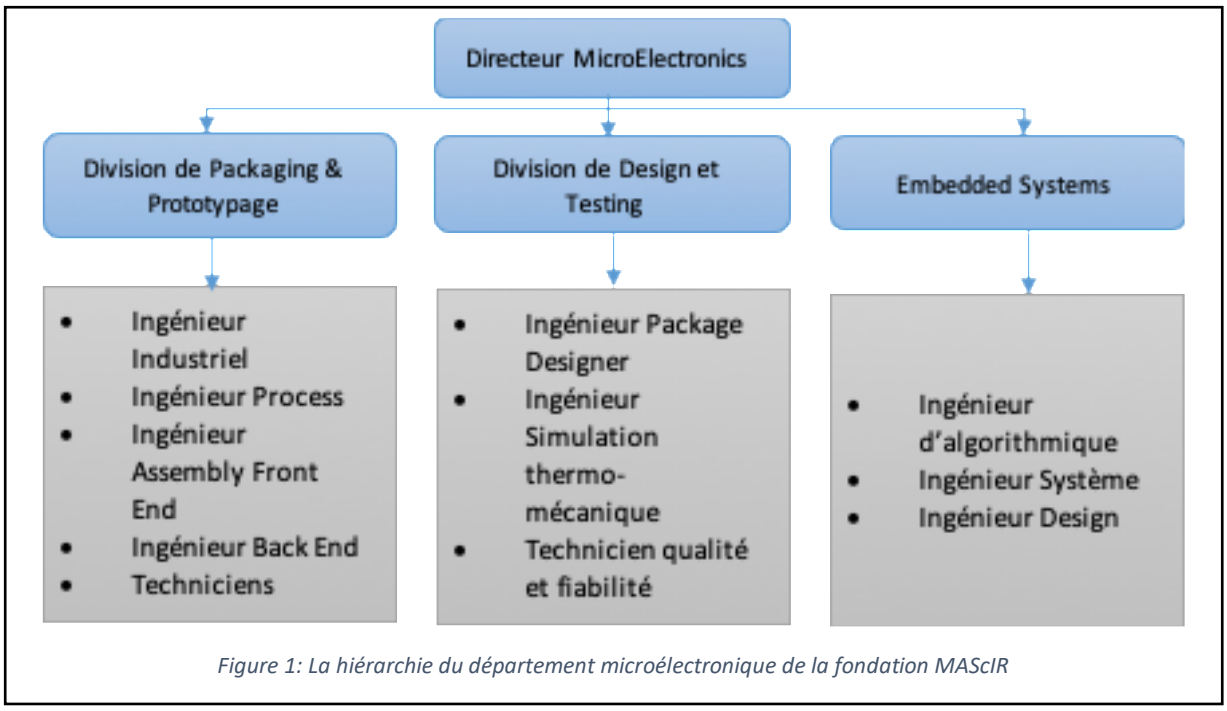


Figure 1: La hiérarchie du département microélectronique de la fondation MAScIR

I.2 Problématique

Notre sujet de stage s'inscrit dans le domaine de l'intelligence artificielle plus précisément dans le domaine d'apprentissage automatique.

Cette thématique, qui a connu une grande attraction pendant les dernières années représente un domaine d'intérêt pour les différentes équipes au sein de MAScIR.

Dans la même thématique, notre projet s'intéressera aux traitements du sol. L'équipe micro électroniques et packaging cherche à fabriquer un dispositif mobile connecté à un serveur distant. Ce dispositif-là sera capable à transmettre des informations sur le sol à analyser tels que :

- Une ou plusieurs images du sol.
- Le spectre proche infrarouge du sol.
- Les coordonnées GPS de l'échantillon à analyser.

Le sol est considéré comme une structure très complexe, même dans le laboratoire les techniciens utilisent plusieurs phases de séparation pour arriver à décortiquer cette structure et donner la valeur de chaque matière chimique. De plus, chaque matière chimique nécessite un traitement spécial pour arriver à donner une valeur exacte.

Cette complexité sera notre défi dans ce projet. En fait, prédire ces quantités chimiques à partir de l'image et du spectre proche infrarouge du sol sera une tâche délicate. Vu ces deux sources contiennent plusieurs informations ce qui influencera sur la qualité du modèle estimateur.

Le spectre proche infrarouge ou l'image donne des informations sur le sol avec tous ces composants confondus. Autrement dit, pour prédire par exemple la quantité du fer dans le sol en utilisant tout le spectre infrarouge ne donnera pas de résultat exact vu les autres matières chimiques ou le taux du H2O influencerons sur ce résultat.

Ce projet du stage s'intéressera à étudier le spectre proche infrarouge et les images du sol pour définir un algorithme capable à extraire les entrées adéquates du modèle d'apprentissage. Cet algorithme sera aussi capable de définir les entrées du modèle pour chaque matière chimique, cette séparation va permettre à notre modèle de maximiser son rendement.

Actuellement, l'analyse du sol se fait dans des laboratoires. Cette analyse donne un bilan sur la quantité de la matière organique et d'autres matières inclus le fer (Fe), Nitrate (NO3), Azote (N) etc.

Le bilan de cette analyse donne une idée au fermier sur la fertilité du sol, la quantité et la nature des engrais qu'il doit prévoir. Cependant, cette analyse prend beaucoup du temps (72 heures) et nécessite des ressources financières importantes en plus de déplacement des échantillons du sol et le risque des erreurs humaines.

Notre travail consiste à développer un modèle statistique de prédiction de la quantité de la matière organique et les autres minéraux qui existent dans le sol, et cela en étudiant son spectre proche infrarouge et son image capturée.

Vu la complexité du sujet et la courte durée de stage, on va essayer dans un premier temps de traiter le spectre proche IR du sol et voir le résultat et après inclure les images.

Au cours de ce projet, nous avons utilisés des échantillons des sols collectés des différentes régions du Maroc, cette diversification va nous permettre de bien tester et valider notre modèle. Ces données seront détaillées de plus dans le chapitre 2.

I.3 Solution

La solution qu'on a proposée, c'est d'utiliser les algorithmes d'apprentissage supervisé pour créer un modèle capable à estimer la quantité des matières organique et minérales dans le sol. Nous avons utilisé en premier lieu le spectre proche infrarouge et la valeur du N-total. Nous avons commencé par une analyse et un prétraitement des données pour définir les algorithmes que nous utiliserons. Nous avons conclu à utiliser les machines à vecteurs support et le perceptron multicouches, et un modèle hybride entre les algorithmes génétiques et le perceptron multicouches. L'implémentation des algorithmes génétiques était pour but de faire une sélection des longueurs d'ondes respectives pour l'estimation du N-total.

I.3.1 La charte du projet

La charte du projet s'agit d'un outil utilisé dans les milieux industriels qui permet de définir les objectifs et les contraintes principales du projet.

Notre travail a suivi la charte suivante (Tableau 1).

Tableau 1: La charte du projet

Raison du lancement du projet	Problème à résoudre dans le projet	Objectifs du projet	Durée du projet	Livrables	Critère de fin fonctionnelle
La nécessité d'un outil qui analyse le sol en temps réel.	La matrice du sol contient plusieurs composants, il faut trouver un moyen pour les séparer.	Définition d'un algorithme d'estimation de chaque valeur chimique dans le sol.	6 mois du 1er février, au 1er Aout.	+ Cahier de charge du projet. + Rapport de stage. + Modèle estimateur	Modèle capable à estimer la quantité d'une matière chimique dans le sol.

I.3.2 Planification du projet

Pour garantir le bon déroulement du projet nous avons dû appliquer une planification. Cette dernière va nous permettre de définir les travaux à réaliser, synchroniser les actions, diminuer des risques et tracer l'état d'avancement du projet.

Ce projet sera devisé en 3 phases :

1. Phase de collection et traitement des données.

   Dans cette phase on va collecter nos jeux de données, faire les prétraitements nécessaires et préparer les données à la phase d'apprentissage.

2. Conception du modèle de prédiction :

Cette phase sera consacrée à l'étude des différents algorithmes qui peuvent nous aider dans l'apprentissage du modèle. Dans cette partie nous essaierons de réaliser et comparer plusieurs modèles.

3. Réalisation :

   C'est la dernière partie de ce projet, dans laquelle on va implémenter notre modèle final et voir les résultats obtenus.

Conclusion

Au cours de ce chapitre, nous avons introduit l'entreprise d'accueil de ce stage, et nous avons défini la problématique du projet et sa solution. Ensuite, nous avons défini la charte et la planification du projet. Dans le prochain chapitre, nous introduirons les jeux des données avec leurs analyses et les différents prétraitements que nous avons effectués.

Chapitre II:Étude des données

Introduction

Dans ce chapitre nous expliquerons la démarche effectuer pour faire l'étude des données. Nous avons commencé par la collecte des données, ensuite la phase d'analyse et du prétraitement des données.

II.1 Collecte des données

Le Maroc se place dans la 4e place des terres agricoles les plus attractives en Afrique selon Land Matrix Report [2]. Ce classement n'est pas au hasard vu la richesse et la diversification du sol marocain. Cet avantage pour les fermiers s'agit d'un inconvénient pour les chercheurs. En outre, et pour une recherche plus juste et crédible, il fallait collecter les échantillons du sol de plusieurs régions du Maroc. Commençant par la région de l'oriental vers les régions de Tadla-Azilal en passant la région du Marrakech ce qui donne un total de 203 échantillons collectés (Figure 2).

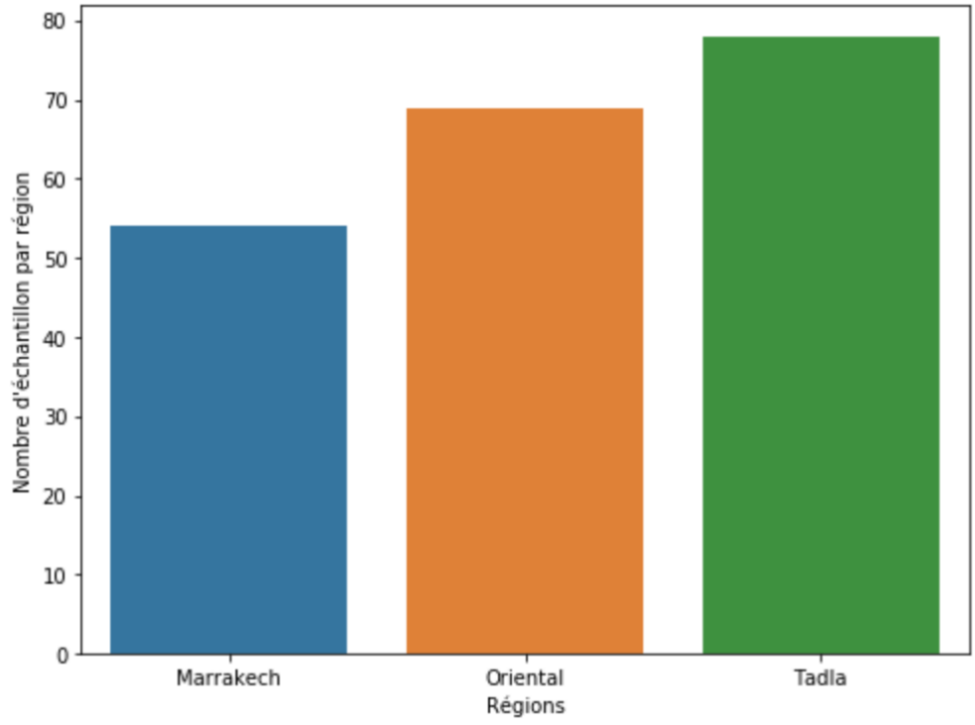


Figure 2: Répartition des échantillons du sol par région.

Par suite, ces échantillons sont distribués sur des laboratoires d'analyses à Rabat, Casablanca et Meknès inclut le laboratoire de MAScIR. Ces derniers effectuent les analyses nécessaires sur le sol et rendent un bilan des quantités des éléments chimiques et matières organiques. Le but de ces analyses c'est d'avoir une information a priori sur les quantités exactes et générer une base de données qui sera exploitée au cours de notre travail.

L'étape qui suit c'est la prise des images et le spectre proche infrarouge.

II.2 Etude des images du sol

Dans cette section, nous expliquerons les démarches effectuées pour collecter les images du sol. Ainsi, les différents prétraitements appliqués pour préparer la base de données. Cette BD sera exploitée dans la phase de conception du modèle d'apprentissage à base des images.

II.2.1 Capture des images

La capture des images s'est effectuée dans les laboratoires de MAScIR en utilisant un appareil photographique à reflex numérique. Les paramètres de l'appareil étaient fixés au cours des captures, de même la distance entre l'objectif de l'appareil et la boîte de pétri dans laquelle le sol a été étalé et aplati (Figure 3)

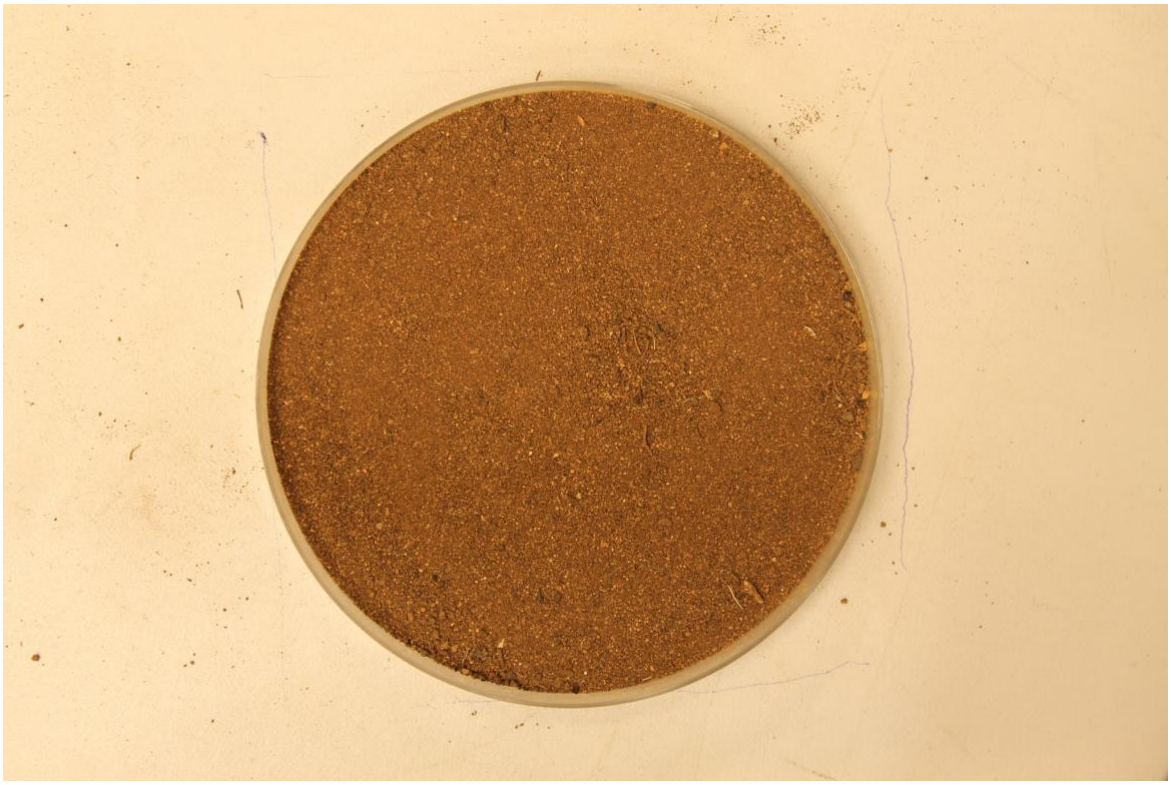


Figure 3: Capture d'un échantillon du sol

II.2.2 Analyse et prétraitement des images

Après la collection des images vient la phase de génération de la BD. Pour cela, nous avons dû à rogner l'imager à partir du centre. Nous avons fixé une résolution de capture des images sur notre appareil photographique à 2848×4272, cette dimension qu'y est assez grande ralentira les traitements ultérieurs et surchargera la mémoire vive de notre station de travail.

La solution était de choisir le point du centre de l'image et on calcule une distance de 25px dans chaque côté, ce qui donne un carré de taille 50px*50px. Le point du centre est pris sachant que la boîte de pétri est toujours centré dans la phase des captures. (Figure 4).

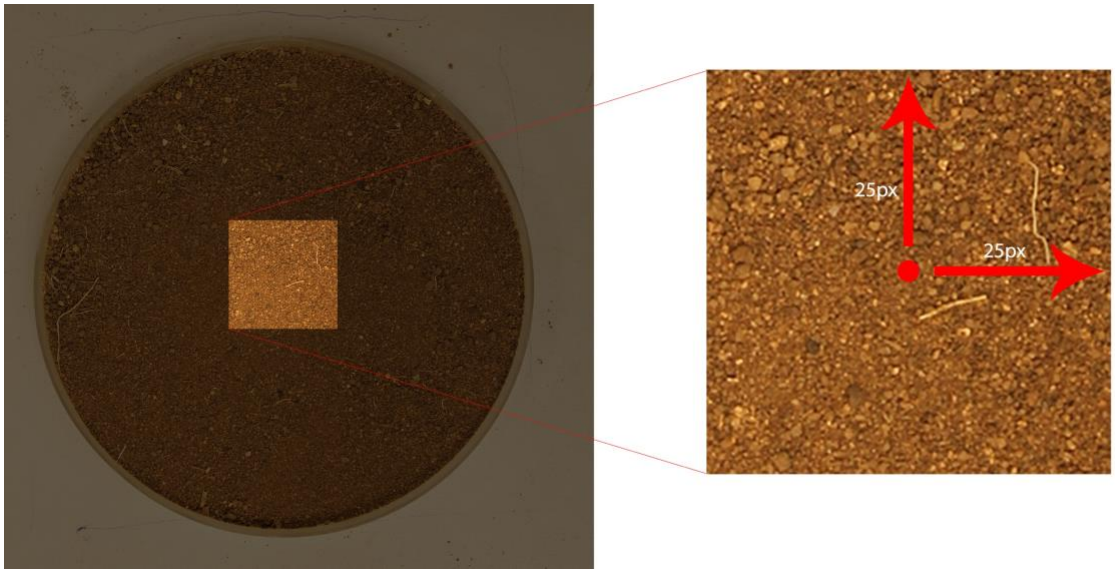


Figure 4 : Opération d'extraction du carré de l'image du sol.

Ce carré de taille 50px*50px est d'une image couleur, donc il s'agit d'une matrice à 3 dimensions. L'opération qui vient juste après l'extraction du carré, c'est la séparation du Rouge, vert, bleu. Donc à partir de la première matrice à 3 dimensions nous aurons 3 matrices à 2 dimensions.

En calculant la moyenne par colonne de chaque matrice nous aurons un vecteur de 50 éléments. Une concaténation avec le résultat de la même opération sur les deux matrices nous donne un vecteur de 150 éléments. En appliquant la même démarche sur toutes les images nous aurons une matrice finale de taille 201*150 ce qui constitue notre première base de données qui sera exploitée ultérieurement. La figure 5 représente une partie de notre base de données avec 2 échantillons du sol à titre d'exemple.

	0	1	2	3	4	5	6	7	8	9	...	140	141	142	143	144	145	146	147	148	149
or272	27.50	26.32	26.48	24.96	23.36	24.58	25.74	22.56	18.82	17.28	...	156.44	155.62	155.04	155.30	153.96	153.24	156.12	159.78	160.44	160.00
or281	29.00	27.96	24.28	25.92	27.36	25.20	24.32	26.56	28.34	31.96	...	157.92	158.34	155.84	152.74	151.06	151.98	152.10	152.48	149.28	145.52
m246	33.46	30.76	28.50	27.12	26.94	28.86	29.86	30.36	32.70	34.16	...	142.02	145.26	151.48	155.58	156.96	155.44	151.80	148.32	148.36	144.98
or275	29.30	29.86	30.66	31.60	31.12	31.38	31.72	32.86	33.20	33.44	...	155.60	155.56	154.72	155.08	154.94	154.40	154.42	155.38	156.20	156.32
td26	26.62	26.54	28.98	31.12	32.84	33.70	35.90	38.04	37.40	32.74	...	157.08	158.60	159.04	157.40	156.12	158.30	160.96	162.24	161.16	160.84

5 rows × 150 columns

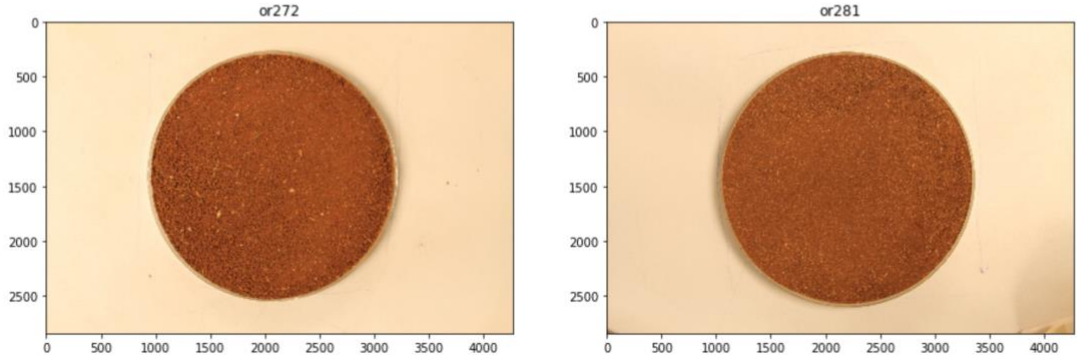


Figure 5: Partie de la BD finale des images avec 2 échantillons du sol.

II.3 Analyse et prétraitement du spectre proche infrarouge du sol

II.3.1 Spectroscopie

La spectroscopie infrarouge repose sur le principe que chaque groupement chimique absorbe la lumière différemment en fonction de sa longueur d'onde. Les bandes d'absorption (zone où la lumière est absorbée) permettent donc d'identifier les groupements atomiques. Beer (1729) et Lambert (1760) ont ainsi proposé d'observer l'atténuation d'un faisceau de la lumière afin de prédire la concentration d'un composé selon l'expression suivante

$$A_\lambda = \varepsilon_\lambda * l * c$$

Avec :

- A_λ **:** C'est l'Absorbance à une longueur d'onde λ donnée.
- ε_λ **:** C'est le coefficient d'extinction molaire.
- l **:** la longueur du trajet optique dans l'échantillon.
- c **:** la concentration de la solution.

La première application analytique de NIR était en 1962 par Hart et Norris [3]. Ils ont fondé un modèle linéaire sur la loi de longueur d'onde unique.

Cependant, étant donné que les pics d'absorption dans le NIR sont larges (l'absorption de la lumière se produit sur les harmoniques des fréquences vibratoires des molécules), se superposent et que de nombreuses interactions entre molécules modifient le signal, ce qui rend difficile à appliquer la loi de Beer -Lambert.

L'identification d'une molécule nécessite l'analyse de la signature spectrale à plusieurs longueurs d'onde. Les bandes d'absorption des composés peuvent également changer en fonction de l'environnement dans lequel ils sont situés (température, charges ioniques des molécules adjacentes).

Par contre, à l'absorption s'ajoute très souvent un autre phénomène : la diffusion. La structure physique de l'échantillon influe de manière significative sur le trajet des photons qu'il contient, que ce soit des fibres, des cellules, une poudre, une suspension ... Cette modification du trajet optique est caractérisée par une séquence de modifications de la direction des photons par des phénomènes de réflexion / réfraction. Ce phénomène, appelé "diffusion multiple", est souvent plus courant que le phénomène d'absorption dans l'environnement. Par conséquent, pour des produits tels que les poudres, on estime qu'il existe un phénomène d'absorption pour 100 phénomènes de diffusion. Ces phénomènes ont deux conséquences majeures : premièrement, la loi de Beer-Lambert n'apparaît plus valable, deuxièmement, le spectre obtenu contient à la fois des informations de nature chimique et de nature physique, car l'atténuation du faisceau n'est plus uniquement due à l'absorption.

L'analyse du spectre proche infrarouge propose plusieurs avantages tels que :

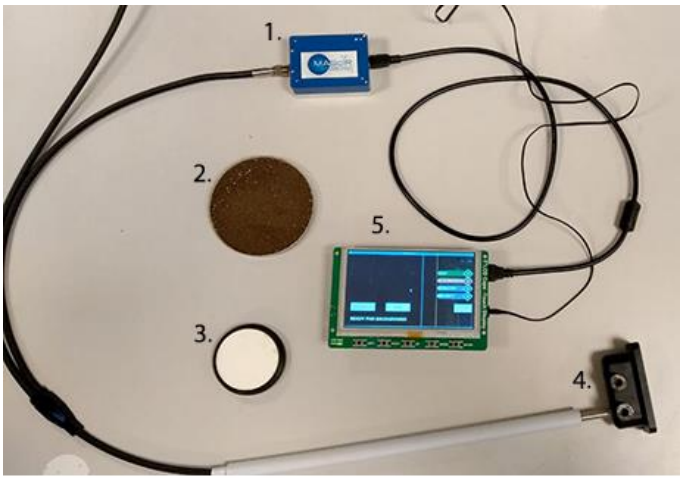
- Analyse rapide multi composant, en temps réel.
- Coût d'analyse moins élevé.
- L'échantillon nécessite peu ou pas de préparation
- Possibilité d'analyse de produits toxiques ou dangereux à distance (+ de 500m en utilisant des fibres optiques)

De même cette méthode d'analyse propose des inconvénients, dont on peut citer :

- Besoin de calibration de l'appareil (Analyse directe très difficile).
- Problème de transfert de calibration d'une méthode d'un appareil à l'autre.

II.3.2 Analyse et prétraitement du spectre

L'acquisition du spectre se fait à l'aide d'un spectromètre. Cet appareil émet un faisceau lumineux et calcule le taux d'absorption par longueur d'onde. Chaque spectromètre à ses propres propriétés. Au cours de la collection des données nous avons utilisé un spectromètre propre à MAScIR. Ce dernier balaye sur les différentes longueurs d'onde de 1300nm vers 2600nm dans le domaine infrarouge.



1. Spectromètre
2. Echantillon du sol
3. Calibrateur
4. Fibre optique
5. Ecran de visualisation

Figure 6: Espace du travail utilisé dans l'acquisition du SPIR.

Pour chaque échantillon, et pour éviter les erreurs de l'appareil, on effectue 5 mesures (Figure 7), et on calcule leur moyenne qui donne le spectre à prendre en considération.

Après l'opération d'acquisition vient la phase de construction de la base de données à base des spectres. Pour chaque échantillon, nous avons 5 mesures. On calcule la moyenne de ces 5 mesures par bande, ce qui nous donne un seul spectre par échantillon. De la même manière sur tous les échantillons pour obtenir à la fin la base de données initiale.

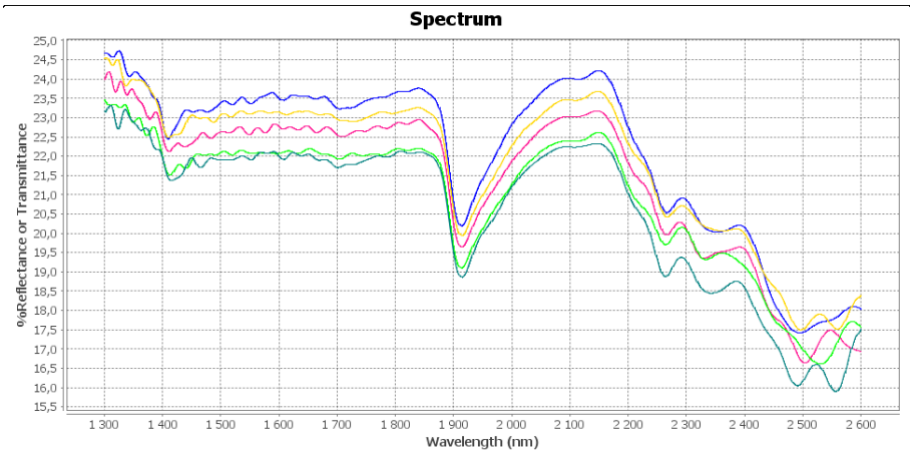


Figure 7: Résultat de 5 mesures spectroscopiques d'un échantillon.

Tant que nous allions effectuer un apprentissage supervisé, nous avions besoin des données de sortie. C'est les données eues aux laboratoires. En premier temps nous procédera à faire un apprentissage au modèle pour estimer la valeur de **N-Total** et après on généralisera le travail sur toutes les matières chimiques**.** La figure 8 représente une partie de la base de données des spectres.

	N total (mg/kg sol)	2498.644704	2495.082403	2491.530244	2487.988185	2484.456183	2480.934195	2477.422179	2473.920092	2470.427891	...	1308.55858
S1	1719.0	34.760547	34.944935	35.172282	35.434620	35.721147	36.019630	36.318014	36.605971	36.876121	...	53.410526
S2	2981.0	46.161930	46.350498	46.570853	46.819466	47.091146	47.379645	47.678438	47.981541	48.284223	...	58.735866
S3	1411.0	45.855803	45.942104	46.038009	46.139697	46.243352	46.345987	46.446154	46.544406	46.643434	...	52.486394
S4	1689.0	39.854379	40.071258	40.325022	40.601396	40.884960	41.161020	41.417423	41.646001	41.843415	...	55.310971
S5	1620.0	43.561252	43.721158	43.924180	44.164269	44.433097	44.720977	45.018021	45.315347	45.606113	...	63.180695

5 rows × 648 columns

Figure 8: Une Partie de la BD du spectre infrarouge du sol.

Pour déterminer les modèles d'apprentissage adéquats à appliquer sur cette base de données nous avons été amenés à appliquer des tests :

1. Test statistique de normalité

Ce test permet de déterminer si les données suivent une distribution normale ou non.

Nous avons appliqué deux tests de normalité le premier est le test de Shapiro-Wilk [4]. Il s'agit d'un test statistique très populaire basé sur la statistique W.

$$w = \frac{\left[\sum_{i=1}^{\left[\frac{n}{2}\right]} a_i\left(x_{(n-i+1)} - x_i\right)\right]^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

Où :

- x_i : Correspond à la série des données triées.
- $\left[\frac{n}{2}\right]$: C'est la partie entière du rapport $\frac{n}{2}$
- a_i : sont des constantes générées à partir de la moyenne et de la matrice de variance covariance des quantiles d'un échantillon de taille n suivant la loi normale. Ces constantes sont fournies dans des tables spécifiques.

La statistique W peut donc être interprétée comme le coefficient de détermination (le carré du coefficient de corrélation) entre la série des quantiles générés à partir de la loi normale et les quantiles empiriques obtenus à partir des données. Plus W est élevé, plus la compatibilité avec la loi normale est crédible. La région critique, rejet de la normalité, s'écrit :

$$W < Wcrit.$$

Les valeurs seuils Wcrit pour différents risques α et effectifs n sont lues dans la table de Shapiro-Wilk.

L'application du test nous a donné un *p*-value = **1.0E-21** < 0.05. Donc on rejet l'hypothèse H_0 et on accepte H_1. Ce qui signifie que les données ne suivent pas une loi normale.

2. Test graphique de normalité

Test par droite d'Henry [4]. C'est une technique graphique qui permet de comparer les distributions de deux ensembles de données.

Si les données sont compatibles avec la loi normale, les points (x(i), x*(i)) forment une droite, dite droite de Henry, alignés sur la diagonale principale.

La figure suivante représente le résultat de ce test sur notre base de données.

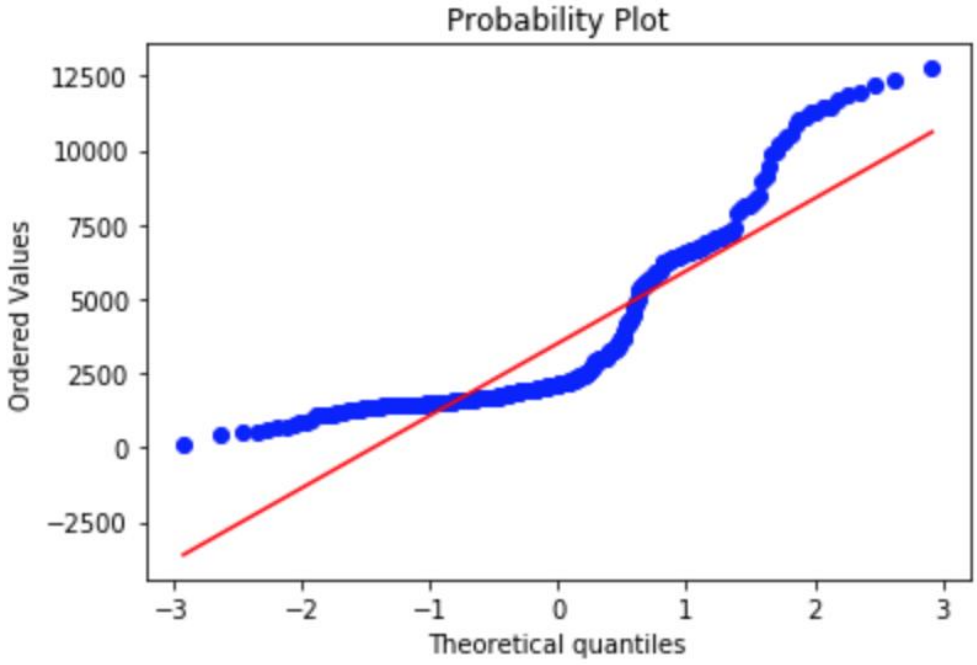


Figure 9: Test de normalité par la droite de Henry.

Tant que la droite d'Henry n'est pas aligné sur la diagonale principale, Les données ne suivent pas une distribution normale.

L'information qu'on peut conclure à partir de ces deux tests, c'est qu'on ne peut pas utiliser des modèles d'apprentissage qui exige que les données suivent une distribution normale tels que : régression linéaire – analyse discriminante linéaire, etc.

3. Détection des valeurs aberrantes

Non seulement la normalité des données qui influence le modèle d'apprentissage, mais aussi l'homogénéité des données.

Dans la phase de collecte de données, on peut toujours commettre des erreurs. Le prélèvement du spectre se fait d'une façon manuelle [cf. Figure 6], et un simple mouvement de la main peut conduire à une variation importante du taux d'absorption malgré les mesures de sécurité qu'on peut prendre lors des acquisitions, et multitude des mesures pour atténuer la marge d'erreur.

De même, la valeur de sortie qu'on doit prédire et que notre modèle va utiliser pour faire l'apprentissage est calculée dans un laboratoire, donc on peut toujours avoir des valeurs erronées ou aberrantes.

Notre objectif dans la phase qui vient après c'est la détection des valeurs aberrantes dans notre base de données.

Il existe plusieurs méthodes pour détecter les valeurs aberrantes. Certaines méthodes statistiques comme le z-score, d'autres à base d'apprentissage non supervisé tel que DBSCAN, ou plus simplement les méthodes graphiques à titre d'exemple : les boîtes à moustaches – histogramme, etc.

Pour déterminer les valeurs aberrantes dans notre base de données, nous avons utilisés les boites à moustaches. Comme expliqué dans [5] les boites à moustaches sont utiles dans la phase d'analyse exploratoire. Ils permettent de visualiser les données et voir la médiane par rapport et le premier quartile et le troisième quartile.

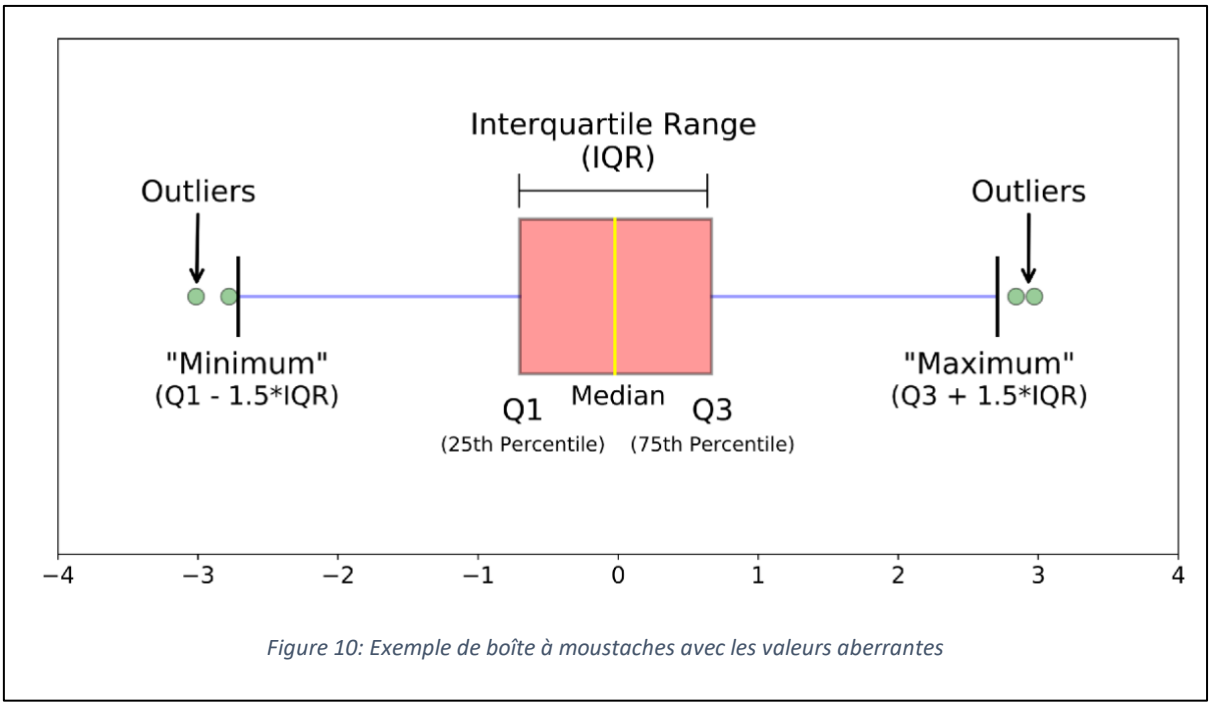


Figure 10: Exemple de boîte à moustaches avec les valeurs aberrantes

L'application de la méthode boîte à moustaches nous montre qu'il y a des valeurs aberrantes dans nos données. (Figure 11)

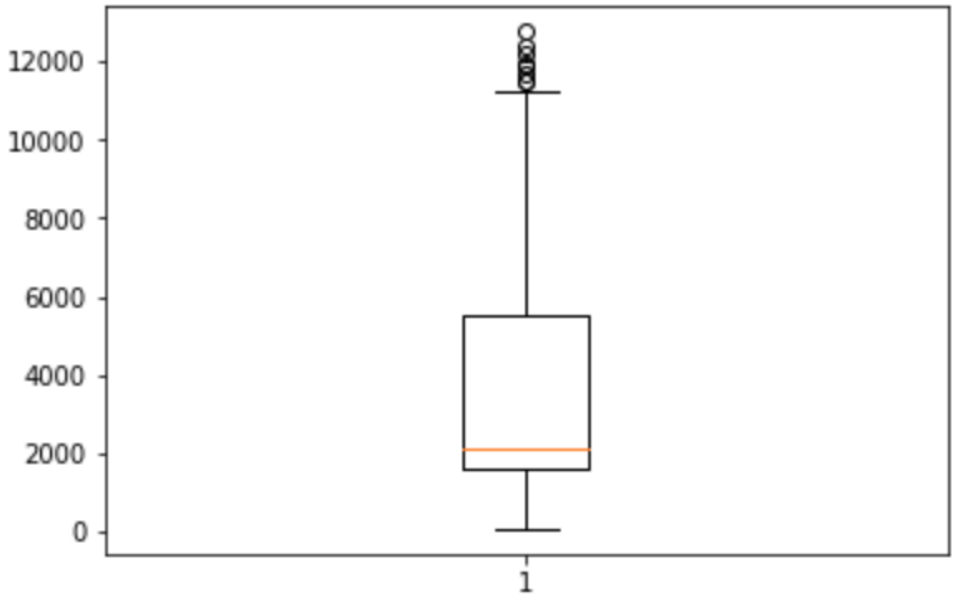


Figure 11: La représentation de nos données par les boîtes à moustaches (Box Plot)

Après la détection des valeurs aberrantes, il faut absolument réagir avant de continuer vers la phase de modélisation. Il y a deux moyens, la correction ou la suppression de ces échantillons. Pour corriger ces valeurs, il faut renvoyer l'échantillon vers le laboratoire pour refaire les calculs, ce qui prend assez de temps. La solution la plus simple dans notre cas c'est la suppression. Mais, en revenant vers la représentation on constate que les valeurs du N-total dans les environs de 12000 sont considérées comme des valeurs aberrantes, à part l'hypothèse d'une faute de calcul dans le laboratoire, cela peut être dû au nombre très peu des échantillons qui donnent cette valeur. Donc la meilleure solution qu'on peut implémenter ici, c'est de garder deux versions de la base de données et effectuera les tests avec les deux.

La suppression des valeurs aberrantes nous poussera à éliminer **79** échantillons, vu que nous n'avons pas assez d'échantillons, 391 au total, cette suppression aura un impact sur la phase d'apprentissage de notre modèle. Peu d'échantillons peuvent conduire à un sur apprentissage. Nous essaierons dans la phase de modélisation de prendre en considération ce dilemme. La figure 12 représente nos données après la suppression des valeurs aberrantes.

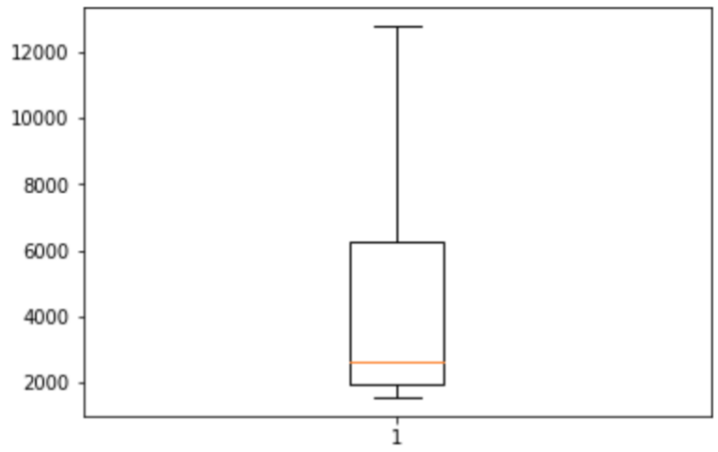


Figure 12: La représentation de nos données par les boîtes à moustaches après la suppression des valeurs aberrantes

Conclusion.

Au cours de ce chapitre, nous avons décrit d'une façon détaillée les données que nous avons, les méthodes utilisées dans la collecte des données, et aussi les prétraitements que nous avons effectués.

Chapitre III: État de l'art

Introduction

La phase de modélisation, qui s'appuie sur la phase de collecte et prétraitement de données, est très cruciale. Dans ce chapitre, nous avons décri d'une façon théorique les différents algorithmes que nous avons utilisés dans la conception de notre estimateur.

III.1 Les machines à vecteur support

III.1.1 Définition

Supports Vectors Machines souvent traduits par l'appellation de Séparateur à Vaste Marge (SVM) sont une classe d'algorithmes d'apprentissage initialement définis pour la discrimination c'est-à-dire la prévision d'une variable qualitative initialement binaire. Ils ont été ensuite généralisés à la prévision d'une variable quantitative. Dans le cas de la discrimination d'une variable dichotomique, ils sont basés sur la recherche de l'hyperplan de marge optimale qui, lorsque c'est possible, classe ou sépare correctement les données tout en étant le plus éloigné possible de toutes les observations. Le principe est donc de trouver un classifieur, ou une fonction de discrimination, dont la capacité de généralisation (qualité de prévision) est la plus grande possible [6].

Cette approche découle directement des travaux de Vapnik en théorie de l'apprentissage à partir de 1995. Elle s'est focalisée sur les propriétés de généralisation (ou prévision) d'un modèle en contrôlant sa complexité. Voir à ce sujet la vignette sur l'estimation d'un risque et la section introduisant la dimension de Vapnik-Chernovenkis comme indicateur du pouvoir séparateur d'une famille de fonctions associé à un modèle et qui en contrôle la complexité. Le principe fondateur des SVM est justement d'intégrer à l'estimation le contrôle de la complexité c'est-à-dire le nombre de paramètres qui est associé dans ce cas au nombre de vecteurs supports. L'autre idée directrice de Vapnik dans ce développement, est d'éviter de substituer à l'objectif initial : la discrimination, un ou des problèmes qui s'avèrent finalement plus complexes à résoudre par exemple l'estimation non paramétrique de la densité d'une loi multidimensionnelle en analyse discriminante [6].

Cet outil devient largement utilisé dans de nombreux types d'application et s'avère un concurrent sérieux des algorithmes les plus performants (agrégation de modèles). L'introduction de noyaux, spécifiquement adaptés à une problématique donnée, lui confère une grande flexibilité pour s'adapter à des situations très diverses (reconnaissance de formes, de séquences génomiques, de caractères, détection de spams, diagnostics...) [6].

À noter que, sur le plan algorithmique, ces algorithmes sont plus pénalisés par le nombre d'observations, c'est-à-dire le nombre de vecteurs supports potentiels, que par le nombre de variables. Néanmoins, des versions performantes des algorithmes permettent de prendre en compte des bases de données volumineuses dans des temps de calcul acceptables [6].

III.1.2 Principe général

Le principe de base des SVM consiste de ramener le problème de la discrimination à celui, linéaire, de la recherche d'un hyperplan optimal. Deux idées ou astuces permettent d'atteindre cet objectif [6] :

- La première consiste à définir l'hyperplan comme solution d'un problème d'optimisation sous contraintes dont la fonction objective ne s'exprime qu'à l'aide de produits scalaires entre vecteurs et dans lequel le nombre de contraintes "active" ou vecteurs supports contrôle la complexité du modèle.
- Le passage à la recherche de surfaces séparatrices non linéaires est obtenu par l'introduction d'une fonction noyau (kernel) dans le produit scalaire induisant implicitement une transformation non linéaire des données vers un espace intermédiaire (feature space) de plus grande dimension. D'où l'appellation couramment rencontrée de machine à noyau ou kernel machine. Sur le plan théorique, la fonction noyau définit un espace hilbertien, dit auto-reproduisant et isométrique par la transformation non linéaire de l'espace initial et dans lequel est résolu le problème linéaire.

La résolution d'un problème de séparation linéaire est illustrée par la figure 13. Dans le cas où la séparation est possible, parmi tous les hyperplans solutions pour la séparation des observations, on choisit celui qui se trouve le plus "loin" possible de tous les exemples, on dit encore, de *marge maximale*.

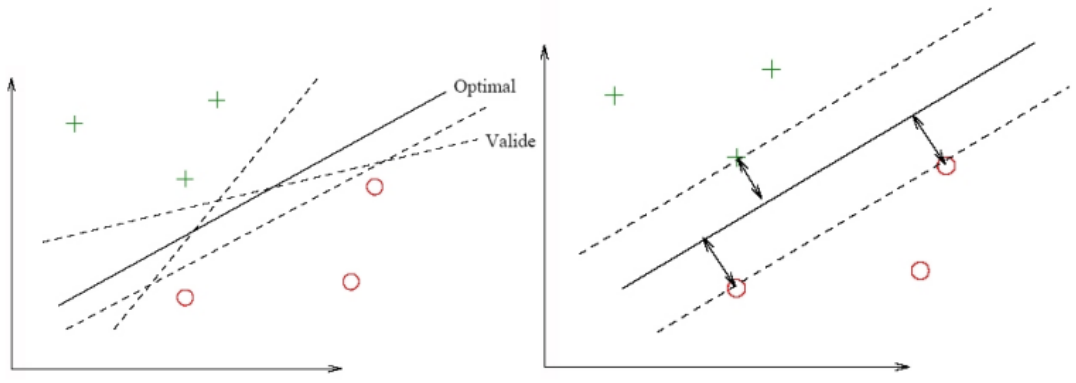


Figure 13: Recherche d'un hyperplan de séparation optimal au sens de la marge maximale.

Dans le cas linéaire, un hyperplan est défini à l'aide du produit scalaire de H par son équation :

$$\langle w, x \rangle + b = 0$$

où w est un vecteur orthogonal au plan tandis que le signe de la fonction $f(x) = \langle w, x \rangle + b$ indique de quel côté se trouve le point x à prédire. Plus précisément, un point est bien classé si et seulement si :

$$yf(x) > 0$$

Mais, comme le couple (w, b) qui caractérise le plan est défini à un coefficient multiplicatif près, on s'impose :

$$yf(x) \geq 1.$$

Un plan (w, b) est un séparateur si :

$$y_i f(x_i) \geq 1 \quad \forall i \in \{1, \dots, n\}$$

La distance d'un point x au plan (w, b) est donnée par :

$$d(x) = \frac{|\langle w, x \rangle + b|}{\|w\|} = \frac{|f(x)|}{\|w\|}$$

Et, dans ces conditions, la marge du plan a pour valeur $\frac{2}{\|w\|^2}$. Chercher le plan séparateur de marge maximale revient à résoudre le problème ci-dessous d'optimisation sous contraintes (problème primal).

$$\begin{cases} min_w \frac{1}{2}\|w\|^2 \\ avec\ \forall i, y_i(< w, x_i > +b) \geq 1 \end{cases}$$

Le problème dual est obtenu en introduisant des multiplicateurs de Lagrange. La solution est fournie par un point-selle (w*, b*, λ*) du lagrangien :

$$L(w, b, \lambda) = 1/2\|w\|_2^2 - \sum_{i=1}^{n} \lambda_i [y_i(< w, x_i > +b) - 1]$$

Ce point-selle vérifie en particulier les conditions :

$$\lambda_i^* [y_i(< w^*, x_i > +b^*) - 1] = 0 \quad \forall i \in \{1, \dots, n\}.$$

Les vecteurs support sont les vecteurs x_i pour lesquels la contrainte est active, c'est-à-dire les plus proches du plan, et vérifiant donc :

$$y_i(< w^*, x_i > +b^*) = 1$$

Les conditions d'annulation des dérivées partielles du lagrangien permettent d'écrire les relations que vérifie le plan optimal, avec les λ_i^* non nuls seulement pour les points supports:

$$w^* = \sum_{i=1}^{n} \lambda_i^* y_i x_i \quad \text{et} \quad \sum_{i=1}^{n} \lambda_i^* y_i = 0$$

Ces contraintes d'égalité permettent d'exprimer la formule duale du lagrangien :

$$w(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{n} \lambda_i \lambda_j y_i y_j < x_i, x_j >$$

Pour trouver le point-selle, il suffit alors de maximiser W(λ) avec $\lambda_i \geq 0$ pour $\forall i \in \{1, \dots, n\}$. La résolution de ce problème d'optimisation quadratique de taille n fournit l'équation de l'hyperplan optimal :

$$\sum_{i=1}^{n} \lambda_i^* y_i < x, x_i > + b^* \quad \text{avec} \quad b^0 = -\frac{1}{2}[< w^*, sv_{class+1} > + < w^*, sv_{class-1} >]$$

Pour une nouvelle observation x non apprise présentée au modèle, il suffit de regarder le signe de l'expression :

$$f(x) = \sum_{i=1}^{n} \lambda_i^* y_i < x, x_i > + b^*$$

pour savoir dans quel demi-espace cette forme se trouve, et donc quelle classe il faut lui attribuer.

III.1.3 Principe du SVM pour la régression

Les SVM peuvent également être mis en œuvre en situation de régression, c'est-à-dire pour l'approximation de fonctions quand Y est quantitative. Dans le cas non linéaire, le principe consiste à rechercher une estimation de la fonction par sa décomposition sur une base fonctionnelle. La forme générale des fonctions calculées par les SVM se met sous la forme :

$$\varphi(x, w) = \sum_{i=1}^{\infty} w_i v_i(x)$$

Le problème se pose toujours comme la minimisation d'une fonction coût, mais, plutôt que d'être basée sur un critère d'erreur quadratique (moindres carrés), celle-ci s'inspire des travaux de Huber sur la recherche de modèles robustes et utilise des écarts absolus.

On note $|i|_\varepsilon$ la fonction qui est paire, continue, identiquement nulle sur l'intervalle $[0, \epsilon]$ et qui croit linéairement sur $[\epsilon, +\infty]$. La fonction coût est alors définie par :

$$E(w, \gamma) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \varphi(x_i, w)|_\varepsilon + \gamma \|w\|^2$$

Ou γ est, comme en régression *ridge*, un paramètre de régularisation assurant le compromis entre généralisation et ajustement. De même que précédemment, on peut écrire les solutions du problèmes d'optimisation.

Les points de la base d'apprentissage associés à un coefficient non nul sont nommés vecteurs support.

Dans cette situation, les noyaux k utilisés sont ceux naturellement associés à la définition de bases de fonctions. Noyaux de Spline ou encore noyau de Dirichlet associé à un

développement en série de Fourier sont des grands classiques. Ils expriment les produits scalaires des fonctions de la base.

III.1.4 Exemple d'application

Les machines à vecteurs support ont assez d'applications dans le domaine de traitement de la parole [7], reconnaissance des formes [8], ou traitement des textes manuscrites.

Les auteurs dans [9] ont utilisés les séparateurs à marge maximale avec fonction noyau d'espace intermédiaire pour estimer le taux de qualité de l'aire. Il s'agit d'une estimation d'une variable quantitative. L'objectif de leur étude est de construire un modèle de régression de la qualité de l'air dans l'agglomération d'Avilés (Espagne) à l'échelle locale.

III.2 Le perceptron multi couche

III.2.1 Définition

Le perceptron multi couches (PMC) ou en anglais Multi Layer Perceptron (MLP) nommé aussi réseau de neurones artificielle, est un algorithme d'apprentissage supervisé qui apprend une fonction $f(.): \mathbb{R}^m \rightarrow \mathbb{R}^o$ sous un entrainement à l'aide d'une collection des données, avec m c'est le nombre de dimensions d'entrée et o le nombre de dimensions de sortie. Étant donné un ensemble de caractéristiques X_i et une cible Y, il peut apprendre une approximation de fonction non linéaire pour la classification ou la régression.

Le PMC s'organise en couches, chaque couche contient un groupement des neurones sans connexion les uns avec les autres, il reçoit un vecteur d'entrée et le transforme en vecteur de sortie. Le PMC a au moins 2 couches, une caché plus la couche de sortie.

La première version du réseau de neurones était le perceptron, il s'agit d'un modèle monocouche qui a était conçu par Frank Rosenblatt en 1958 [10]. Ce modèle était capable à simuler des fonctions simples. Cependant, en 1969 Minsky et Papert ont publié un ouvrage où ils ont montré les limites du perceptron dans des problèmes plus complexes. L'exemple le plus simple c'est le cas de la fonction XOR.

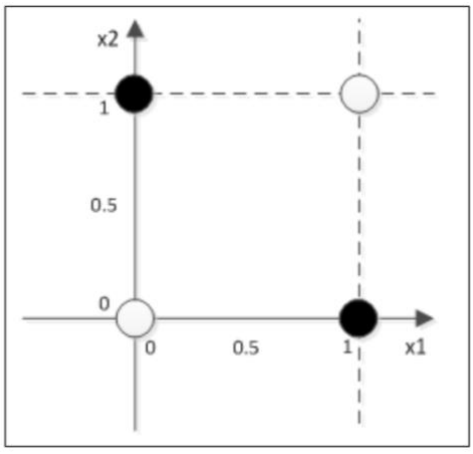


Figure 14: Les différentes valeurs de la fonction XOR

Comme le montre la figure 14, on ne peut tracer une ligne qui peut séparer les deux classes. Autrement dite, les deux classes ne sont pas linéairement séparables dans cette dimension. Ces limites-là ont poussé les recherches vers le développement de ce modèle et la création du réseau de neurones multicouches.

III.2.2 Principe général

Dans les années 1940, le neurophysiologiste Warren McCulloch et le mathématicien Walter Pits ont conçu la première implémentation mathématique d'un neurone artificiel combinant les fondements de la neuroscience et des opérations mathématiques. À cette époque, de nombreuses études étaient en cours sur la compréhension du cerveau humain et sur la manière de le simuler, mais uniquement dans le domaine des neurosciences. L'idée de McCulloch et Pits était une vraie nouveauté car elle ajoutait la composante mathématique.

De plus, considérant que le cerveau est composé de milliards de neurones, chacun interconnecté avec d'autres, ce qui donne plusieurs milliards de connexions, nous parlons d'une structure de réseau géante. Cependant, chaque unité de neurone est très simple, agissant comme un simple processeur capable de sommer et de propager des signaux.

On peut dire que l'ANN est une structure inspirée de la nature et présente donc des similitudes avec le cerveau humain. Comme le montre la figure suivante, un neurone naturel est composé d'un noyau, de dendrites et d'un axone. L'axone s'étend dans plusieurs branches pour former des synapses avec les dendrites d'autres neurones. Ainsi, le neurone artificiel a une structure similaire. Il contient un noyau (unité de traitement), plusieurs dendrites (analogues aux entrées) et un axone (analogue à la sortie [11].

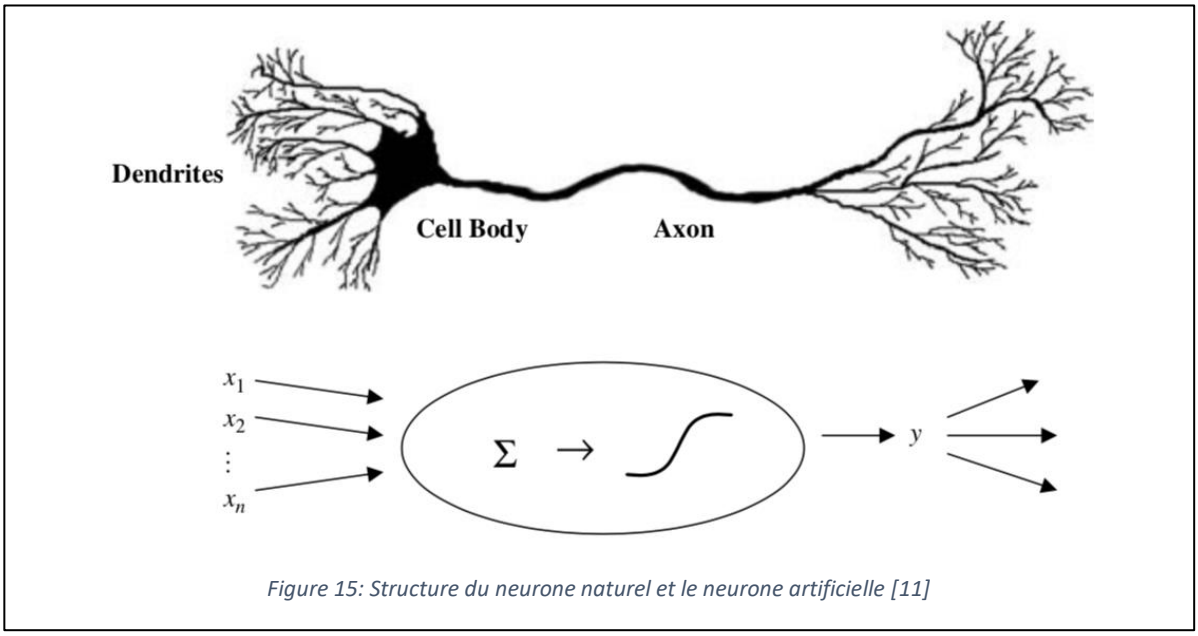


Figure 15: Structure du neurone naturel et le neurone artificielle [11]

Les neurones naturels se sont révélés être des processeurs de signaux car ils reçoivent des micro-signaux dans les dendrites qui peuvent déclencher un signal dans l'axone en fonction de leur force ou de leur amplitude. On peut alors penser à un neurone ayant un collecteur de signal dans les entrées et une unité d'activation dans la sortie pouvant déclencher un signal qui sera transmis à d'autres neurones. Donc, nous pouvons définir la structure de neurone artificielle comme indiqué dans la figure suivante.

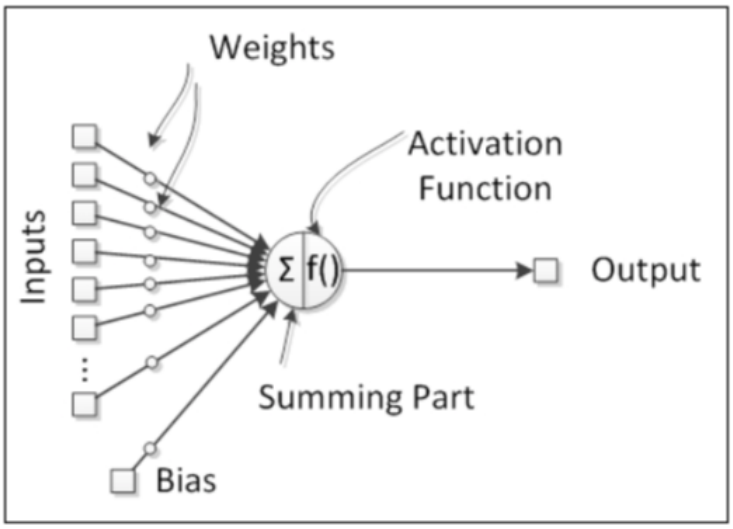


Figure 16: Structure d'un neurone artificielle

Chaque perceptron dans le réseau a des entrés x_i avec des poids w_{ji}. Les poids représentent les connexions entre les neurones et ont la capacité d'amplifier ou d'atténuer les signaux neuronaux, par exemple, de les multiplier, de les modifier ainsi. Ainsi, en modifiant les signaux du réseau neuronal, les poids neuronaux ont le pouvoir d'influencer la sortie d'un neurone. Par conséquent, l'activation d'un neurone dépend des entrées et des poids.

Le neurone artificiel peut avoir un composant indépendant qui ajoute un signal supplémentaire à la fonction d'activation. Cette composante s'appelle le biais. Tout comme les entrées, les biais ont également un poids associé.

Le neurone artificiel passe par deux phases. La première phase consiste à appliquer la somme pondérée des entrées.

$$a_j = \sum_i w_{ji} * x_i$$

La deuxième phase c'est l'application de la fonction d'activation $y_j = g(a_j)$ cette valeur sera transmise aux neurones avales. Les fonctions d'activation les plus utilisées sont :

- Sigmoïde
- Tangente hyperbolique
- Fonction à seuil
- Fonction linéaire

La figure 17 représente ces différentes fonctions et leurs représentation graphique.

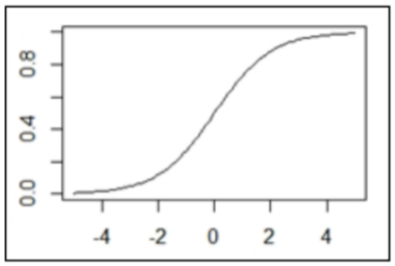
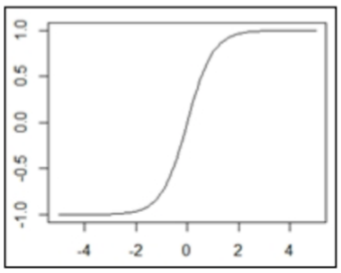
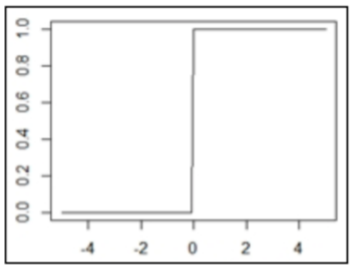
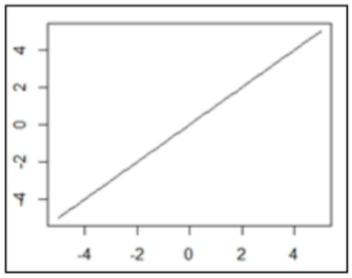
Function	Equation	Chart
Sigmoid	$f(x) = \frac{1}{1+e^{-x}}$	
Hyperbolic tangent	$f(x) = \frac{1-e^{-x}}{1+e^{-x}}$	
Hard limiting threshold	$f(x) = \begin{cases} 0 & if\ x < 0 \\ 1 & if\ x \geq 1 \end{cases}$	
Linear	$f(x) = x$	

Figure 17: Les différentes fonction d'activation du MLP

L'apprentissage du réseau de neurones s'effectue par la modification des poids jusqu'à la stabilisation du modèle (l'erreur quadratique devient minimale), et cela après plusieurs itérations. Il existe plusieurs règles de modification dont on peut citer :

- Loi de Hebb : $\Delta wji = \alpha xiyj$
- Règle de delta : $\Delta wji = \alpha(dj - yj)xi$
- Règle de Grossberg : $\Delta wji = \alpha(xi - wji)yj$

III.2.3 Exemple d'application

Les réseaux de neurones ont des applications avec un grand sucées dans plusieurs domaines tels que : l'aéronautique – le domaine médical – la vision par ordinateur – le finance etc.

Dans [12], les auteurs ont utilisé le MLP pour la prévision du taux du Ozone O_3 et le dioxyde de nitrogène NO_2 dans la région de Bilbao.

III.3 Les algorithmes génétiques

III.3.1 Définition

En informatique il y a des problèmes dont on ne trouve pas un algorithme qui peut nous donner la solution dans un temps polynomial, ces types des problèmes sont définis par la théorie de complexité autant que des problèmes NP-difficile. On va donc considérer une métaheuristiques qui va nous donner une bonne solution (proche de l'optimale) dans laps du temps raisonnable.

Parmi les algorithmes métaheuristiques il y a les algorithmes génétiques ou genetics algorithms en anglais (GA). Ce sont des programmes informatiques qui imitent les processus de l'évolution biologique pour résoudre des problèmes et modéliser des systèmes évolutifs. Décrit par John Holland pour la première fois dans les années 1960 et les a ensuite développés et ses étudiants et collègues de l'Université du Michigan dans les années 1960 et 1970.

Holland avait deux objectifs : comprendre le phénomène de « l'adaptation » tel qu'il se présente dans la nature et de trouver des moyens d'importer les mécanismes d'adaptation naturelle dans les systèmes informatiques.

Pourquoi utiliser l'évolution comme source d'inspiration pour résoudre les problèmes de calcul ? Les mécanismes d'évolution semblent bien convenir à certains des problèmes informatiques les plus pressants dans de nombreux domaines. De nombreux problèmes de calcul impliquent la recherche parmi un très grand nombre de solutions possibles. Un exemple est le problème de l'ingénierie des protéines par calcul, dans lequel un algorithme recherchera parmi le grand nombre de séquences possibles d'acides aminés une protéine aux propriétés spécifiées. Un autre exemple est la recherche d'un ensemble de règles permettant de prévoir les hauts et les bas d'un marché financier, tel que les devises. De tels problèmes de recherche

peuvent souvent tirer parti d'un usage efficace du parallélisme, dans lequel de nombreuses possibilités différentes sont explorées simultanément de manière efficace [13].

Les AGs constituent une classe de stratégies de recherche réalisant un compromis entre l'exploration et l'exploitation. Ils représentent des méthodes qui utilisent un choix aléatoire comme outil pour guider une exploration intelligente dans l'espace des paramètres codés. Ce sont des algorithmes itératifs de recherche globale dont l'objectif est d'optimiser une fonction prédéfinie appelée fonction coût ou fonction « fitness » f [14].

Les algorithmes génétiques emploient un vocabulaire emprunté à la génétique naturelle tels que gêne, chromosome, etc. Nous expliquerons intégralement ces mots le fonctionnement de l'algorithme dans la prochaine section.

III.3.2 Principe général

Selon Lerman et Ngouenet (1995) un algorithme génétique est défini par :

- Individu / chromosome / séquence : s'agit d'une solution potentielle au problème.
- Population : un ensemble de chromosomes ou de points de l'espace de recherche.
- Environnement : l'espace de recherche.
- Fonction de fitness : la fonction - positive - que nous cherchons à maximiser.

L'algorithme génétique se base sur les trois opérateurs suivant : sélection – mutation et croisement.

❖ La Sélection.

La sélection est un processus dans lequel des individus d'une population sont choisis selon les valeurs de leur fonction coût ou « fitness » pour former une nouvelle population. Les individus évoluent par des itérations successives de la sélection, appelées générations. Chaque individu est sélectionné proportionnellement à sa fonction « fitness », donc, un individu avec une fonction « fitness » plus élevée aura plus de chance d'être sélectionné qu'un autre avec une valeur de « fitness » inférieure. Cette fonction peut être envisagée comme une mesure de profit ou de qualité qu'on souhaite maximiser.

Un opérateur simple de sélection est la technique de la roulette pondérée où chaque individu d'une population occupe une surface de la roulette proportionnelle à la va- leur de sa fonction « fitness ». Pour la reproduction, les candidats sont sélectionnés avec une probabilité proportionnelle à leur « fitness ». Pour chaque sélection d'un individu, une simple rotation de la roue donne le candidat sélectionné. Cependant cette sélection n'est pas parfaite. En effet, le risque de favoriser un individu ou un petit en- semble d'individus constitue un inconvénient qui risque d'appauvrir la diversité de la population.

Un autre opérateur de sélection s'appelle sélection par tournoi. Cette technique utilise la sélection proportionnelle sur des paires d'individus, puis choisit parmi ces paires l'individu qui a le meilleur score d'adaptation.

Sélection par rang, cette technique de sélection choisit toujours les individus possédant les meilleurs scores d'adaptation, le hasard n'entre donc pas dans ce mode de sélection. En fait, si n individus constituent la population, la sélection appliquée consiste à conserver les k meilleurs individus (au sens de la fonction d'évaluation) suivant une probabilité qui dépend du rang (et pas de la fonction d'évaluation).

Sélection uniforme, cette technique de sélection se fait aléatoirement, uniformément et sans intervention de la valeur d'adaptation. Chaque individu a donc une probabilité 1/P d'être sélectionné, où P est le nombre total d'individus dans la population.

❖ La mutation.

La mutation opère sur le génotype d'un seul individu. Elle correspond, dans la nature, à une « erreur » qui se produit quand le chromosome est copié et reproduit. Dans une approche numérique, pour une chaîne binaire, elle consiste par exemple à faire pour un allèle un échange entre le « 0 » et le « 1 ». Si des copies exactes sont toujours garanties, alors le taux de mutation est égal à zéro. Cependant, dans la vie réelle, l'erreur de copie peut se produire dans diverses circonstances comme sous l'influence d'un bruit. La mutation change les valeurs de certains gènes avec une faible probabilité. Elle n'améliore pas, en général, les solutions, mais elle évite une perte irréparable de la diversité.

❖ Le croisement.

L'opérateur de croisement permet la création de nouveaux individus selon un processus fort simple. Il permet donc l'échange d'information entre les chromosomes (individus). Tout d'abord, deux individus, qui forment alors un couple, sont tirés au sein de la nouvelle population issue de la reproduction. Puis un (potentiellement plusieurs) site de croisement est tiré aléatoirement (chiffre entre 1 et l − 1). Enfin, selon une probabilité p_c que le croisement s'effectue, les segments finaux (dans le cas d'un seul site de croisement) des deux parents sont alors échangés autour de ce site (Figure 18).

Cet opérateur permet la création de deux nouveaux individus. Toutefois, un individu sélectionné lors de la reproduction ne subit pas nécessairement l'action d'un croisement. Ce dernier ne s'effectue qu'avec une certaine probabilité. Plus cette probabilité est élevée et plus la population subira de changement.

Quoi qu'il en soit, il se peut que l'action conjointe de la reproduction et du croisement soit insuffisante pour assurer la réussite de l'AG. Ainsi, dans le cas du codage binaire, certaines informations (i.e. caractères de l'alphabet) peuvent disparaître de la population. Ainsi aucun individu de la population initiale ne contient de 1 en dernière position de la chaîne, et que ce

1 fasse partie de la chaîne optimale à trouver, tous les croisements possibles ne permettront pas de faire apparaître ce 1 initialement inconnue. En codage réel, une telle situation peut arriver si utilisant un opérateur simple de croisement, il se trouvait qu'initialement toute la population soit comprise entre 0 et 40 et que la valeur optimale était de 50. Toutes les combinaisons convexes possibles de chiffres appartenant à l'intervalle [0,40] ne permettront jamais d'aboutir à un chiffre de 50. C'est pour remédier entre autres à ce problème que l'opérateur de mutation est utilisé.

Le croisement à un point, si le génotype est une chaîne binaire de longueur n. Le croisement à un point place un point de croisement au hasard. Un enfant prend une section avant le point de croisement d'un parent et prend l'autre section après le point de croisement de l'autre parent puis recombine les deux sections pour former une nouvelle chaîne binaire. L'autre enfant se construit inversement.

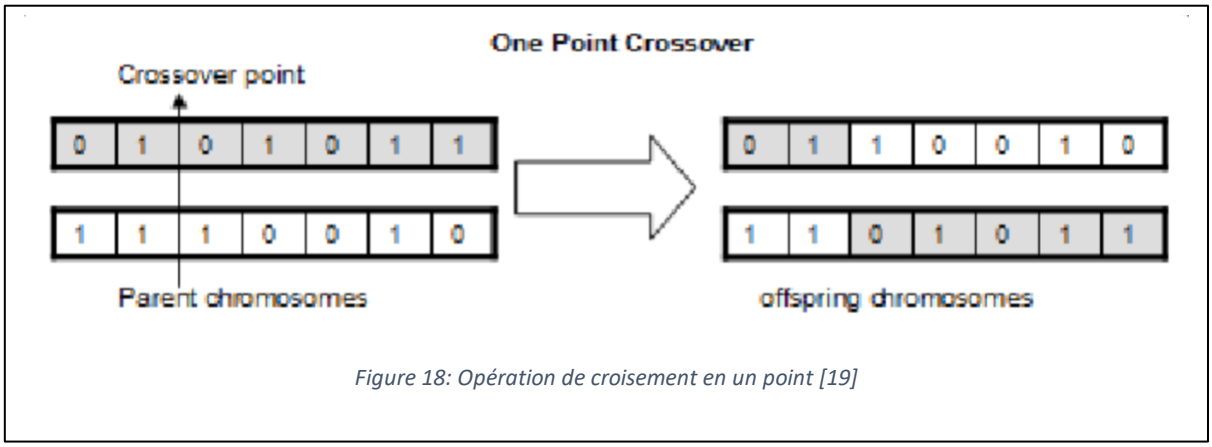


Figure 18: Opération de croisement en un point [19]

Le croisement à deux points, place deux points de croisement au hasard, et prend une section entre les points d'un parent et les autres sections en dehors des points de l'autre parent puis les recombine (Figure 19).

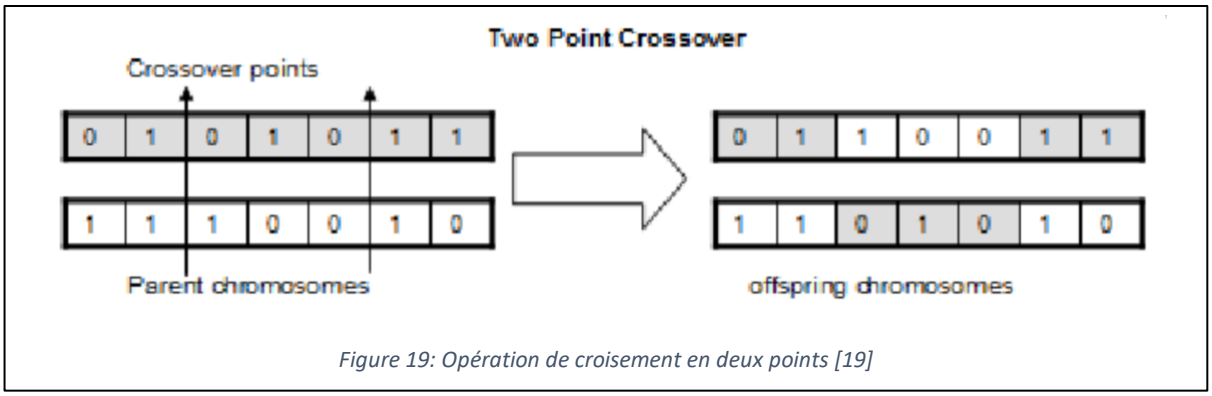


Figure 19: Opération de croisement en deux points [19]

La figure 20 résume le principe des algorithmes génétiques.

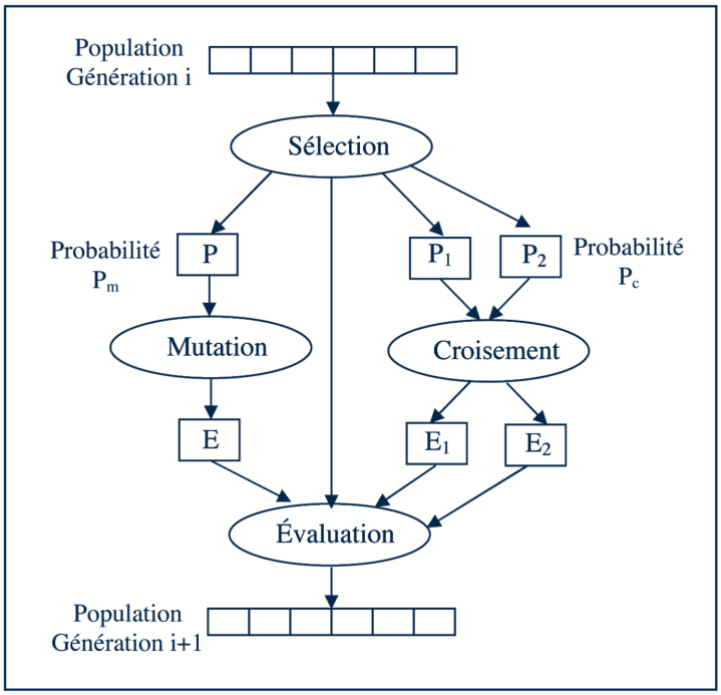


Figure 20: Principe général d'un algorithme génétique

III.3.3 Exemple d'application

Il existe plusieurs applications des algorithmes génétiques dans le domaine d'optimisation. Parmi les problèmes célèbres dont il y a une implémentation des AG on trouve : Traveling salesman problem - gestion du trafic aérien [15] etc.

Or, dernièrement on trouve les algorithmes génétiques en fusion avec des algorithmes d'apprentissage ou ce qu'on appelle des modèles hybrides, Les auteurs de [16], ont proposé une approche basée sur des algorithmes génétiques pour la discrétisation des caractéristiques et la détermination des poids de connexion pour les réseaux de neurones artificiels afin de prédire l'indice de prix des actions. Dans leur étude, GA est utilisée non seulement pour améliorer l'algorithme d'apprentissage, mais également pour réduire la complexité de l'espace des fonctions. GA optimise simultanément les poids de connexion entre les couches et les seuils de discrétisation des caractéristiques. Les poids génétiquement développés atténuent les limites bien connues de l'algorithme de descente de gradient. En outre, la discrétisation des fonctionnalités recherchées globalement réduit la dimensionnalité de l'espace des fonctionnalités et élimine les facteurs non pertinents. Les résultats expérimentaux montrent que l'approche GA du modèle de discrétisation des caractéristiques est supérieure aux deux autres modèles conventionnels.

Conclusion

Dans ce chapitre nous avons détaillé les algorithmes que nous avons implémenté dans notre solution. Nous avons commencé par les machines à vecteurs support, le perceptron multicouche et les métaheuristiques avec les algorithmes génétiques.

Chapitre IV: Implémentation et résultats

Introduction

Le dernier chapitre de ce rapport sera consacré à l'implémentation des différents algorithmes que nous avons expliquée dans le chapitre 3. De même nous utiliserons les données que nous avons décrit dans le chapitre 2.

Notre objectif c'est de créer un modèle capable à estimer la quantité de la matière organique et les matières chimiques existant dans un échantillon du sol. Pour atteindre cet objectif, nous avons commencé par l'application des machines à vecteurs support ensuite les réseaux de neurones artificiels et finalement une méthode hybride entre les réseaux de neurones et les algorithmes génétiques.

IV.1 Environnement de travail

Au cours de l'étude et du développement de ce projet, l'équipe s'est mis d'accord sur le fait de fixer l'environnement de travail : côté matériel et logiciel.

Au niveau logiciel, le choix s'est posé sur la programmation de tous les scripts avec Python 3.6.8. Ce choix était évident vu que tous les systèmes d'exploitation avec mise à jour récente acceptent cette version. Nous avons aussi évité de passer à une version plus récente du Python pour des problèmes de compatibilité avec les bibliothèques.

Au niveau matériel, il y avait un problème, MAScIR a bien ses propres serveurs et stations de travail, cependant, ils sont partagés entre les équipes, et on ne peut pas charger le serveur pour une longue durée. Alors, nous avons fait le choix d'utiliser le cloud.

Tout au long du projet, nous avons utilisé un serveur fourni par Google Cloud. Le serveur est un VPS où nous avons installé Debian 9 (Debian Stretch). La configuration du serveur est comme suite (tableau 2) :

Tableau 2: Configuration de la station de travail.

OS	Debian 9
Processeur	Intel(R) Xeon(R) CPU @ 2.30GHz
Nombre des cœurs du processeur	16
Mémoire vive	14GB
Mémoire de stockage	256 SSD

IV.2 Comparaison des modèles.

Pour comparer entre les résultats des différents modèles, nous avons utilisé deux indicateurs :

R^2 :

C'est le coefficient de détermination (prononcé R Square), il permet de mesurer la qualité de la prédiction d'une régression. Il est défini par :

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$$

Avec n est le nombre des mesures y_i la valeur de mesure n° i, $\hat{y}_i$ la valeur prédite correspondante et $\bar{y}_i$ La moyenne des mesures.

RMSE: (Root Mean square error):

L'erreur de la racine carrée, il s'agit des différences entre les valeurs prédites par un modèle ou un estimateur et les valeurs observées. RMSE est toujours non négatif, et une valeur de 0 (presque jamais atteinte dans la pratique) indiquerait un ajustement parfait aux données. En général, un RMSE inférieur est préférable à un plus élevé. RMSE est défini par :

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(predicted_i - actual_i)^2}{N}}$$

IV.3 Application du SVM

Le premier algorithme que nous avons testé c'était les machines à vecteur support. Nous rappelons que nous sommes devant un problème de régression, et vu que nous 'avons que peu des données et qui en plus, ne suivent pas une distribution normale. L'algorithme SVM a moins des paramètres par rapport aux autres algorithmes comme MLP.

Pour l'implémentation, nous avons utilisé Scikit-learn. C'est une bibliothèque open source qui contient plusieurs algorithmes d'apprentissage et fonctions statistiques.

Le modèle d'apprentissage demande la fonction noyau à implémenter comme paramètre. Nous avons le choix entre quatre fonctions *Linear – Poly – rbf – sigmoid.* Nous avons testé toutes ces fonctions d'une façon indépendante.

Pour la phase d'apprentissage, nous avons divisé les données en deux parties. Une partie ayant 80% pour l'apprentissage et 20% pour le test. La division est aléatoire.

L'implémentation de l'algorithme SVM pour la régression dans le cas des données brutes et prétraitées a donné les résultats suivants :

Le temps d'apprentissage pour le SVM avec le noyau linéaire est très petit. Le taux du R^2 est moyenne mais n'est pas encore acceptable (tableau 3). On trouve aussi des valeurs observées non estimé par le modèle (Figure 21).

Tableau 3: Résultats du SVM régression avec le noyau linéaire et données prétraitées.

Temps d'apprentissage	0.055s	
Résultats d'apprentissage	R^2	0.59
	RMSE	1784.42
Résultats du test	R^2	0.63
	RMSE	1568.66

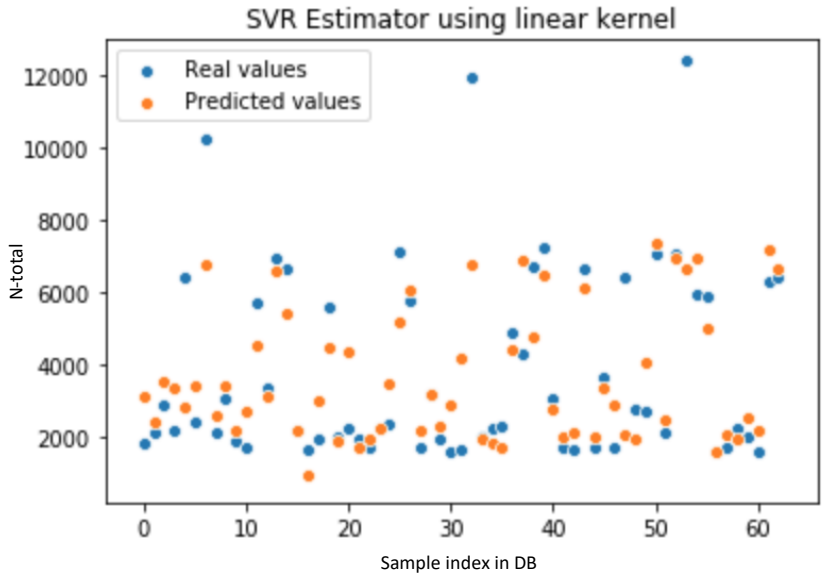


Figure 21: Comparaison entre les valeurs réelles et les estimations SVR(noyau Linear et données prétraitées).

Le noyau poly prend assez du temps, et ne donne pas des bons résultats (Tableau 4). On remarque aussi des valeurs estimes loin des valeurs observées.

Tableau 4: Résultats du SVM régression avec le noyau poly et données prétraitées

Temps d'apprentissage		1149.43s
Résultats d'apprentissage	R^2	0.32
	RMSE	2299.96
Résultats du test	R^2	-2.51
	RMSE	4898.07

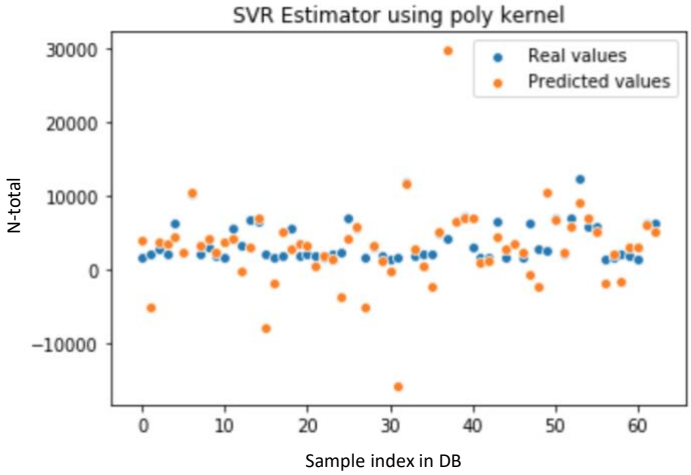


Figure 22: Comparaison entre les valeurs réelles et les estimations SVR(Noyau Poly et données prétraitées).

Les deux noyaux RBF et Sigmoïde ne convergent pas vers la solution.

Tableau 5: Résultats du SVM régression avec le noyau RBF et données prétraitées

Temps d'apprentissage		0.035s
Résultats d'apprentissage	R^2	-0.24
	RMSE	3121.21
Résultats du test	R^2	-0.17
	RMSE	2835.44

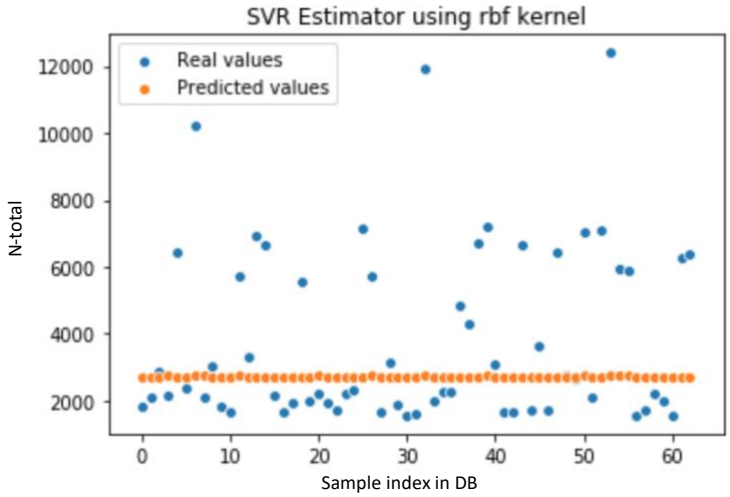


Figure 23: Comparaison entre les valeurs réelles et les estimations SVR(Noyau RBF et données prétraitées).

Tableau 6: Résultats du SVM régression avec le noyau Sigmoïde et données prétraitées

Temps d'apprentissage	0.034s	
Résultats d'apprentissage	R^2	-0.25
	RMSE	3122.79
Résultats du test	R^2	-0.18
	RMSE	2836.53

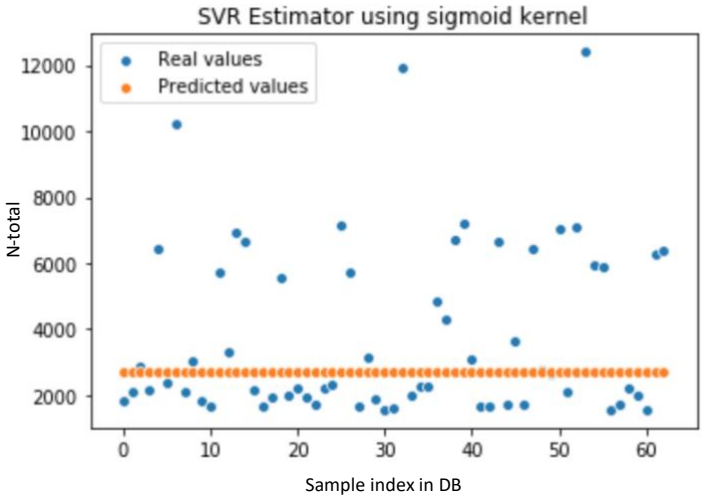


Figure 24: Comparaison entre les valeurs réelles et les estimations SVR(Noyau Sigmoïde et données prétraitées).

L'implémentation de l'algorithme SVM pour la régression dans le cas des données brutes a donné des résultats presque dans le cas des données prétraités.

Tableau 7: Résultats du SVM régression avec le noyau Linéaire et données brutes

Temps d'apprentissage	0.11 s	
Résultats d'apprentissage	R^2	0.61
	RMSE	1662.93
Résultats du test	R^2	0.58
	RMSE	1841.74

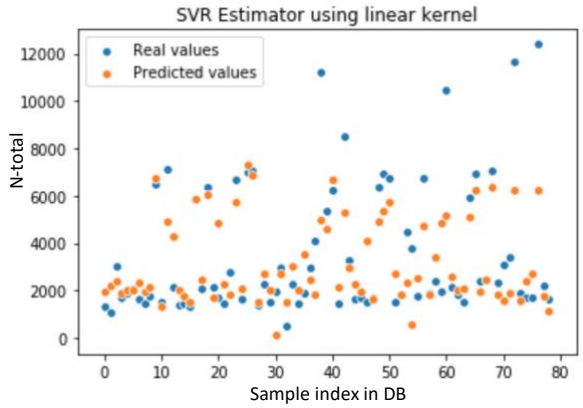


Figure 25: Comparaison entre les valeurs réelles et les estimations SVR(Noyau Linéaire et données brutes).

Tableau 8: Résultats du SVM régression avec le noyau Poly et données brutes

Temps d'apprentissage	1142.80 s	
Résultats d'apprentissage	R^2	0.08
	RMSE	2558.45
Résultats du test	R^2	-0.53
	RMSE	3533.38

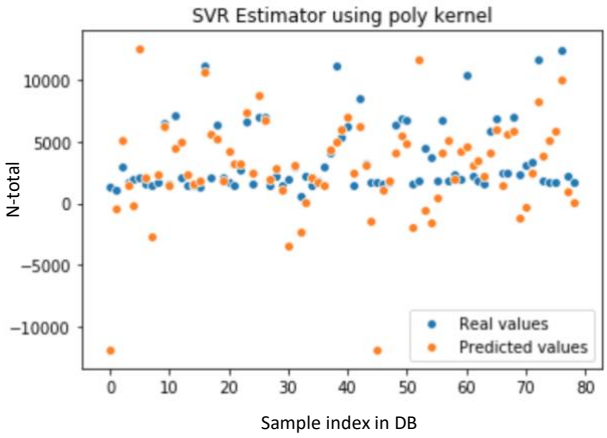


Figure 26: Comparaison entre les valeurs réelles et les estimations SVR(Noyau Poly et données brutes).

Tableau 9: Résultats du SVM régression avec le noyau RBF et données brutes

Temps d'apprentissage	0.054 s	
Résultats d'apprentissage	R^2	-0.25
	RMSE	2997.04
Résultats du test	R^2	-0.25
	RMSE	3198.79

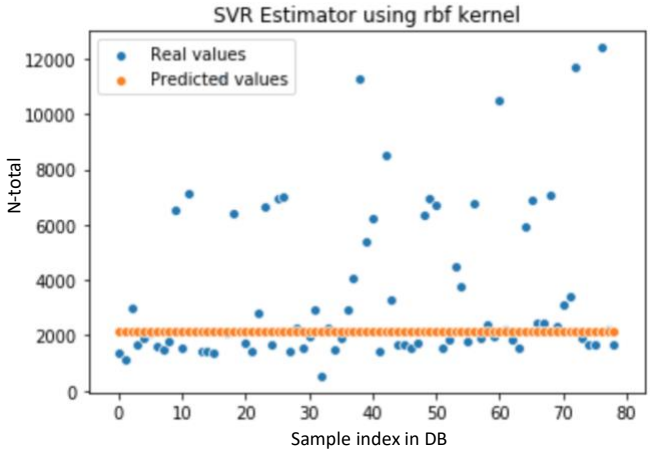


Figure 27: Comparaison entre les valeurs réelles et les estimations SVR(Noyau RBF et données brutes).

Tableau 10: Résultats du SVM régression avec le noyau Sigmoïde et données brutes

Temps d'apprentissage	0.052s	
Résultats d'apprentissage	R^2	-0.26
	RMSE	2998.68
Résultats du test	R^2	-0.26
	RMSE	3199.89

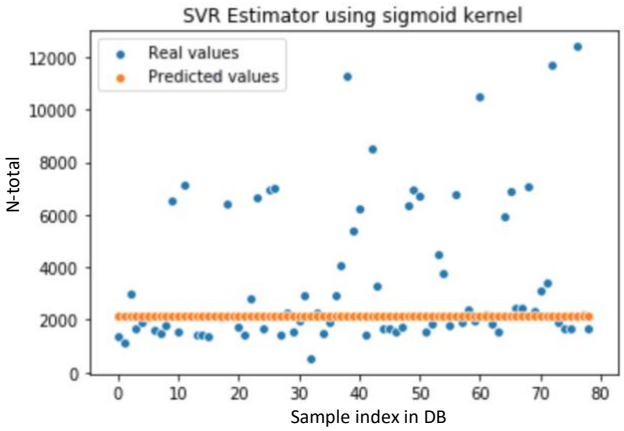


Figure 28: Comparaison entre les valeurs réelles et les estimations SVR(Noyau Sigmoïde et données brutes).

D'après les résultats obtenus dans les deux cas, données brutes et prétraités, nous avons conclu que l'implémentation d'un modèle avec SVM n'est pas faisable.

IV.4 Application du MLP

L'application du SVM n'a pas donné des bons résultats. Le deuxième algorithme que nous avons choisi c'est le perceptron multicouche. D'après l'état de l'art, le MLP donne des résultats mieux que le SVM. Cependant, le MLP nécessite plus de temps dans la phase d'apprentissage et plus de paramètres à indiquer au modèle.

Une autre contrainte exiger par le MLP c'est la normalisation des données. Cette contrainte permet d'aider le MLP à converger rapidement.

IV.4.1 Normalisation des données

La normalisation (également appelée normalisation du score z) transforme les données de sorte que la distribution résultante ait une moyenne de nulle (=0) et un écart type unitaire (=1). À part, le cas du MLP, la normalisation est obligatoire dans le cas où les données ont des unités différentes ou la variance est grande.

La formule de normalisation est comme suite :

$$x' = \frac{x - x_{mean}}{\sigma}$$

Avec : x le vecteur initial, x_{mean} La valeur moyenne du vecteur, et σ l'écart type.

IV.4.2 Réglage des hyper paramètres

Le problème du réseau de neurones c'est les paramètres. Un bon choix donne de bons résultats, mais ce n'est pas toujours évident. Le choix du nombre des couches cachées, le nombre des neurones dans chaque couche, la fonction d'activation, le nombre d'itérations, etc. est très délicat.

La solution de ce problème est d'utiliser l'algorithme GridSearch. Il s'agit d'un processus consistant effectuant un réglage hyper-paramètre afin de déterminer les valeurs optimales pour un modèle donné. Ceci est significatif, car les performances de l'ensemble du modèle sont basées sur les valeurs des hyper paramètres accordé.

Le GridSearch fonctionne comme suit : premièrement on donne le modèle souhaité, et les paramètres du modèle comme un dictionnaire (tableau associatif en python clé-valeur). Au lieu d'une valeur par clé on donne un tableau des valeurs. L'algorithme va créer des combinaisons de ces paramètres et donnera les paramètres ayant les bons résultats sur le modèle.

Les paramètres que nous avons donnés à l'algorithme :

```
parameters = {
    'activation': ['logistic', 'relu'], 'solver': ['lbfgs', 'adam'], 'max_iter': [10000, 150000, 1000000],
    'hidden_layer_sizes': [(100,), (200, 100), (200, 200, 100), (400, 150), (1000, 500)]
}
```

Après 16 heures de calcul, l'algorithme nous a donné les paramètres suivant autant que meilleurs paramètres pour le MLP.

{'activation': 'logistic', 'hidden_layer_sizes': (100,), 'max_iter': 10000, 'solver': 'lbfgs'}

IV.4.3 Résultats

L'implémentation du MLP était avec la bibliothèques Scikit-learn et le modèle MLPRegressor. Les paramètres sont celles déterminés par l'algorithme GridSearch. Les données standardisées ont donné les résultats suivants (Tableau 11) :

Tableau 11: Résultats du MLP.

		Données prétraitées	Données brutes
Temps d'apprentissage		76.10s	74.95s
Résultats d'apprentissage	R^2	0.95	0.93
	RMSE	603.21	723.69
Résultats du test	R^2	0.81	0.76
	RMSE	1223.09	1385.06

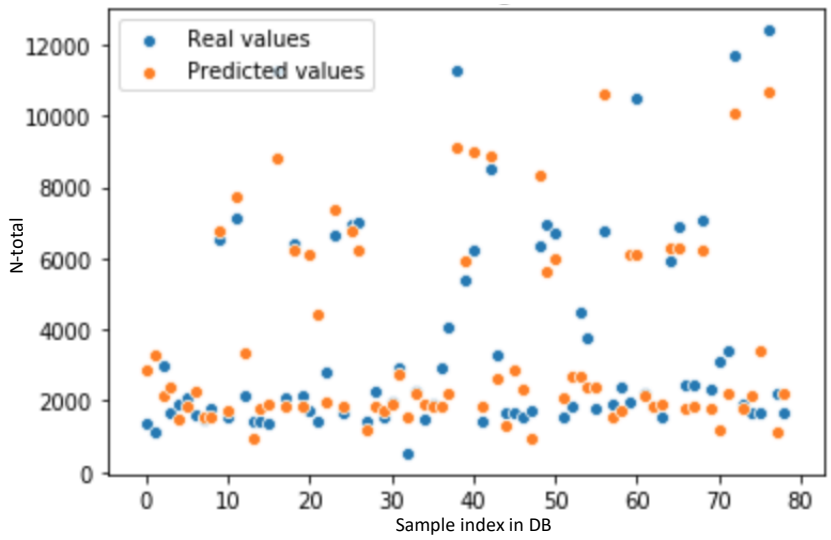


Figure 29: Comparaison entre les valeurs réelles et les estimations (données brutes)

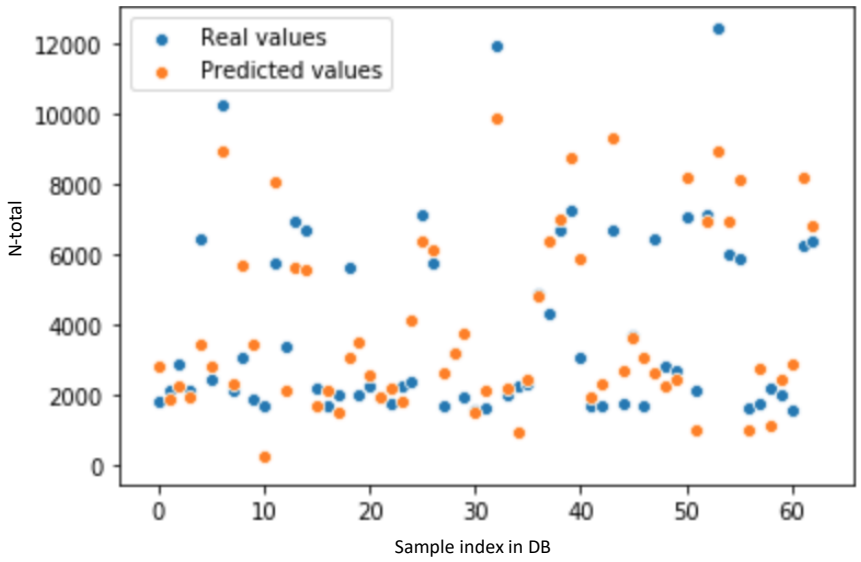


Figure 30:Comparaison entre les valeurs réelles et les estimations (données prétraitées)

Le temps d'apprentissage du MLP reste acceptable. Aussi, le taux du R^2 est très satisfaisant. Cependant, le modèle estime mal les données du test.

IV.5 Application du MLP avec fenêtre glissante

Après l'implémentation du MLP nous avons eu une amélioration de 61%. Nous voulons améliorer ce résultat en revenant à la théorie de spectroscopie.

Le SPIR donne le taux d'absorption par longueur d'onde. Autrement dit, au changement de la bande, certaines molécules dans le sol absorbent la lumière de cette longueur d'onde et d'autres non. Or, ici nous avons essayé d'estimer le N-total avec tout le spectre, et le spectre contient l'information de plusieurs matières chimiques. D'une autre manière, on donne du bruit au modèle, des informations qui n'ont aucune relation avec la sortie Y (N-total).

De ce raisonnement vient l'idée de diviser le spectre à des fenêtres. Nous essayons de trouver l'intervalle des longueurs d'onde dont N-total est actif, et d'utiliser cet intervalle pour prédire la valeur du N-total.

IV.5.1 Définition de la méthode.

L'algorithme des fenêtres glissantes est comme suite :

1. On commence par l'indice 0 et on sélectionne un nombre x des longueurs d'onde.
2. L'intervalle sélectionné sera donné au MLP pour voir le résultat.
3. On ajoute un pas y<x, et on sélectionne encore une fois un nombre x des longueurs d'onde

Cette opération s'effectue jusqu'à la dernière longueur d'onde. Nous rappelons que nous avons 647 longueurs d'onde.

IV.5.2 Résultats

L'implémentation du MLP et les fenêtres glissantes avec les paramètres déterminés par l'algorithme GridSearch a donné les résultats figure 31. Les données sont prétraitées et standardisées.

Nous avons implémenté une fenêtre de taille 50 bandes, et une incrémentation de 30 bandes. Le meilleur score (données test) qu'on a pu avoir avec les fenêtres glissantes est $R^2 = 0.61$ dans l'intervalle [90-140[.

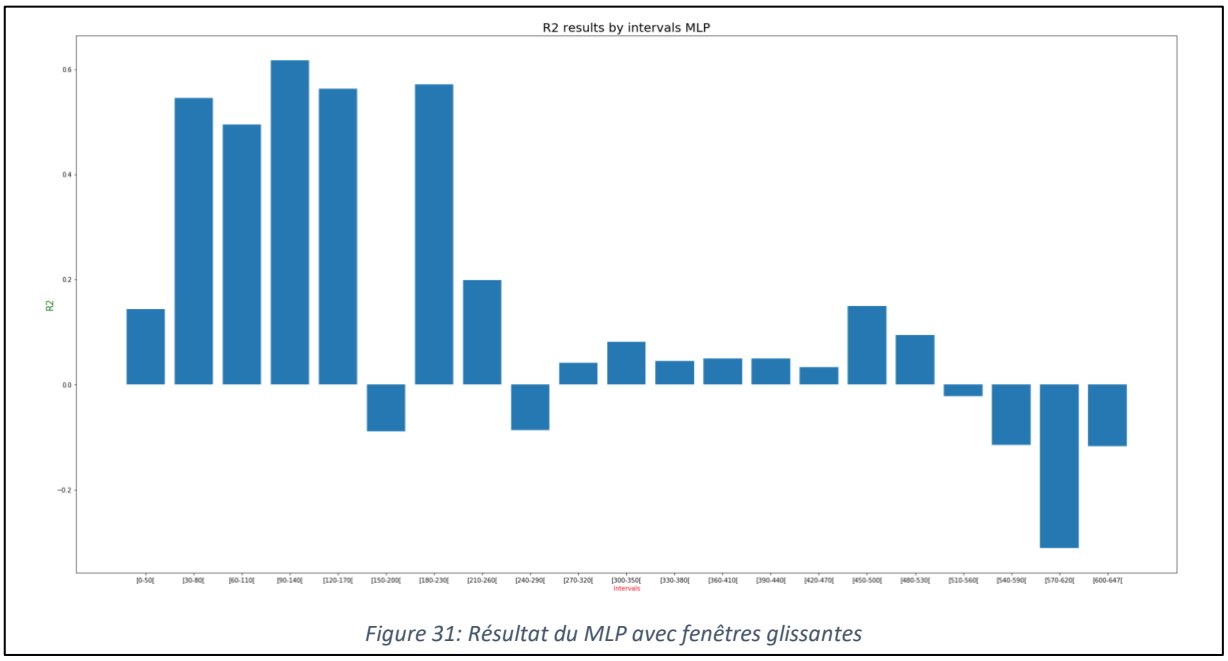


Figure 31: Résultat du MLP avec fenêtres glissantes

IV.6 Application du MLP avec les algorithmes génétiques

Après l'application des fenêtres glissantes, nous avons conclu que les longueurs d'onde régissant sur N-total ne sont pas successives. Donc plusieurs longueurs d'onde dans tout le spectre influencent sur cette valeur. La solution c'est de concevoir un algorithme qui permet de choisir différentes longueurs d'onde du spectre d'une façon aléatoire et de nous donner la suite des longueurs d'onde où le N-total est actif.

La solution c'est l'optimisation par les métaheuristiques, et nous avons choisi les algorithmes génétiques vu leurs efficacités et leurs simplicités.

L'algorithme est écrit en Python et la bibliothèque numpy.

IV.6.1 Codage

Le codage que nous avons choisi est un codage réel. Chaque gêne représente l'indice d'une longueur d'onde. Les indices sont dans l'intervalle 0 et 646. La figure suivante représente un exemple d'un individu de 10 gênes.

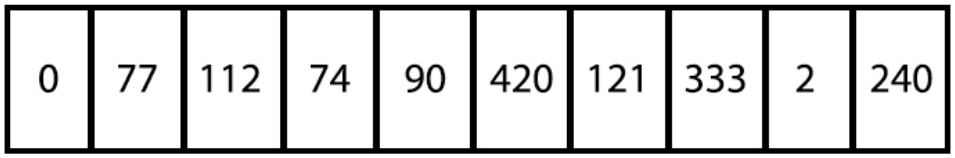


Figure 32: Exemple d'un individu de 10 gênes

IV.6.2 Génération de la population initiale

La population initiale contient ***n*** individus générer d'une façon aléatoire, mais sans redondances des individus.

IV.6.3 Sélection

Après l'opération d'initiation de la population vient l'opération de sélection. Dans cette étape on sélectionne un certain nombre ***s*** avec la probabilité de sélection (Ps) des individus sur lesquelles on effectuera les opérations de croisement et de la mutation. Les autres individus n'auront aucun changement.

Dans cette phase on implémente une sélection uniforme. Tous les individus ont la même probabilité d'occurrence.

IV.6.4 Mutation

Après la sélection, on prend ***m*** individus de la population ***s*** avec une probabilité (Pm). Le choix toujours est aléatoire. Ces individus passeront à la phase de mutation.

Dans cette phase on prend un individu de la population ***m,*** et on change une gêne aléatoirement avec une autre valeur.

Chaque individu de la population ***m,*** nous génère un individu enfant.

IV.6.5 Croisement

Pour le croisement, on prend ***c*** individus de la population ***s*** avec une probabilité (Pc). Ces individus passeront à la phase de croisement.

Dans cette phase on prend 2 individus aléatoirement, et on génère 2 points pour faire un croisement en 2 points.

Pour des contraintes de programmation, on génère les deux points de cette façon : premièrement on génère le premier 1 point d'une façon que le point soit entre la médiane du vecteur et avant sa fin. Le deuxième point se génère à partir du premier avec la formule (taille du vecteur – le premier point), automatiquement le deuxième point sera entre 1 et avant la médiane.

IV.6.6 Fonction de fitness

Pour la fonction de fitness, nous utiliserons le R^2 résultant de la prédiction du MLP. Chaque individu donnera une suite des longueurs d'onde, ces longueurs d'onde seront passées au MLP pour l'apprentissage et il retournera la valeur du R^2.

Nous cherchons à maximiser cette valeur. L'individu ayant le meilleur R^2 contiendra les longueurs d'ondes représentantes la valeur du N-total.

IV.6.7 Reproduction

Après la phase de mutation et croisement, nous aurons de nouveaux individus, et pour l'itération suivante il faut revenir à la taille ***n*** de la population initiale. Dans cette phase on sélectionne les ***n*** meilleurs individus qui passeront à l'itération suivante, et cela à base de leurs valeurs R^2.

IV.6.8 Résultats

L'implémentation du modèle hybride MLP et GA prend beaucoup du temps et aussi les ressources.

Nous avons écrit l'algorithme d'une façon qu'il enregistre le résultat de chaque itération dans un fichier log. Aussi en cas d'erreur, il enregistrera l'exception dans un autre fichier, et continuera les itérations.

L'algorithme est exécuté avec les paramètres suivants :

- Taille de la population initiale (***n***) : 100
- Taille de l'individu : 10,25,100
- Nombre d'itérations : 1000
- Probabilité de sélection (Ps) : 0.8
- Probabilité de mutation (Pm) : 0.3
- Probabilité de croisement (Pc) : 0.7

Au cours d'exécution de l'algorithme, la charge du CPU était de 99.96%, et chaque itération prend environ 30min. La figure suivante représente une partie du fichier log.

```
[2019-05-28 12:26:12] DEBUG: <!> Data loaded and standardized
[2019-05-28 12:26:12] DEBUG: Individual size: 10
[2019-05-28 12:26:12] DEBUG: Genetics algorithm object initialized
[2019-05-28 12:26:12] DEBUG: Initial population generated
[2019-05-28 12:26:12] DEBUG: Iteration number: 0
[2019-05-28 12:26:12] DEBUG: Selection with parameters ps: 0.8 - pm: 0.3 - pc: 0.7
[2019-05-28 12:54:22] DEBUG: Top fitness: 0.7999981261906769
[2019-05-28 12:54:22] DEBUG: Top individual: [131 331   1 236 487  49 210 302  34 531]
[2019-05-28 12:54:22] DEBUG: Iteration number: 1
[2019-05-28 12:54:22] DEBUG: Selection with parameters ps: 0.8 - pm: 0.3 - pc: 0.7
[2019-05-28 13:23:53] DEBUG: Top fitness: 0.8051174664541616
[2019-05-28 13:23:53] DEBUG: Top individual: [155 178 561 110 189  80 276 343 139  38]
[2019-05-28 13:23:53] DEBUG: Iteration number: 2
[2019-05-28 13:23:53] DEBUG: Selection with parameters ps: 0.8 - pm: 0.3 - pc: 0.7
[2019-05-28 13:54:52] DEBUG: Top fitness: 0.8297773803273083
[2019-05-28 13:54:52] DEBUG: Top individual: [302  34 531 236 487  49 210 131 331   1]
[2019-05-28 13:54:52] DEBUG: Iteration number: 3
[2019-05-28 13:54:52] DEBUG: Selection with parameters ps: 0.8 - pm: 0.3 - pc: 0.7
[2019-05-28 14:26:23] DEBUG: Top fitness: 0.8199689440151424
[2019-05-28 14:26:23] DEBUG: Top individual: [145 330  26 216 645  12 589 418 414 353]
[2019-05-28 14:26:23] DEBUG: Iteration number: 4
[2019-05-28 14:26:23] DEBUG: Selection with parameters ps: 0.8 - pm: 0.3 - pc: 0.7
[2019-05-28 14:58:56] DEBUG: Top fitness: 0.8329665440455614
[2019-05-28 14:58:56] DEBUG: Top individual: [514 147  37 236 487  49 210  48 413 346]
[2019-05-28 14:58:56] DEBUG: Iteration number: 5
[2019-05-28 14:58:56] DEBUG: Selection with parameters ps: 0.8 - pm: 0.3 - pc: 0.7
[2019-05-28 15:32:12] DEBUG: Top fitness: 0.8217615428296351
[2019-05-28 15:32:12] DEBUG: Top individual: [155  26 561 110 189  80 276 343 139  38]
```

Figure 33: Partie du fichier log de l'exécution de l'algorithme génétique

Après 48h d'exécution, nous avons pris l'individu ayant le meilleur R^2 dans le fichier log, et nous avons eu les résultats suivants (Tableau 12). Le taux du R^2 est acceptable pour les données d'apprentissage que pour les données du test. De même, le temps d'apprentissage et de l'ordre de 7s.

Tableau 12: Résultats du MLP et GA

		Données prétraitées	Données brutes
Temps d'apprentissage		6.10s	7.95s
Résultats d'apprentissage	R^2	0.97	0.96
	RMSE	429.24	482.13
Résultats du test	R^2	0.87	0.84
	RMSE	900.18	1269.27

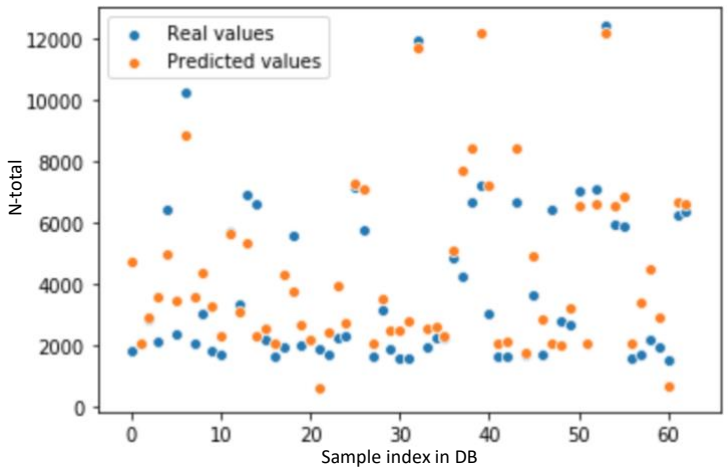


Figure 34: Comparaison entre les valeurs réelles et les estimations MLP et GA (données traitées)

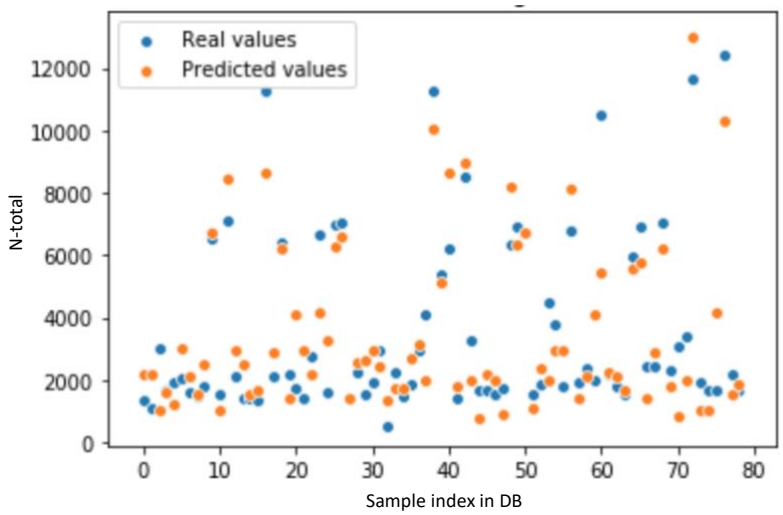


Figure 35: Comparaison entre les valeurs réelles et les estimations MLP et GA (données brutes)

Conclusion

Au cours de ce chapitre, nous avons détaillé les différentes méthodes que nous avons implémentées pour la création du modèle. En partant du SVM vers MLP et finalement nous avons proposé une méthode hybride entre MLP et les algorithmes génétiques.

Malgré le temps de calcul de MLP et le GA, nous avons pu avoir une amélioration au niveau de la performance de l'algorithme de même nous avons réduit la quantité des données requise pour l'apprentissage et le temps la suite

Conclusion générale

Le sol est considéré comme une matrice qui contient de la matière organique, des minéraux de l'eau et des gaz. Pour l'analyser et extraire la quantité de chaque matière chimique, il fallait un traitement spécial au niveau du laboratoire. Le problème dans cette analyse, c'est qu'elle prend plus de 72h et des ressources financières importantes. Notre projet de ce stage au sein de la fondation MAScIR était de proposer une solution à ce problème en utilisant les algorithmes d'apprentissage.

Nous avons proposé au cours de ce projet un modèle capable à estimer la valeur du N-totale à base du spectre proche infrarouge en utilisant le perceptron multicouche et les algorithmes génétiques.

La solution était de déterminer les longueurs d'ondes dans lesquelles les molécules objet de la mesure sont actives. Par suite nous utilisons ces longueurs d'onde pour déterminer la quantité de cette matière chimique à l'aide du MLP.

Cependant, le processus de calcul des algorithmes génétiques est très lent, comme perspectives nous chercherons à optimiser ce processus de même nous chercherons à implémenter le parallélisme sur cet algorithme et de bénéficier de la puissance GPU.

Le deuxième souci que nous avons c'est les matières chimiques non visibles par le SPIR comme le Fer (Fe) par exemple. Ce type des molécules nécessite un autre type d'analyse spectral ou au lieu d'utiliser le spectre on peut implémenter les images qu'on a présentées.

Bibliographie

[1] stanford.edu, [En ligne]. Available: http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html.

[2] L. Matrix, «Land Matrix : International Land Deals for Agriculture,» 2016. [En ligne]. Available: https://www.landcoalition.org/sites/default/files/documents/resources/land_matrix_2016_analytical_report_ii.pdf.

[3] K. H. N. a. C. G. J. R. Hart, «Determination of the Moisture Content of Seeds by Near-Infrared Spectrophotometry of Their Methanol Extracts.,» 1962.

[4] R. Rakotomalala, «Tests de normalité Techniques empiriques et tests statistiques,» [En ligne]. Available: https://eric.univ-lyon2.fr/~ricco/cours/cours/Test_Normalite.pdf.

[5] D. F. Williamson, «The Box Plot: A Simple Visual Method to Interpret Data,» *ACADEMIA AND CLINIC,* 1 6 1989.

[6] M. u. Toulouse, «Machines à vecteurs supports,» [En ligne].

[7] K. Aida-zade, A. Xocayev et S. Rustamov, «Speech recognition using Support Vector Machines,» *IEEE 10th International Conference on Application of Information and Communication Technologies (AICT),* 2016.

[8] J. W. a. C. Watkins, «Support Vector Machines for Multi-Class Pattern Recognition».

[9] P. G. N. P. R. F. J. d. C. D. F. I.-R. a. A. Suárez Sáncheza, «Application of an SVM-based regression model to the air quality study at local scale in the Avilés urban area (Spain),» *Mathematical and Computer Modelling.*

[10] F. ROSENBLATT, «THE PERCEPTRON: A PROBABILISTIC MODEL FOR INFORMATION STORAGE AND ORGANIZATION IN THE BRAIN,» *Psychological Review,* 1958.

[11] D. T.Larose, An introduction to data mining, 2005.

[12] G. I.-B. I. M. E. Agirre-Basurkoa, «Regression and multilayer perceptron-based models to forecast hourly O3 and NO2 levels in the Bilbao area,» 2005.

[13] M. Mitchell, «Genetic Algorithms: An Overview».

[14] f. t. p. B. ihsen saad, «Application des algorithmes génétiques aux problèmes d'optimisation».

[15] J.-B. G. Nicolas Durand, «Algorithmes génétiques appliqués à la gestion du trafic aérien».

[16] I. H. Kyoung-jae Kim, «Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index,» 2000.

[17] supportivy. [En ligne]. Available: https://supportivy.com/comprendre-les-boites-a-moustaches-vers-la-science-des-donnees/.

[18] B. JeanGaudart, «Comparison of the performance of multi-layer perceptron and linear regression for epidemiological data,» 2002.

[19] M. U. R. T. Yilmaz Kaya, «A Novel Crossover Operator for Genetic Algorithms: Ring Crossover,» 2012.

Proposition d'une approche pour estimer la quantité des matières chimiques et organiques dans le sol

Résumé

Ce rapport intitulé "Proposition d'une approche pour estimer la quantité des matières chimiques et organiques dans le sol" récapitule mon projet de fin d'études de mon cycle Master. L'analyse du sol se fait dans un laboratoire. En plus des ressources financières, cette analyse prend environ 72h pour avoir le bilan des quantités des matières chimiques et organiques. Ce projet consiste à faire une étude et conception d'un modèle capable à estimer la quantité des matières chimiques dans le sol et cela à base du spectre proche infrarouge et/ou les images du sol. Le but de la conception de ce modèle c'est d'avoir la possibilité de générer les bilans en temps réel. Nous avons utilisé en premier temps un modèle à base des machines à vecteurs support et un autre à base du perceptron multicouches pour estimer la quantité du N-total dans les échantillons du sol. Finalement, nous avons conçu un modèle hybride entre les algorithmes génétiques et MLP. L'implémentation des algorithmes génétiques et du perceptron multicouche nous a donné des résultats proches aux valeurs observés par rapport à l'implémentation du SVM ou MLP d'une façon indépendante.

Mots clés : sol, spectroscopie, SVM, MLP, GA

Suggesting an approach to estimate the quantity of chemical and organic matter in soil

Abstract

This report entitled "Suggesting an Approach to Estimate the Quantity of Chemical and Organic Materials in the Soil" summarizes my graduation project from my Master's degree cycle. Soil Analysis is done in a laboratory. In addition to financial resources, this analysis takes about 72 hours to have the quantities report of chemical and organic materials. This project consists of making a study and design a model able to estimate the quantity of the chemical materials in the ground and that based on the near-infrared spectrum and / or the images of the soil. The purpose of this model is to have the ability to generate the report in real time. We first used a model based on support vector machines and another based on multilayer perceptron to estimate the amount of N-total in soil samples. Finally, we designed a hybrid model between genetic algorithms and MLP. The implementation of genetic algorithms and multilayer perceptron gave us results close to the observed values compared to the SVM or MLP implementation in an independent way.

Keywords: Soil, Spectroscopy, SVM, MLP, GA

Master Systèmes Intelligents & Réseaux
Département d'Informatique
Faculté des Sciences et Techniques de Fès

A.U. 2018 - 2019