Water World

Adrien Chabert

Travail de Bachelor

Table des matières

[Table des matières 2](#_Toc12219832)

[Introduction 3](#_Toc12219833)

[Installation 5](#_Toc12219834)

[Humidité du sol 6](#_Toc12219835)

[Facteurs impactant l’humidité 6](#_Toc12219836)

[Taux d’humidité optimal 6](#_Toc12219837)

[Recherche de littérature 8](#_Toc12219838)

[ConservWater 8](#_Toc12219839)

[Autres études 9](#_Toc12219840)

[Machine Learning 10](#_Toc12219841)

[Régression 10](#_Toc12219842)

[Régression linéaire multiples méthode des moindres carrés 10](#_Toc12219843)

[Régression Ridge 10](#_Toc12219844)

[Régression de Lasso 11](#_Toc12219845)

[Elasticnet 11](#_Toc12219846)

[ARIMA 11](#_Toc12219847)

[Gradient Boosting 11](#_Toc12219848)

[Méthodologie 13](#_Toc12219849)

[Récolte de donnée 13](#_Toc12219850)

[Élimination des données inappropriées 14](#_Toc12219851)

[Choix des facteurs à fournir à notre apprentissage 15](#_Toc12219852)

[Choix de l’algorithme d’apprentissage 16](#_Toc12219853)

[Amélioration de l’algorithme 17](#_Toc12219854)

[Analyse des résultats 18](#_Toc12219855)

[Elaboration d’un programme d’arrosage 20](#_Toc12219856)

[Conclusion 21](#_Toc12219857)

[Bibliographie 22](#_Toc12219858)

# Introduction

L’eau, H2O, est à première vue une ressource abondante sur notre terre. En effet, plus de 70% de la surface de la terre est recouvert d’eau. On la retrouve sous forme liquide (mers, lacs, rivières, nappes), sous forme de glace et sous forme de gaz (dans les nuages, dans l’atmosphère). Cependant sur l’ensemble de l’eau présente sur Terre, seulement 3% de cette eau est de l’eau douce, soit 35 millions de kilomètres cubes. A titre de comparaison, le lac Léman a un volume de 89km3.

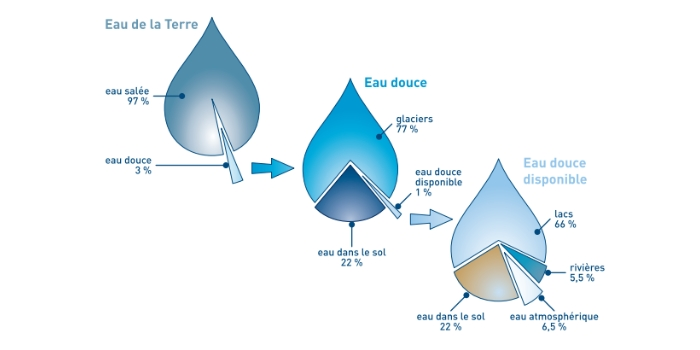


Figure 1 Eau sur Terre, http://www.septiemecontinent.com/pedagogie/lesson/eau-terre-origine-repartition/

Sur ces 3% d’eau douce 77% se trouve sous forme de glace et 22% sont profondément enfoui sous Terre. Ainsi sur l’ensemble de l’eau douce sur terre, seulement 1% de celle-ci est liquide et disponible que ce soit sous forme de lac, de rivière, dans les nappes phréatiques ou dans l’atmosphère. De plus, la Russie, l'Inde, le Canada, les États-Unis, l'Indonésie, le Congo et la Chine se partagent à eux neufs 60% du débit mondial d’eau. De ce fait, l’or bleu est inégalement réparti parmi les êtres humains sur Terre.

L’eau douce est essentielle à la vie sur Terre pour tous les êtres vivants. Pour l’Homme elle est utilisée pour s’hydrater, pour l’agriculture, pour les usages ménagers ou encore pour l’industrie. La consommation mondiale d’eau douce en 2000 était 4 km3 par an soit 1,3 millions de litres par an par habitant. Cette demande est en hausse, pour cause, sur Terre, il y a de plus en plus d’être humain.

70% de notre consommation en eau est utilisé par l’agriculture. En considérant que de nombreux pays se trouve déjà en stress hydrique une grande partie de l’année et que ce nombre ne tend pas à décroître, améliorer l’agriculture, en la rendant moins demandant en eau, est un des enjeux majeurs de l’humanité pour les prochaines années.

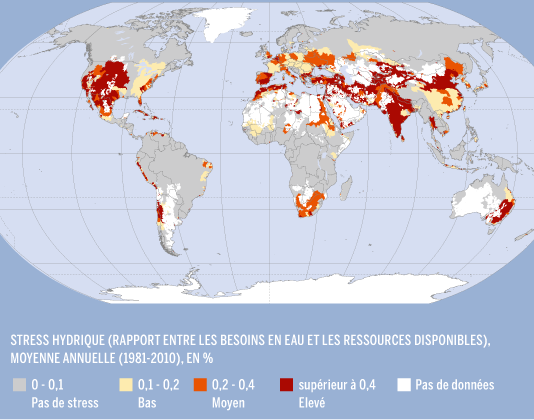


Figure 2: Carte du monde du stress hydrique, <https://www.lemonde.fr/ressources-naturelles/article/2015/03/20/la-crise-de-l-eau-illustree-en-5-graphiques_4597592_1652731.html>

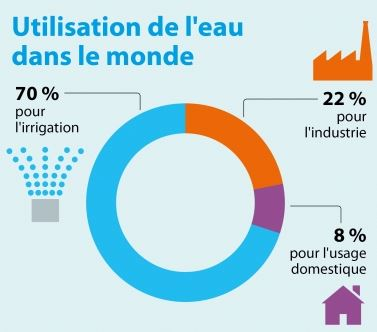


Figure 3Utilisation de l'eau, https://www.planetoscope.com/consommation-eau/239-consommation-d-eau-dans-le-monde.html

Différentes méthodes ont déjà été étudier : cultiver des espèces qui nécessite une moins grande quantité d’eau ou encore améliorer les systèmes d’irrigation. C’est ce dernier point qui nous intéresse pour mon projet. Plus précisément, je m’intéresse à comment utiliser l’eau de manière plus efficiente. Irriguer les champs avec uniquement la quantité d’eau qui est nécessaire au bien-être des plantations et de suivre un planning d’arrosage des plus efficace, permettrait d’éviter un surplus d’arrosage et donc d’un gaspillage d’eau.

Ainsi le but de mon projet est de prédire la quantité d’eau qu’il faut arroser pour que la terre reste à une certaine humidité. Une humidité qui serait idéale à l’évolution de la plante.

# Installation

Une importante phase du projet est la phase de récolte de donnée. Pour ce faire, deux bacs à plantes ont été mis à disposition avec pour chacun un arrosage automatique journalier programmable et des capteurs sensorielles. Les bacs se trouve dans des pièces fermées à côté de fenêtre. Un premier bac a été placé dans une salle bien-exposée, orienté sud-ouest. Ce bac sera nommé *Déméter*. Dans celui-ci est planté des oignons et du basilic. Le deuxième est placé dans une salle moins bien exposée, orienté nord-est et sera nommé *Cérès*. Dans ce bac, des épinards sont cultivés. Le choix de ce placement a été choisi car les épinards demandent une plus faible exposition que le basilic et les oignons. Dans les bacs, un arrosage goute à goute pour chaque pied de plante a été installé. Les capteurs sont placés à environ deux centimètres de la zone d’arrosage du goute à goute.

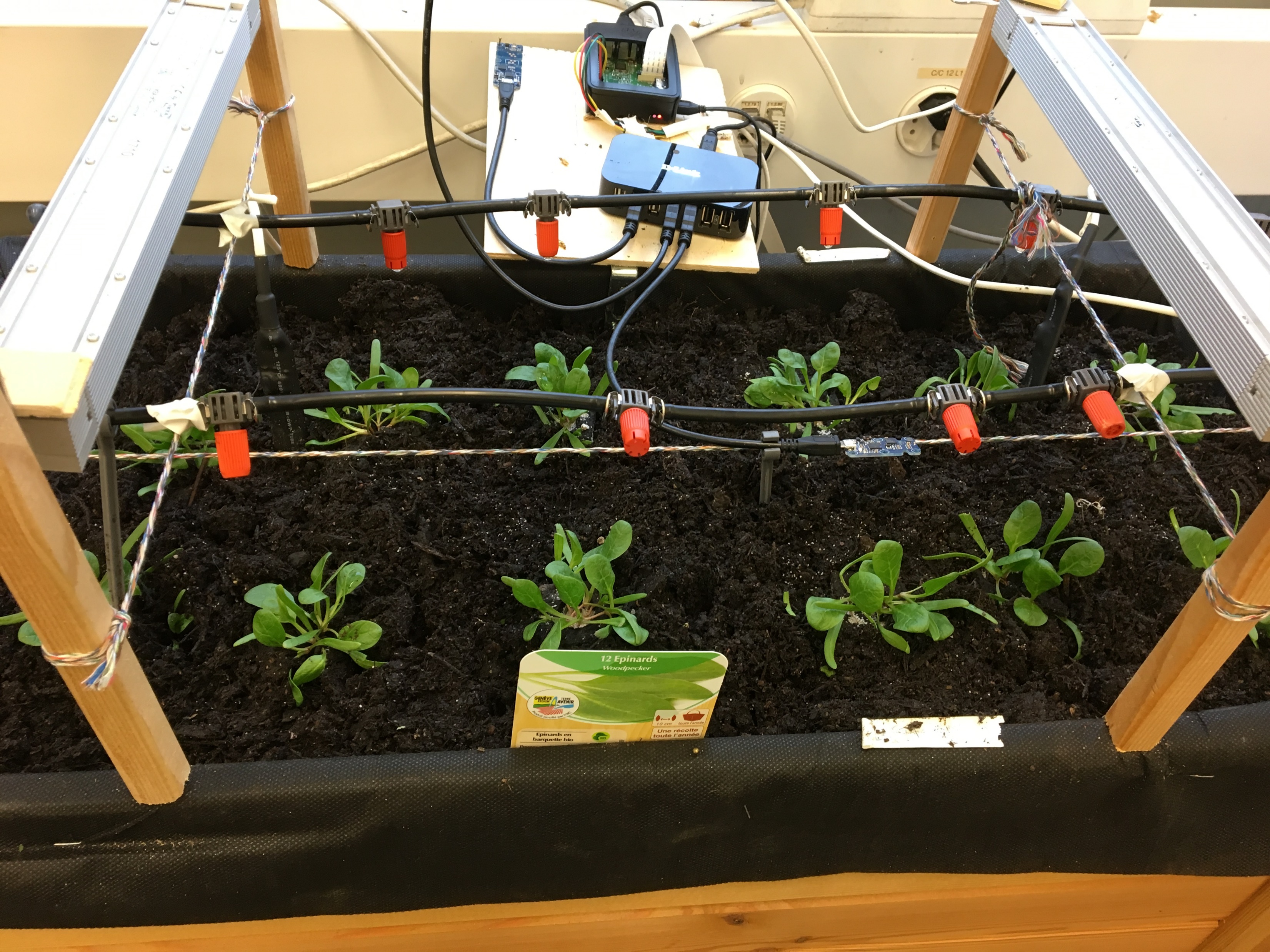


Figure 4 : Photo du bac de Cérès. On peut voir des plants d'épinard, un arrosage goûte à goûte avec des goûteurs en orange et des capteurs plantés dans la terre sur la gauche et la droite.

Chaque bac dispose de deux capteurs enterrés et d’un capteur de luminosité indépendant. Ces capteurs fournissent de très nombreuses informations utiles au calcul de l’évolution de l’humidité dans le sol. Les deux capteurs enterrés permettent de récolter des informations sur l’humidité, la température et l’ensoleillement au-dessus du capteur. Les données sont récoltées toutes les 30 secondes.

En ce qui concerne l’arrosage, des pompes permettent le transfert d’eau d’un réservoir aux goutte-à-goutte. Ces dernières s’activent à 11h pendant l’heure d’été et à midi pendant l’heure d’hiver. Il nous ait possible de définir le temps de pompage. C’est pourquoi l’arrosage ne sera pas mesuré en litre mais en seconde, qui correspondent au temps que la pompe transfert de l’eau aux goutteurs.

# Humidité du sol

Le pourcentage d’humidité dans le sol est le facteur principal de ce projet. Le but de ce projet étant de prédire la quantité d’eau à arroser pour être à une certaine quantité d’humidité après une durée déterminée. Ce taux est calculé en pourcentage. Il s’agit du rapport entre la masse d’eau présente dans le sol et la masse du sol. Mettre le capteur dans de l’eau reviendrait à obtenir une humidité de 100%.

## Facteurs impactant l’humidité

L’humidité dans le sol est influencée par différents facteurs. Selon les études, ces facteurs sont l’ensoleillement, la quantité d’eau arrosé, le vent, le type de terre, la qualité de la terre et des végétaux qui pompent l’eau dans la terre. Le vent, qui est un facteur avec une forte influence à l’extérieur, le vent assèche les terres, n’a pas été pris en compte dans notre expérience car les bacs sont à l’intérieur de salle fermé. Le type de terre n’est également pas pris en compte, car la même terre a été utilisé dans les deux bacs. Il s’agit de terre universelle qu’on peut trouver communément dans le commerce. L’ensoleillement n’a pas été pris en compte également car les capteurs ont été vite protéger du soleil par le feuillage des plantes et car l’ensoleillement est fortement corrélé à la température.

L’évolution de la plante n’a également pas été considéré car celui-ci est trop difficile à calculer et les résultats qui seront exposés par la suite montre le faible impact de ce capteur sur l’humidité pendant l’expérience.

Ainsi la température, l’humidité du sol actuel et l’évolution de la terre ont été pris en compte pour la prédiction du taux d’humidité dans le sol. Une étude de ces facteurs a été mené la partie « *Choix des facteurs à fournir à notre apprentissage »* du chapitre « *Méthodologie ».*

## Taux d’humidité optimal

Afin d’utiliser l’eau de manière plus efficace, il faut déterminer le taux d’humidité optimal à l’évolution de la plante. Celui-ci dépend du type de plante et du type de sol. Je n’ai trouvé aucun chiffre définissant le taux d’humidité optimal du sol pour une plante. Mais, d’après mes informations, le besoin en eau des oignons est relativement restreint alors que ceux du basilic et des épinards sont importants. Ainsi l’humidité du sol pour la culture d’oignon doit être en générale inférieur à celle du basilic et des épinards.

En ce qui concerne l’humidité suivant le type de sol, une plus grande documentation est disponible. Chaque type de sol a des caractéristiques différentes dépendante de sa consistance. La capacité de rétention et le point de flétrissement permanent sont des caractéristiques essentielles à la détermination l’humidité d’un sol.

La capacité de rétention correspond à la quantité maximale d’eau qu’un sol peut contenir. Ce taux peut être calculer en quantifiant la quantité d’eau deux à trois jours après un arrosage saturant le sol. En dessus de ce taux, il est inutile d’arroser car le sol est incapable de contenir de l’eau supplémentaire ce qui engendrera du gaspillage d’eau.

Le point de flétrissement permanent correspond à l’humidité du sol à laquelle une plante à l’aide de ses racines n’est plus capable d’en extraire l’eau. Le sol est trop sec. En dessous de ce taux, le sol est incapable de subvenir aux besoins des plantes. Entre ces deux taux (la capacité de rétention et le point de flétrissement) se trouve la quantité d’eau utilisable. Plus cette plage d’eau est grande, plus un sol est propice au développement de plantes. Il ne faut pas attendre le point de flétrissement pour arroser car plus on est proche de ce seuil, plus la plante aura des difficultés à s’approvisionner en eau. Pour l’arrosage par goute à goute, il est conseillé d’arroser lorsqu’il ne reste plus que 80% d’eau disponible. Alors que pour l’irrigation par aspersion, il est conseillé d’irriguer à partir de 50% d’eau disponible.

Le sol utilisé pour notre expérience est une terre universelle qui peut facilement être acheter dans le commerce. Ceci correspond à un sol de type de loam limoneux. Ainsi le but de notre logiciel sera de rester au-dessus des 25% d’humidité. La plage d’eau disponible est d’environ 15%.

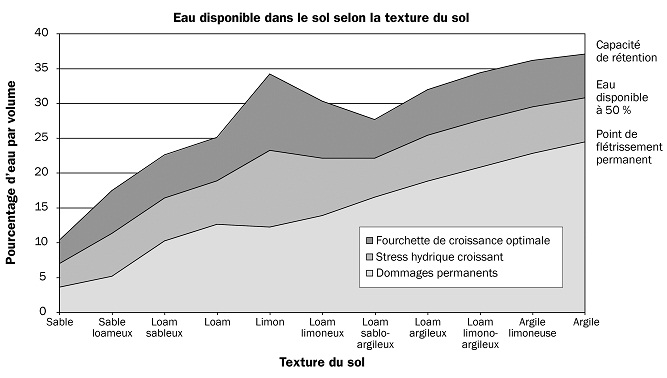


Figure 5 : Humidité optimal pour un sol. <http://www.omafra.gov.on.ca/french/engineer/facts/11-038.htm>

# Recherche de littérature

## ConservWater

Une large étude a été mené par l’institut de technologie de Californie à Pasedena et cette dernière a publié un approfondi article dans le IJOEAR (Journal international de recherche en environnement et en agriculture) nommé « *Global custom-tailored machine learning of soil water content for locale specific irrigation modeling with high accuracy »* écrit par Aadith Moorthy. Cet article fait une analyse de la situation actuelle, un comparatif des résultats de prédiction d’arrosage en utilisant différents algorithmes et propose un algorithme de Machine Learning dédié à la prédiction d’arrosage pour la culture de plants dans le monde entier. Ils ont créé un algorithme utilisant du machine Learning. Cet algorithme est nommé ConserWater™ algorithm.

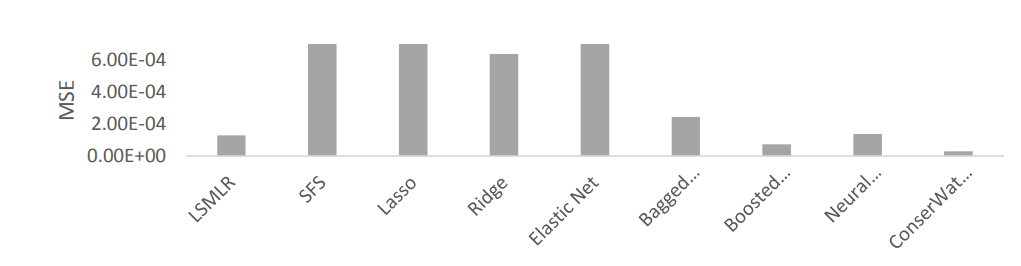
Selon leurs analyses, ils ont montré que leur algorithme se comporte mieux que des algorithmes statiques tel que Aqua Crop, proposé par Land and Water Division of the Food and Agricultural Organization (FAO). Leur algorithme obtient également de meilleurs résultats que des algorithmes de Machine Learning basé sur les itérations dans le temps tels que ARIMA et SARIMA. Ceci vient du fait que ARIMA et SARIMA aient besoin de données avec des propriétés statiques. Ceci n’est pas le cas étant que la météo ne suit aucune loi statiques (déviation standard ou moyenne). 

Figure 6: Erreur moyenne au carré (MSE) sur 30 jours pour différents algorithmes en ne donnant que l'humidité de la première journée, <https://ijoear.com/Paper-October-2016/IJOEAR-SEP-2016-34.pdf>

On peut voir que leur algorithme (ConserWater) obtient un bien meilleur résultat que d’autres algorithmes de machine Learning classique et fréquemment utilisé. Il est intéressant de noter que la régression linéaire multiple moindre carré (LSMLR) obtient de très bon résultat par rapport aux autres algorithmes de Machine Learning. Ceci est d’autant plus intéressant que c’est un algorithme avec une faible complexité d’implémentation.

A noter qu’il est important de séparer les données par rapport à leur région. L’humidité ne réagit pas de façon identique dans une région aride que dans une région tropicale.

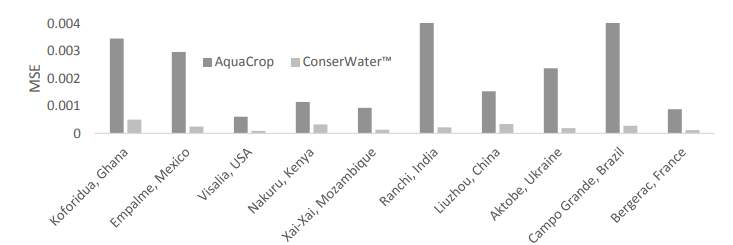


Figure 7: Erreur MSE entre Aqua Crop et ConserWater suivant les régions, <https://ijoear.com/Paper-October-2016/IJOEAR-SEP-2016-34.pdf>

L’article affirme que l’utilisation de machine Learning peut permettre une économie d’eau de 25% par rapport à l’usage habituel. Mais que l’utilisation de capteurs sensorielles installés par les maraichers peut revenir à un coût rédhibitoire pour les petites fermes. C’est pourquoi ils ont développé un algorithme utilisant les données récoltées par les nombreux capteurs météorologique déjà existant, notamment en utilisant le *World Weather online service*. Ils ont à disposition 30'000 stations de collecte de données. Les données ont été collecté sur un peu plus de 1 an.

Une telle étude n’est pas du tout comparable à l’étude qui a été pour ce projet. Le nombre de données collecté est amplement supérieur aux nôtres. De plus cette étude californienne a été mené en extérieur sur différents continents. Ainsi le nombre de facteur impactant l’humidité est beaucoup plus important que pour notre étude en intérieur. Il leur a fallu prendre en compte le vent, les averses, la condensation, la rosée, etc. Cependant il est intéressant de noter qu’une simple linéaire régression permet d’obtenir des résultats très satisfaisant et que l’économie possible réalisé est conséquente.

## Autres études

#### Estimating soil moisture using remote sensing data: A machine learning approach

C’est une étude réalisée en 2009 sur des données récoltées au Colorado pendant 5 ans. Il s'agit d'estimer l'humidité du sol avec les données du capteur en utilisant la technique appelée Support Vector Machine (SVM). "SVM est basé sur une théorie statistique d'apprentissage qui utilise un espace d'hypothèses de fonctions linéaires basé sur une approche par noyau." Ils ont obtenu de meilleurs résultats avec SVM qu'avec les modèles de réseau de neurones et qu’avec la régression linéaire à variables multiples.

#### Planning d’arrosage hebdomadaire avec Jojoba Israel

Jojoba est une société qui vend des capteurs d’humidité. L’entreprise a réalisé une étude sur des données récolté à l’aide de 22 capteurs sur 2 ans. Le but de leur expérience était de faire un planning hebdomadaire d’arrosage. Ils ont comparé différents algorithmes d’apprentissage et ont obtenu que le meilleur résultat était obtenu avec Gradient Boosted Regression Trees, avec un taux de précision de 93%. Ils ont jugé les résultats satisfaisants et qu’ils aidaient les ingénieur agronome à définir leur arrosage.

# Machine Learning

Les algorithmes de Machine Learning ou autrement appelé algorithme d’apprentissage sont des algorithmes qui permettent à la machine d’apprendre à partir de données en se basant sur les statistiques de ces données. On parle alors d’intelligence artificielle. Le plus connu de ces algorithmes est Alpha GO développé par DeepMind qui a permis de battre au jeu de Go des professionnels. Avant l’arrivé d’Alpha GO, le jeu de Go était considéré comme impossible à simuler pour un logiciel tant son nombre de possibilité de jeu est immense.

Le principe d’un algorithme d’apprentissage est le suivant. L’algorithme d’apprentissage reçoit une large base de données. Ces données contiennent des valeurs pour différents paramètres et un résultat pour cette donnée, on parle dans ce cas d’apprentissage supervisé. Puis à partir de cette base de données, l’algorithme construit un modèle qui lui permettra de prédire un résultat en fonction des paramètres utilisé pour l’apprentissage (paramètres qui lui ont été fourni dans la base de données). Dans le cas où il n’y a pas de résultats fournis pour chaque donnée de la base de données, on parle de d’apprentissage non supervisé. L’algorithme ne fourni pas une prédiction mais est alors capable de construire des structures sous-jacentes au données. Par exemple, on fournit une la large base de données sur des animaux, l’algorithme non-supervisé peut être capable de les séparer en fonction s’ils sont des mammifères, amphibiens, poissons, oiseaux, etc.

Pour mon travail, il faut utiliser un apprentissage supervisé étant donné qu’on veut prédire la quantité d’humidité. Il existe deux classes d’apprentissage supervisé différentes, la classification et la régression. Lors que les résultats sont discrets, par exemple le résultat est une espèce d’iris, on parle de classification. Lors que les résultats sont continues, on utilise des régressions. J’ai utilisé ces derniers car la valeur de l’humidité est continue entre 0 et 100 (on parle de pourcentage d’humidité).

## Régression

### Régression linéaire multiples méthode des moindres carrés

Il s’agit du modèle classique de régression linéaire et facilement utilisable. Il consiste à utiliser un vecteur w = (w0, w1, …, wp) qui serviront de coefficient pour prédire notre résultat.

Il faut trouver le meilleur vecteur w qui minimise l’erreur selon la formule des moindres carrés entre la prédiction ) et le résultat escompté . Plus il y de dépendance entre les facteurs, plus la méthode des moindres carrés devient très sensible. C’est pourquoi il faut faire attention aux données qu’on utilise. Cette méthode n’est pas conseillée s’il n’y a aucune vérification des données.

### Régression Ridge

Deux caractéristiques sont importantes à l’estimation de la qualité de prédiction. La première est le biais. Il s’agit de du carré de l’écart en l’espérance de la prédiction et la valeur réelle. La deuxième est la variance ou l’instabilité, c’est-à-dire, la dispersion de la prédiction. Afin de minimiser cette instabilité on ajoute une contrainte sur les coefficients. Il s’agit d’une contrainte L2 sur la taille du coefficient (voir la formule ci-dessous). Plus l'alpha est élevé, plus les coefficients sont robustes. La régression linéaire par la méthode des moindres carrés ordinaires peut conduire à un système d'équations surdéterminé ou sous-déterminé. Avec la régularisation L2, il donne la préférence à une solution particulière aux propriétés souhaitables.

### Régression de Lasso

Cette régression est à utiliser lorsque on souhaite réduire le nombre de variables dont dépend la solution donnée. Il faut effectuer une régularisation L1 sur les coefficients w. Ainsi on tend vers le principe du rasoir d’Ockham, faire un système aussi simple que possible, mais pas forcément la meilleure S'il existe une forte corrélation entre les composants, Lasso privilégiera l'un au détriment de l'autre ce qui peut nuire à la qualité du modèle.

### Elasticnet

Il s’agit d’une combinaison entre la méthode Ridge et la méthode Lasso. Il faut combiner une régression L1 et une régression L2 sur les coefficients w. Elasticnet permet de tirer les avantages des deux méthodes, c’est-à-dire la capacité de sélection des variables et exclusion des variables non pertinentes (Lasso) ainsi que le partage des poids des variables corrélés (Ridge).

### ARIMA

ARIMA (Autoregression Models for Time Series Forecasting) est un modèle de prédiction pour les séries temporelles. Cet algorithme est fait pour les systèmes où la valeur à prédire dépend d’observations effectué à des périodes précédentes. Concrètement une régression linéaire est effectuée à partir des observations précédentes.

Cette méthode est utilisée pour des données qui font preuve de propriété stochastique non statistique[[1]](#footnote-1). Ceci n’est pas le cas de notre projet. La météo ne suit aucune loi. Elle est aléatoire et indépendante du temps.

### Gradient Boosting

Le concept du boosting est le suivant. Un modèle de prédiction est construit et des prédictions sont établi. Les résultats des prédictions sont classifiant en fonction de leur difficulté à prédire, c’est-à-dire, la réussite de la prédiction. Un poids faible est affecté aux données dites de difficulté facile et un poids fort est attribué aux données dites à forte difficulté. Ensuite un nouveau modèle est créé en prenant en compte la difficulté des données dans le but d’améliorer la prédiction. Ce schéma est répété jusqu’à obtenir une prédiction correcte. Ce système permet d’identifier les faiblesses d’apprentissage et d’en améliorer celui-ci.

Le gradient boosting est légèrement différent. Ce dernier ne cherche pas à minimiser les pertes mais à optimiser une fonction fournie par l’utilisateur. Ce changement correspond mieux à l’usage du monde réel.

# Méthodologie

Afin de créer un système capable de conseiller l’arrosage sur une période donnée, il faut lui fournir un algorithme de prédiction. Pour ce faire, un algorithme d’apprentissage doit lui être fourni ainsi que des données qui permettront à celui-ci d’appendre. Plusieurs choix sont nécessaires à la réussite de l’apprentissage. Ces choix sont les suivants. Comment récolte-t-on les données ? Quelles données doivent être fourni à l’algorithme et celle qui ne doivent pas ? Quel algorithme d’apprentissage utiliser ? Comment améliorer cet algorithme ?

## Récolte de donnée

Deux capteurs d’humidité, de température et de luminosité se trouvent dans chacun des deux bacs à plantes. Chaque valeur des capteurs est saisie toutes les 15 secondes. Les données ont été récoltés sur trois mois, de mi-mars à mi-juin. Il est important de noter que les deux bacs ne sont pas exposés pareil, ne contiennent pas les mêmes plantes et les capteurs ne sont pas tous placer à la même distance d’un arrosage. L’arrosage se fait sur un point spécifique étant donné que c’est un arrosage goûte à goûte. Ainsi l’humidité du sol varie en fonction de la distance du capteur à un point d’arrosage. Les deux capteurs dans le bac Cérès et un des capteurs dans Demeter sont à une distance de 2 centimètres d’un point d’arrosage. Alors que le deuxième capteur dans Demeter est à une distance de 4 centimètres. C’est pourquoi, j’ai choisi de ne pas utiliser ce dernier capteur.

Les épinards, qui ont été planté dans le bac Cérès, sont arrivés à maturité deux mois après la plantation, c’est-à-dire mi-mai. Alors que les basilics et oignons étaient toujours en bonne santé mi-juin. Ces différentes se manifestent sur les courbes d’humidité des capteurs de leur bac respectif. Ci-dessous un graphique de l’erreur[[2]](#footnote-2) si on rassemble des deux bacs ou si on les sépare. C’est pourquoi j’ai décidé de ne pas rassembler les données des deux bacs et d’appliquer des algorithmes d’apprentissage indépendamment entre les bacs.

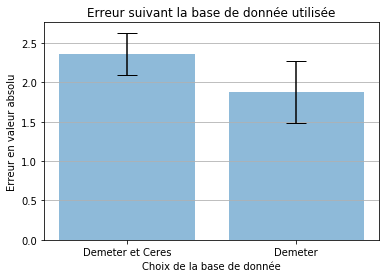


Figure 8 : Erreur en valeur absolu en fonction du choix des données utilisés.

Il est inutile de récupérer les données toutes les 15 secondes car les capteurs sont parfois imprécis et comporte du bruit. Il est préférable de récolter les données toutes les 30 minutes afin de récupérer une moyenne des facteurs sur 30 minutes. Ceci permet d’éviter l’apparition de bruit sur les valeurs obtenus. Chaque facteur suit ainsi une courbe homogène.

L’arrosage est journalier et toujours programmer à 11h pour l’heure d’hiver et midi pour l’heure d’été. Ainsi la représentation graphique de chaque journée comporte un pic à 11h ou 12h, puis une phase d’évaporation rapide de l’eau et enfin une troisième phase de stabilisation de l’humidité. La deuxième et troisième phases sont beaucoup plus marquées pour Demeter que Cérès.

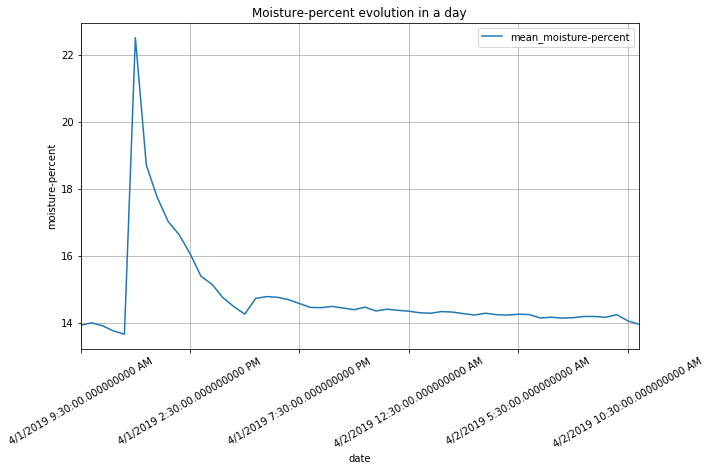


Figure 9 : Evolution classique de l'humidité dans le bac Demeter. L’arrosage pour cette journée était de 20 secondes à midi.

Toutes les journées sont séparées. Une journée commence à l’arrosage et se termine à l’arrosage suivant. Ainsi chaque journée, qui contient 48 itérations (une itération toutes les 30 minutes), est utilisé pour l’apprentissage de l’algorithme de prédiction.

Il est important d’avoir un échantillon donné d’apprentissage qui représente toutes les combinaisons d’arrosage. C’est pourquoi il faut varier la quantité d’arrosage. Un arrosage de 10, 15, 20, 35, 40 et 45 secondes[[3]](#footnote-3) a été effectué pour le bac Demeter et un arrosage de 10, 15, 20 et 30 secondes a été effectué pour le bac Cérès. Il est important de noter que x secondes pour Déméter ne correspond pas à x secondes pour Cérès. En effet x secondes de pompages d’eau chez Demeter doit approvisionner en haut 10 goûtes à goûte. Alors x secondes de pompages d’eau chez Cérès doit approvisionner 6 goûtes à goûte. Ainsi, à seconde équivalente, un arrosage chez Cérès est deux fois plus conséquent que pour Déméter.

## Élimination des données inappropriées

Des problèmes de récolte de données sont apparus pendant les 3 mois de récolte comme notamment un capteur défectueux ou un arrosage incomplet ou encore une panne avec le Raserberi qui collecte les données. C’est pourquoi il a fallu vérifier la qualité des données et supprimer celles qui étaient incomplètes ou incorrectes. Chaque journée qui comporte des itérations incomplètes ont été supprimé. Un autre moyen aurait été de faire une interpolation linéaire des données manquantes. Cependant, il est arrivé que des données manquait sur plusieurs journées ou que données manquantes étaient celle de la période d’arrosage. Ainsi il était impossible d’interpoler correctement les données manquantes sans induire en erreur notre apprentissage. Les journées avec des arrosages doubles ou incomplets ont également été supprimées. Celle-ci influençaient en erreur notre algorithme d’apprentissage.

Lors de mise en place du système, une nouvelle terre a été utilisé pour l’expérience. La terre était donc relativement humide et ne réagissait pas de la même façon à un arrosage qu’un mois après. C’est pourquoi les données des deux trois premières journées ont également été supprimés car la terre n’était pas encore suffisamment stabilisée. Graphiques ?

## Choix des facteurs à fournir à notre apprentissage

Comme mentionné précédemment, l’humidité du sol est influencée par les précipitations, par la température, la luminosité et le vent. Dans notre cas, il s’agit d’une étude en intérieur c’est pourquoi le facteur du vent est exclu. La luminosité n’est également pas prise en compte car elle est fortement corrélée par la température et nos bacs ont une très forte densité de feuillage. Ainsi, les rayons du soleil ne touchent pas directement le sol et nos capteurs. De plus, la seule précipitation perçut par les bacs à plante est l’arrosage automatique programmé quotidiennement.

Ainsi les facteurs restants sont la température, l’humidité de base dans le sol et l’arrosage. Comme on peut le voir à la figure 7, la variation de l’évolution de l’humidité dans le sol n’est pas régulière. Pendant les quatre premières après l’arrosage, une forte baisse de l’humidité est constatée. Ensuite, la baisse d’humidité est beaucoup plus faible. Ainsi j’ai jugé bon de prendre comme facteur le temps depuis le dernier arrosage et la quantité de ce dernier (*ArrosageHist)*. Important, *ArrosageHist* et *Arrosage* ne sont pas les mêmes valeurs. *Arrosage* indique la valeur de l’arrosage à une date et une heure précise. Ainsi quand on n’arrose pas, la valeur de cet arrosage est de 0. Alors que *ArrosageHist* indique la valeur du dernier arrosage effectué. Ainsi pendant toutes une journée la valeur est inchangée.

Un dernier facteur peut être également pris en compte, il s’agit du nombre de journée écoulé depuis le début de l’expérience (*Index)*. En effet, on peut imaginer que la terre, qui était à l’origine neuve, s’appauvrisse et donc qu’elle retienne moins facilement l’eau et ainsi l’humidité. Les plantes ont également crû pendant l’expérience et donc on peut penser que les plantes pompent plus d’eau à la fin de l’expérience qu’au début.

Pour rassembler toutes ces valeurs, j’ai utilisé la librairie pandas dans python qui permet d’utiliser des dataframes. Les dataframes sont très pour la manipulation de données. Elles s’apparentent à des matrices mais à la différence que les colonnes peuvent être nommées. Voici une représentation de dataframe que j’ai utilisé pendant mon travail. Elle contient tous les paramètres cités auparavant.

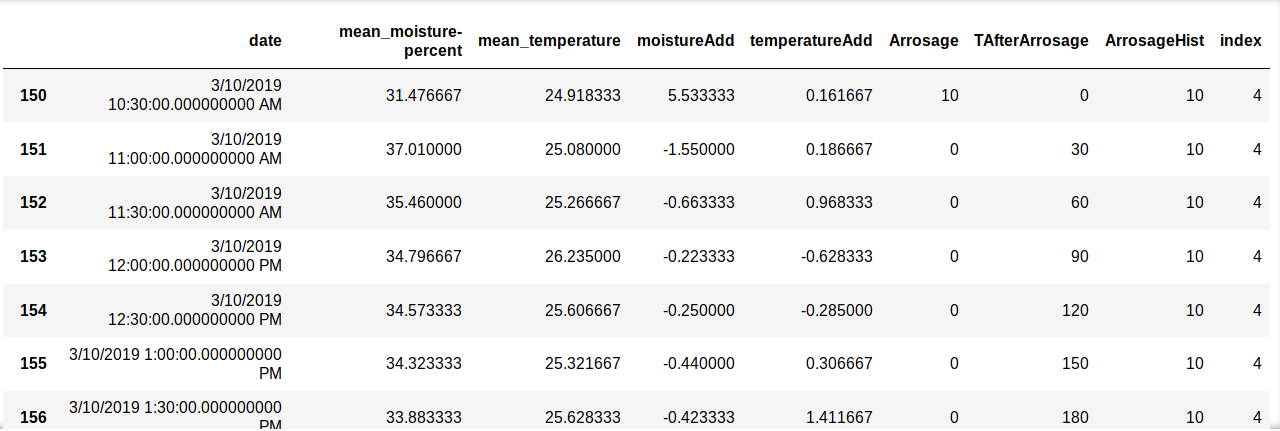


Figure 10 : Représentation d'une dataframe avec les informations sur la date, l'humidité, la température, l'arrosage et la journée de collecte de données.

Afin de déterminer l’importance de ces différents facteurs, un tableau comparatif de l’erreur obtenu en prenant en compte ces différents paramètres dans notre algorithme de machine Learning. Pour mener ce teste, j’ai choisi de prendre une régression linéaire multiple à moindre carré. J’ai fait ce choix car dans mes recherches, j’ai pu voir qu’elle permettait d’obtenir de très bons résultats. Ci-dessous se trouve un graphique montrant l’erreur obtenu en variant les facteurs pris en compte. De part mes recherche et mes résultats, la température, l’humidité de base, la quantité d’arrosage et le temps après arrosage sont des facteurs primordiaux pour une bonne précision de prédiction. Selon mes observations, le temps depuis le début de l’expérience et l’historique de l’arrosage ne permettent pas d’améliorer les performances de l’algorithme ainsi ils ne sont pas pris en compte.

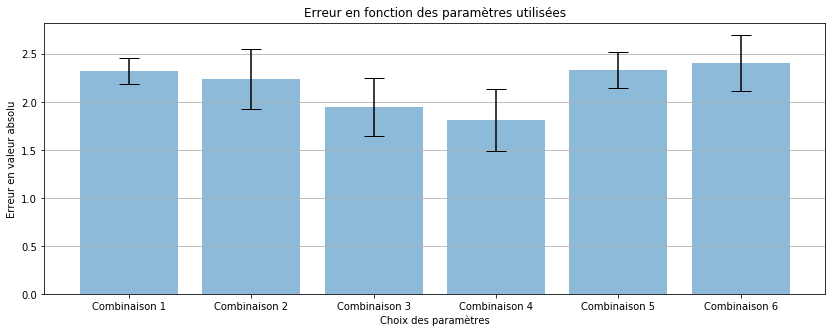


Figure 11 : Graphique de l'erreur en fonction des paramètres utilisés pour la régression linéaire multiple. Combinaison 1 : l'arrosage et l'humidité actuel. Combinaison 2 : l'arrosage, l'humidité et la température actuel. Combinaison 3 : l'arrosage et l’humidité actuel et le temps depuis le dernier arrosage. Combinaison 4 : l'arrosage, l'humidité et la température actuel et le temps depuis le dernier arrosage. Combinaison 5 : l'arrosage et l’humidité actuel et la valeur du dernier réel arrosage effectué. Combinaison 6 : l'arrosage et humidité actuel et combien de journée se sont écoulé depuis le début de l’expérience.

## Choix de l’algorithme d’apprentissage

Afin de trouver le meilleur algorithme pour prédire l’humidité sur une période donnée et ainsi créer un programme d’arrosage précis, j’ai effectué un comparatif entre plusieurs algorithmes. Le but étant de faire la meilleure prédiction d’humidité sur une journée entière. La prédiction doit être aussi proche que possible à n’importe quel moment de la journée. Rappel, une journée commence à l’arrosage et se termine à l’arrosage suivant. Une seule régression était faite sur toute la journée avec les paramètres sélectionner ci-dessous.

Pour mener à bien des tests, il faut séparer les données en deux, une partie de données d’apprentissage (données de training) et une autre partie de données de test. Il ne faut pas utiliser les mêmes pour apprendre que pour tester. De plus cette séparation doit être totalement aléatoire. Les données utilisées pour le choix de l’algorithme sont celle fourni par le bac de Demeter. Après élimination des journées erronées ou incomplètes, le nombre de journées restantes est de 91. 77 journées ont été utilisé pour apprendre et 14 pour tester, ce qui fait un ratio d’environ 84% de données d’apprentissage et 16% de données de test. Le calcul de l’erreur se fait de la manière suivante. Il s’agit de la moyenne de la différence en valeur absolu entre la prédiction et la valeur réelle pour chaque itération.

Grâce à mes recherches j’ai choisi de tester les algorithmes de tester les algorithmes d’apprentissage suivants : régression linéaire multiple moindre carré, régression de Lasso, Elastic Net et boost. Pour la régression de Lasse et élasticnet, il est nécessaire de fournir un paramètre d’apprentissage. Il s’agit du taux de régularisation. Ainsi il est nécessaire de trouver ce meilleur taux. Il m’a fallu varier ce taux et en garder le meilleur.

Graphique

Selon mes résultats j’ai pu voir que la régression linéaire multiple moindre carré obtient le meilleur rapport faciliter d’utiliser et précision de l’algorithme.

## Amélioration de l’algorithme

Comme mentionné dans la partie *Récolte de données* (page.10), une journée peut être séparé en 3 parties distinctes. La première, très courte (30 minutes), correspond à l’arrosage. La deuxième à la période d’évaporation ou infiltration de l’eau arrosé. Enfin la troisième correspond à la partie de stabilisation de l’humidité avec une légère perte d’humidité. Cette dernière perdure jusqu’au prochain arrosage. Afin d’améliorer la précision, chaque journée est séparée suivant ces trois parties et une régression linéaire est appliqué sur chacune de ces trois parties distinctement. Ceci permet grandement d’améliorer la précision de prédiction. La première partie est uniquement les 30 minutes qui suivent l’arrosage. La deuxième et troisième partie se partage le reste de la journée. La séparation a été défini de sorte qu’elle minimise l’erreur.

Une question se pose. Est-ce que cette limite dépend de la quantité de précipitation ? On peut penser que plus la précipitation est conséquente, plus la période d’évaporation sera longue. Ainsi pour chaque quantité s’arrosage fourni, j’ai cherché la meilleure séparation. Voici des graphiques représentant l’erreur en fonction de la variation pour chacun des arrosages effectués dans l’ordre chronologique (10, 20, 40, 35, 45, 15). L’apprentissage et les tests ont été effectués uniquement sur les journées arrosées par la quantité correspondante.

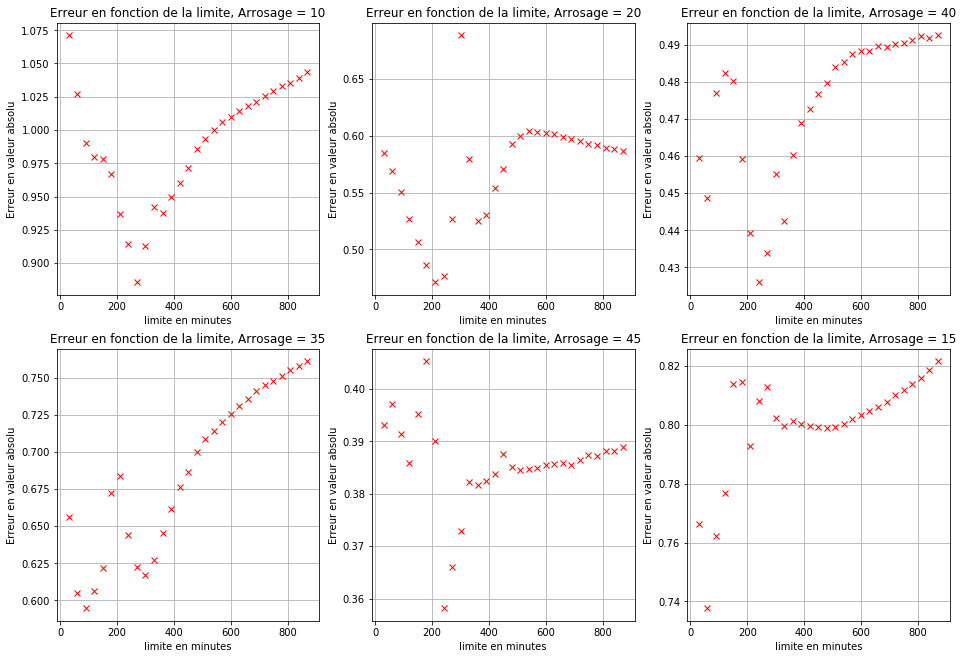


Figure 12 : Erreur en fonction de l'arrosage et de la limite utilisé entre la partie d'évaporation et la partie de stabilisation. Les données ont été séparés suivant la quantité d’arrosage. Par exemple, le graphe en haut à gauche n’utilise que les données avec un arrosage de 10 secondes. On peut voir que la limite ne dépend pas de la quantié d'arrosage. Plus d’arrosage ne signifie pas que la limite optimale après une plus longue période.

On peut voir que la quantité de précipitation n’a qu’un très faible impact sur cette limite. Elle se situe en moyenne vers de 240 minutes après l’arrosage. Pour un arrosage de 35 et 45, deux minimums locaux existent. Il s’agit environ de 90 minutes et 240 minutes après l’arrosage. Seul le minimum de 240 minutes ait conservé car il s’agit du même que pour les autres arrosages. Cette même limite optimale est obtenue si on mélange tous les arrosages utilisés. Ainsi la séparation à 240 minutes entre la partie d’évaporation et stabilisation sera la même pour tous les arrosages et sera utilisé pour la création de mon algorithme de prédiction car elle permet une bien précision.

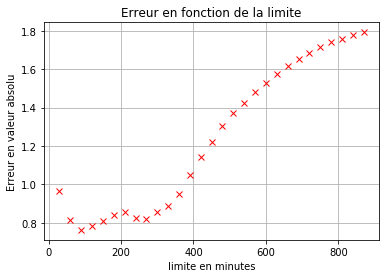


Figure 13: Erreur en fonction de la limite utilisée pour toutes les valeurs d'arrosage rassemblées.

## Analyse des résultats

Le but du projet est de fournir un logiciel permettant de définir un plan d’arrosage sur plusieurs jours pour atteindre un certain niveau d’humidité. La création du modèle de prédiction a été basé sur des données d’arrosage journalier. Un arrosage était programmé chaque jour. Dans le domaine du pratique, on ne veut pas toujours arroser chaque jour ou si on désire fortement augmenter l’humidité en utilisant le moins d’eau disponible, on peut arroser plusieurs fois par jour. C’est pourquoi, mon logiciel, qui est présenté dans le chapitre suivant « *Elaboration d’un programme d’arrosage* », a la fonctionnalité d’arroser une, deux ou aucun fois par jour. Ce choix de se limiter à un arrosage maximum a été fait car il faut au moins 6 heures pour que l’humidité se stabilise correctement après un arrosage. De plus cette limitation a deux arrosages maximums par jours, permet de l’utilisation d’eau. S’il faudrait arroser plus que deux fois par jour pour atteindre une certaine quantité d’humidité, on peut considérer que la demande est disproportionnée ou que l’environnement de la culture n’est pas adapté à la culture de plantes.

Ainsi notre prédiction doit être correct après 12h, 24h et plus. Selon les résultats affichés précédemment, deux algorithmes obtiennent de très bons résultats. Il s’agit de la régression linéaire multiple avec la méthode des moindres carrés (LSMLR) séparé en trois parties et de la régression gradient boosting. Ce dernier obtient une très légère meilleure précision. Cependant les tests ont toujours été effectués sur des journées comportant un unique arrosage. Aucune journée comportant aucun arrosage ou deux arrosages n’est disponible pour effectuer des tests. Mais si on observe l’évolution de l’humidité des données récoltés (voir graphique ci-dessous), la pente de l’humidité a tendance à être très faible en fin de journée. Ceci est généralement également le cas avec la méthode LSMLR. A contrario, avec la régression gradient boosting, la pente est plus élevée en fin de journée qu’avec LSMLR. Ainsi si on envisage qu’on arrosage qu’un jour sur deux, on peut imaginer et conclure que LSMLR aura une meilleure prédiction que gradient boosting. De plus, cette dernière méthode est souvent utilisée comme une boîte noire. On ne sait pas réellement ce qui se fait et les résultats dépendent fortement des données analyser. C’est pour ces deux raisons que j’ai décidé d’utiliser pour mon logiciel la régression linéaire multiple. Nos futures prédictions auront donc tendance à être inférieur ou supérieur d’1% d’humidité.

# Elaboration d’un programme d’arrosage

A présent, le meilleur algorithme de prédiction et les meilleurs facteurs ont été choisis. Les résultats obtenus permettent d’obtenir une précision très correcte. En moyenne l’erreur à une itération donnée est d’environ 1 %. L’implémentation d’un logiciel fournissant la quantité d’eau qui est nécessaire d’arroser est enfin possible. Ce logiciel demandera à l’utilisateur de fournir le pourcentage d’humidité actuelle, le nombre de jours sur lesquels se déroule la prédiction, l’humidité désiré à la fin de cette durée et la température prévue pendant cette période. Ensuite le logiciel fourni la quantité d’arrosage journalière qu’il faut pour obtenir l’humidité désirée.

En ce qui concerne la température, j’ai fait le choix de demander à l’utilisateur uniquement la température pendant la journée et la température pendant la nuit. Il s’agit d’une estimation. Une autre solution aurait été de prendre la météo fournie par le web mais cette dernière correspond à la météo en extérieur et n’est pas du tout comparable à la température en intérieur. De plus on demande qu’une simple estimation car le facteur température n’est pas des plus impactant et qu’une simple ouverture de fenêtre ou de store peut changer drastiquement la température.

Le logiciel fait le choix d’un arrosage homogène durant la période de simulation C’est-à-dire que l’arrosage aura de très faible variation entre les jours. L’arrosage de base est prévu pour s’effectuer à midi durant le fuseau horaire d’été ou à 11h durant l’heure d’hiver. S’il s’avère qu’un arrosage quotidien est insuffisant pour atteindre le taux d’humidité souhaité, le logiciel change d’un seul arrosage quotidien à un deux arrosages quotidiens. Dans ce cas, un arrosage se déclenchera à midi et un autre à minuit. S’il s’avère qu’il ne faut pas arroser, le système nous informe également. La quantité maximale d’un arrosage est de 50 secondes. Cette limite a été fixée car le système d’arrosage est limité, il y a des risques de fuite d’eau. De plus par expérience trop d’arrosage n’est pas efficace et la perte d’eau est trop grande. Le but premier de cette expérience étant d’économiser de l’eau. Si malgré tout, deux arrosages quotidiens sont insuffisants pour atteindre l’objectif d’humidité, le logiciel l’indique et fournit le plan d’arrosage qui s’y rapprocherai le plus avec deux arrosages journaliers.

Voici ci-dessous une image de l’utilisation du logiciel. Elle est précaire mais rempli entièrement les fonctionnalités souhaitées. Ce logiciel permet de définir la quantité d’arrosage uniquement pour le bac Déméter étant donné que seule ses données ont été utiliser pour la construction du modèle.

Image

# Conclusion

Le logiciel créé remplit, avec une précision relativement correcte, les fonctionnalités escomptées. C’est-à-dire il nous renvoie un planning d’arrosage pour atteindre un certain niveau d’humidité du sol. Les résultats sont précis à +/- 1% d’humidité en moyenne. Ainsi il est conseillé de ne pas demander un planning sur une trop longue période. Par exemple, en moyenne, un planning sur 4 jours aura une différence de 4% d’humidité entre la prédiction effectuée et les résultats observés en partant que les paramètres de température insérés par l’utilisateur sont proches de la réalité. Etant donnée que la plage d’eau disponible dans le sol utilisé est d’environ 15%, ce pourcentage s’avère être dans une marge d’erreur relativement correcte.

Le logiciel fonctionne uniquement pour le bac Demeter et si on place un capteur d’humidité à 2 centimètres d’un arrosage. Seules les données observées dans le bac Déméter sont utilisées pour l’apprentissage de l’algorithme. Ce choix a dû être fait car des problèmes de récolte de données sont survenu dans le bac Cérès. Le programme créé est réutilisable pour d’autres bac sous réserve que les capteurs sont placés à une même distance d’un point d’arrosage et que les capteurs et l’arrosage soient étalonnés.

En ce qui concerne les points à améliorer, j’envisagerai les changements suivants :

* Avoir plus de données d’apprentissage. La période de collecte de données aurait pu être plus longue. De plus il aurait fallu étalonner les capteurs d’humidité entre eux. En effet ceci aurait pu permettre d’utiliser les données de bacs différents. Un plus grand nombre de données d’apprentissage permet d’améliorer la précision.
* Avoir des données d’apprentissage sur une fréquence d’arrosage variables. En effet ceci aurait pu permettre de voir l’impacte de deux arrosages par jour sur l’humidité et répondre à la question « Est-ce que l’humidité réagit différemment à la fréquence d’arrosage ? »
* La question du bien-être de la plante n’a pas été approfondie. Une étude de l’impacte de l’humidité sur la croissance d’une espèce de plante aurait été intéressante. Il y aurait fallu pour ceci avoir plusieurs plants avec des arrosages programmables indépendamment à température et terre identique. Ceci aurait par exemple permis de répondre à la question « Est-ce que les épinards préfèrent une humidité de 15%, 20% ou 25% ? »

# Bibliographie

* <https://eric.univ-lyon2.fr/~ricco/cours/slides/regularized_regression.pdf>
* https://scikit-learn.org/stable/modules/linear\_model.html
* <https://machinelearningmastery.com/autoregression-models-time-series-forecasting-python/>
* <https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab>
* <https://ijoear.com/Paper-October-2016/IJOEAR-SEP-2016-34.pdf>
* <https://www.semanticscholar.org/paper/Applying-machine-learning-on-sensor-data-for-the-Goldstein-Fink/46a0541b6c3ed4563a31ad395620531389548baa>
* <https://www.sciencedirect.com/science/article/pii/S0309170809001705>
* [www.omafra.gov.on.ca/french/engineer/facts/11-038.htm#3](http://www.omafra.gov.on.ca/french/engineer/facts/11-038.htm#3)

1. Un processus est statique ou stationnaire quand ses propriétés qui le caractérise sont indépendante du temps. [↑](#footnote-ref-1)
2. L’erreur utilisé sur tous mes calculs est la suivante. Elle correspond à la moyenne de la différence en valeur absolu entre la prédiction et la valeur réelle pour chaque itération (toutes les 30 minutes) [↑](#footnote-ref-2)
3. L’arrosage se mesure en secondes. Cette quantité représente le temps de pompage de l’eau qui sera ensuite déverser aux plantes par l’intermédiaire d’un arrosage goûte à goûte. [↑](#footnote-ref-3)