

Plateforme numérique de guides cultureux dans un contexte de changement climatique et de baisse des rendements agricoles

P. François Mbissane Faye^{1,*}, Baye Mor Sall¹, Bineta Dembélé¹, J. Ana Awa Faye²

¹Université du Sine Saloum El-Hâdj Ibrahima NIASS, Département Mathématiques, Informatique, Sénégal

²Université Aix-Marseille, Institut de Management Public et Gouvernance Territoriale, France

Reçu: 15 Février 2023 / Reçu sous sa forme révisée: 08 Juin 2023 / Accepté: 28 Juin 2023

Résumé:

Dans l'agriculture, l'estimation du rendement des cultures est essentielle pour améliorer la productivité et les processus décisionnels tels que les prévisions des marchés financiers et la résolution des problèmes de sécurité alimentaire. L'objectif principal de cet article est de disposer d'outils pour prédire et améliorer la précision des prévisions de rendements des cultures à l'aide d'algorithmes de Machine Learning (ML) comme CART¹, KNN² et SVM³. Nous avons développé une application mobile et une application web qui utilisent ces algorithmes pour une utilisation pratique par les agriculteurs. Les tests montrent que notre système (architecture de collecte et de déploiement, application web et application mobile) est opérationnel et permet de valider des connaissances empiriques sur les paramètres agro climatiques en plus de l'aide à la prise de décision proactive. Les résultats expérimentaux obtenus sur les données agricoles et les performances des algorithmes de ML sont comparés à l'aide d'une validation croisée afin d'identifier les plus efficaces suivant les données agricoles. Les applications mises en œuvre démontrent que l'approche proposée est efficace pour prédire le rendement des cultures et donne des réponses rapides et précises aux agriculteurs pour l'aide à la prise de décision.

Mots-clés : Prédiction ; Apprentissage automatique ; Intelligence Artificielle ; Agriculture numérique.

Abstract:

Crop yield prediction is a key to improving productivity and decision-making processes such as financial market forecasting and addressing food security issues. The main objective of this paper is to have tools to predict and improve the accuracy of crop yield forecasts using machine learning (ML) algorithms such as CART¹, KNN² and SVM³. We developed a mobile app and a web app based on these algorithms for practical use by farmers. The tests show that our system (collection and deployment architecture, web application and mobile application) is operational and validates empirical knowledge on agro-climatic parameters in addition to proactive decision-making support. The experimental results obtained on the agricultural data

*Auteur correspondant:

Adresse e-mail: pascal.faye@ussein.edu.sn (P.F.M. Faye)

¹Classification And Regression Trees; ²K-nearest neighbours; ³Support Vector Machine.

and the performance of the ML algorithms are compared using cross-validation in order to identify the most effective ones according to the agricultural data. The applications implemented demonstrate that the proposed approach is effective in predicting crop yields and provides timely and accurate responses to farmers for decision support.

Keywords: Prediction; Machine Learning; IOT; Digital Agriculture; Smart Agriculture.

1. Introduction

L'augmentation considérable de la population, la détérioration des terres et les changements climatiques posent de grands défis au secteur agricole en termes de productivité et de durabilité. Bien que les agriculteurs soient qualifiés, il existe un écart énorme entre les connaissances scientifiques et technologiques et leur disponibilité dans les zones rurales. L'un des principaux défis pour la sécurité alimentaire d'un pays est le changement climatique et ses effets sous la forme d'événements météorologiques extrêmes. L'augmentation de la température de 1 à 2,5 degrés Celsius prévue pour 2030 est susceptible d'avoir de graves effets sur les rendements des cultures [1], car elle permet des changements dans la photosynthèse, augmente le taux de respiration des plantes et affecte les populations de ravageurs.

L'un des objectifs proposés d'ici 2030 est « pas de faim » et l'autre objectif est « la promotion de l'agriculture durable » [2]. L'agriculture durable contribue à autonomiser les petits agriculteurs, à mettre fin à la pauvreté et à améliorer la croissance financière des pays. Pour assurer un accès durable à des aliments nutritifs de manière

universelle, les pays imposent une production alimentaire et des pratiques agricoles continues [3]. L'observance agricole opportune et économique est essentielle pour atteindre ces objectifs. Dans ce contexte, l'estimation du rendement des cultures est cruciale pour vérifier et mettre en place des processus cognitifs plus élevés tels que l'assurance-récolte, la prédiction du marché monétaire et la résolution des problèmes de sécurité alimentaire. Avec les évolutions de la technologie [4 - 6], l'objectif de la présente étude est d'utiliser les algorithmes d'apprentissage automatique de l'intelligence artificielle [7] et les systèmes de contrôle pour conseiller, modifier et améliorer la productivité agricole.

Des algorithmes d'apprentissage automatique (ML) tels que la régression linéaire et la régression linéaire multiple, ont été utilisés pour faire des prédictions de rendement [8 - 11]. Notre objectif est de développer une application Web et une application mobile qui intègrent ces algorithmes basés sur CART, KNN et SVM, afin de fournir aux agriculteurs (utilisateurs) une approximation du rendement des cultures qui sera produite en fonction de l'intrant donné et de trouver également la relation entre le

rendement (variable dépendante) et d'autres variables indépendantes.

Pour ce faire, nous commençons par un état de l'art pour ensuite faire l'étude détaillée des notions de bases pour la compréhension de nos travaux. Après ces parties, nous allons décrire de manière succincte les types d'algorithmes ML utilisés avant de faire la description de notre système, la conception et la réalisation de nos applications web et mobile. Enfin, nous allons terminer par la conclusion et les recommandations pour les travaux futurs.

2. Etat de l'art

Un cadre de réseau neuronal convolutif - réseau neuronal récurrent (CNN-RNN) pour la prédiction du rendement des cultures a été introduit par Khaki *et al.* [12]. Dans leurs travaux, d'autres modèles tels que le Random Forest (RF), les réseaux profonds entièrement neuronaux (DFNN) et les algorithmes LASSO ont été comparés à CNN-RNN pour prédire le rendement en maïs et en soja. Les prédictions ont été effectuées dans toute la Corn Belt aux États-Unis pour les années 2016, 2017 et 2018. Les résultats étaient basés sur trois catégories, ayant le sol, les conditions météorologiques et la gestion comme attributs, et la précision pour le maïs et le soja était respectivement de 87,82% et 87,09%.

Pour prédire le rendement des cultures, Ramesh et Vardhan [3] d'une part, et Almahdi [13] d'autre part, ont utilisé un classificateur RandomForest. Dans leurs travaux, une interface graphique basée

sur le Web a été conçue pour qu'un agriculteur connaisse le rendement des cultures à l'avance. L'ensemble de données contient des détails sur la production végétale du Maharashtra où l'étude a été menée. Un modèle de réseau neuronal artificiel de rétropropagation a été proposé par Meena et Singh [14] pour prévoir le rendement des cultures. Contrairement aux modèles flous, des facteurs physiques pour les prévisions de rendement ont été utilisés. Les rapports annuels d'évaluation des prévisions (AFER) sont comparés et ont été réduits de 11,40% à 3,82%.

Une analyse empirique pour la prévision du rendement des cultures a été effectuée dans [15 - 16] dans le but de mettre en évidence la prévision du rendement du « bajra » ou de la culture du mil perlé en mettant en œuvre des modèles statistiques appropriés tels que des modèles de régression et de séries chronologiques. Des modèles tels que la moyenne mobile intégrée auto-régressive (ARIMA) et un modèle ARIMA avec une variable exogène (ARIMAX) ont également été utilisés pour la prédiction. Le modèle ARIMAX a produit le meilleur résultat pour 'bajra' par rapport au modèle de série chronologique de régression.

Une prévision du rendement des cultures à l'aide de l'apprentissage automatique a été proposée par Prema *et al.* [17]. Ils ont utilisé une régression empilée pour la production du rendement des cultures, basée sur un facteur supplémentaire d'éléments nutritifs du sol. Les algorithmes de réseau neuronal efficace (ENeT), LASSO et kernel ridge présentaient des erreurs minimales

respectivement de 4%, 2% et 1%. Une page Web a été utilisée comme interface pour afficher le résultat prévu. Des applications mobiles telles que *uzhavan* [18] fournissent à l'agriculteur des installations leur permettant de connaître les informations sur les composantes du programme, les modèles de subvention, les semences et les engrais. Au regard de tout ce qui précède, l'on peut noter que l'intégration d'un algorithme ML dans une application Web et mobile est manquante.

La sélection de données appropriées est une partie très importante de tout algorithme d'apprentissage automatique. Les ensembles de données du référentiel de science des données ont été utilisés pour prévoir les rendements des cultures [19]. De nombreux chercheurs, dont Salwe et Khedekar [10] et, Rakshith [20], ont utilisé cet ensemble de données et en ont tiré des informations utiles. Dans notre système, l'ensemble des données utilisées pour la conception des applications web ont été prises sur le site web de la FAO et de la NASA [19]. Différents outils et plateformes ont été utilisés tels que MIT App Inventor [21, 22] pour la création d'une application mobile et Python [23] pour une application web pour la prédiction et la visualisation des résultats. Les données se concentrent principalement sur les informations culturelles, leurs itinéraires, les intrants (N, P, K), le pH et le climat. Les paramètres climatiques comprennent la température, l'humidité et les précipitations. Le tableau 1 rassemble les types de données des différents paramètres.

Tableau 1

Type de données des différents paramètres.

Attributs	Types de données
pH	Float
Précipitation	Float
Phosphore	Float
Potassium	Float
Azote	Float
Température	Float
Humidité de l'air	Float
Culture	String

La partie suivante, donne les concepts de base pour la compréhension de la pertinence des algorithmes ML que nous avons utilisés.

3. Concepts de base

Dans le présent travail, l'ensemble de données recueilli a été exploré et des techniques de prétraitement des données telles que l'imputation des valeurs manquantes ont été utilisées. Et ce, avant d'appliquer les techniques d'apprentissage.

3.1. Apprentissage supervisé

Les algorithmes d'apprentissage supervisé sont les algorithmes ML les plus couramment utilisés. Ces algorithmes prennent l'échantillon de données pour produire la sortie (la connaissance), c'est-à-dire les étiquettes ou les réponses, associée à chaque échantillon de données. L'objectif principal est d'apprendre l'association entre les échantillons de données d'entrée et les sorties correspondantes après avoir effectué plusieurs instances d'entraînement [9, 24, 25].

Considérons la variable d'entrée x et la variable de sortie Y . L'objectif d'un algorithme d'apprentissage supervisé est de trouver une fonction f de mise en correspondance de la variable d'entrée (x) avec la variable de sortie (Y), c'est-à-dire une expression du type $Y=f(x)$. Ce qui permettra d'obtenir de nouvelles données d'entrée (x). Nous pouvons facilement prédire la variable de sortie (Y) pour ces nouvelles données d'entrée. La démarche peut être schématisée grâce à l'exemple présenté à la figure 1.

3.2. Apprentissage non supervisé

L'apprentissage non supervisé regroupe un type d'algorithme ML dans lequel les modèles sont formés à l'aide d'un ensemble de données non étiquetées et sont autorisés à agir sur ces données sans aucune supervision [9, 25, 26]. L'objectif est de trouver la structure sous-jacente d'un ensemble de données, de regrouper ces données en fonction de leurs similarités, fréquences, etc. La démarche peut être représentée comme ci-dessous (Figure 2).

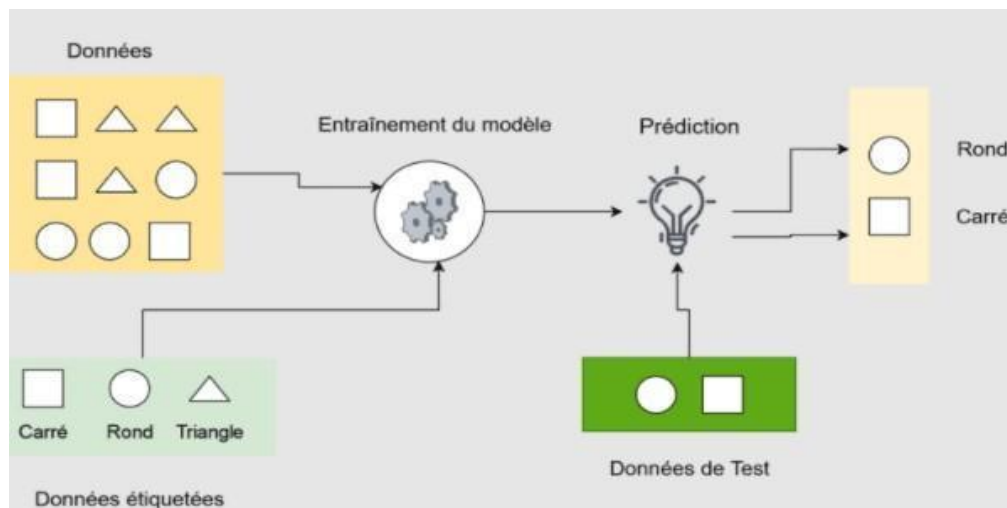


Fig. 1. Fonctionnement de l'apprentissage supervisé.

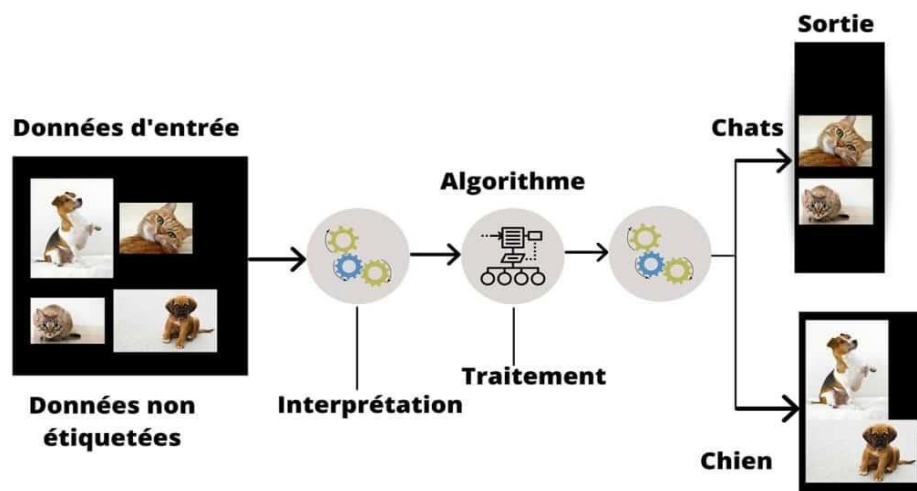


Fig. 2. Fonctionnement de l'apprentissage non supervisé.

Les données d'entrée sont non étiquetées, ce qui signifie qu'elles ne sont pas catégorisées et que les sorties correspondantes ne sont pas données non plus. Les données sont transmises au modèle d'apprentissage automatique afin de l'entraîner. Ces données brutes sont interprétées pour trouver les modèles cachés des données et appliquer ensuite des algorithmes appropriés tels que le clustering k-means. Une fois qu'il a appliqué l'algorithme approprié, l'algorithme divise les données en groupes en fonction des similarités et des différences entre les données.

3.3. Apprentissage par renforcement

Avec ce type d'apprentissage, un agent (algorithme) doit faire un choix parmi une liste d'actions. Ensuite, en fonction de l'action choisie, il recevra un retour de l'environnement (provenant d'un humain dans certaines situations ou d'un autre algorithme) : c'est soit une récompense pour un bon choix, soit une pénalité pour une mauvaise action. L'agent apprend quelle stratégie (ou choix d'actions) maximise le cumul de récompenses. Ce type d'apprentissage est souvent utilisé dans le cadre de la robotique, de la théorie des jeux et des véhicules autonomes [27 - 29].

4. Choix du type d'apprentissage

Sur la base des types de données collectées, nous n'avons utilisé que des

algorithmes d'apprentissage supervisé ou non-supervisé (CART, KNN ou SVM [31]) basés sur:

- la régression linéaire;
- les arbres de décision;
- et les méthodes de classification.

En effet, toutes nos données sont étiquetées ou non et les sources bien connues.

4.1. La régression linéaire

La régression linéaire est un type d'analyse prédictive de base. Le concept général de la régression est d'étudier deux questions :

- Un ensemble de variables prédictives permet-il de prédire une variable de résultat ?
- Quelles sont les variables les plus significatives et qui ont le plus d'impact sur le résultat ?

On utilise ces estimations de régression pour expliquer les relations entre variable dépendante et une ou plusieurs variables indépendantes. La forme la plus simple de l'équation de régression avec une variable dépendante et une variable indépendante est définie par la formule $y = c + b \cdot x$ (y : variable dépendante estimée ; c : constante ; b : coefficient de régression ; x : variable indépendante). On parle ici de régression linéaire simple. Pour la régression linéaire multiple on écrira $y = c + b \cdot x_1 + \dots + b_n \cdot x_n$, avec x_1 jusqu'à x_n , les variables indépendantes et b jusqu'à b_n les coefficients de régression respectifs des variables. Le type d'environnement et le fonctionnement de l'apprentissage par renforcement sont présentés respectivement aux figures 3 et 4.



Fig. 3. Type d’environnement pour un apprentissage par renforcement.

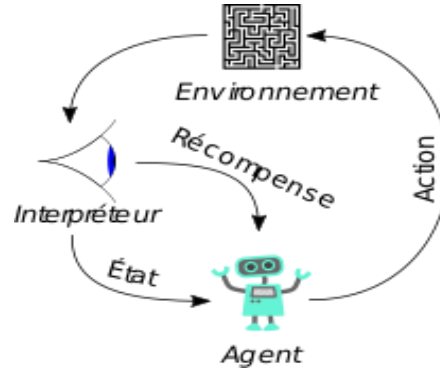


Fig. 4. Fonctionnement de l’apprentissage par renforcement.

4.2. Les arbres de classification et de régression

Les arbres de classification et de régression, aussi connus sous le nom de CART (Classification And Regression Tree), sont une forme simple d’arbres de décision. Cette structure n’utilise que des algorithmes et des structures de données. Ces arbres n’ont que deux composantes:

- Les nœuds de branchement, qui représentent une seule variable d’entrée et offrent un seul point de partage sur la variable.
- Les nœuds feuilles, qui représentent les deux variables de sortie.

A l’exécution de l’algorithme par la machine, la prédiction est faite en suivant les divisions du nœud en branche jusqu’à atteindre un nœud feuille. Ce nœud feuille est la prédiction (la sortie). L’algorithme des K plus proches voisins ou K-nearest neighbors (KNN) (Figure 5) est un algorithme ML qui appartient à la classe des algorithmes d’apprentissage supervisé facile à mettre en œuvre qui peut être utilisé pour résoudre les problèmes de classification et de régression.

En apprentissage supervisé, un algorithme reçoit un ensemble de données qui est étiqueté avec des valeurs de sorties correspondantes sur lequel il va pouvoir s’entraîner et définir un modèle de prédiction. Cet algorithme pourra par la suite être utilisé sur de nouvelles données afin de prédire leurs valeurs de sorties. Les principales étapes de l’algorithme des K plus proches voisins sont :

- Étape 1 : Sélectionner le nombre K de voisins.
- Étape 2 : Calculer les distances Euclidienne ou de Manhattan.

$$\sum_{i=1}^n |X_i - Y_i| \quad \text{(Euclidienne) (1)}$$

$$\sqrt{\sum_{i=1}^n (X_i - Y_i)^2} \quad \text{(Manhattan) (2)}$$

Pour un point non classifié aux autres points.

- Étape 3 : Prendre les K voisins les plus proches selon la distance calculée.
- Étape 4 : Compter parmi ces K voisins, le nombre de points appartenant à chaque catégorie.
- Étape 5 : Attribuer le nouveau point à la catégorie le plus présent parmi ces K voisins.
- Étape 6 : Le modèle sera prêt lorsque tous les points seront dans une catégorie.

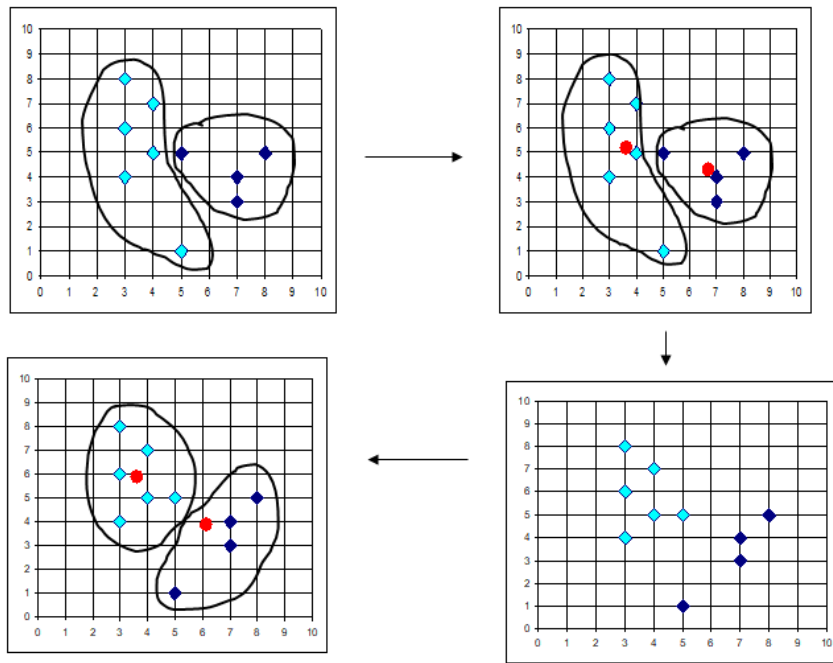


Fig. 5. Modèle K-NN.

Pour détecter des anomalies dans les données ou sur les prédictions successives nous utilisons les SVM qui permettent de résoudre des problèmes de classification, de régression ou de détection d'anomalie.

4.3. Les machines à vecteur de support (SVM)

La famille d'algorithmes SVM reste efficace dans les espaces de grandes dimensions et même si le nombre de dimensions est supérieur au nombre d'échantillons. Elle utilise un sous-ensemble de points d'entraînement dans la fonction de décision (vecteurs de support). Elle a pour but de séparer les données en classes à l'aide d'une frontière même pour des données qui ne

sont pas linéairement séparables de telle façon que la distance entre les différents groupes de données et la frontière qui les sépare soit maximale. Cela, à l'aide de fonctions mathématiques qui permettent de séparer les données en les projetant dans un *feature space* (un espace vectoriel de caractéristique de plus grande dimension). Cette distance est aussi appelée « *marge* ». Les algorithmes SVMs sont ainsi qualifiés de « *séparateurs à vaste marge* » et les « *vecteurs de support* » étant les données les plus proches de la frontière. Sur les figures 6 et 7 sont présentés respectivement la séparation avec SVM et l'utilisation de SVM pour la classification.

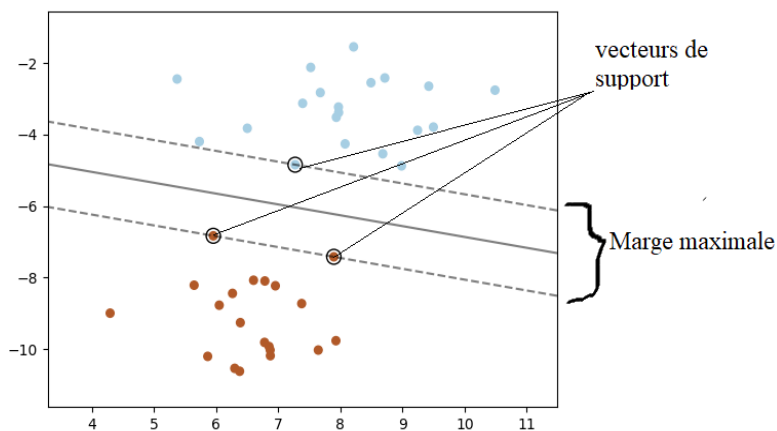


Fig. 6. Séparation avec SVM.

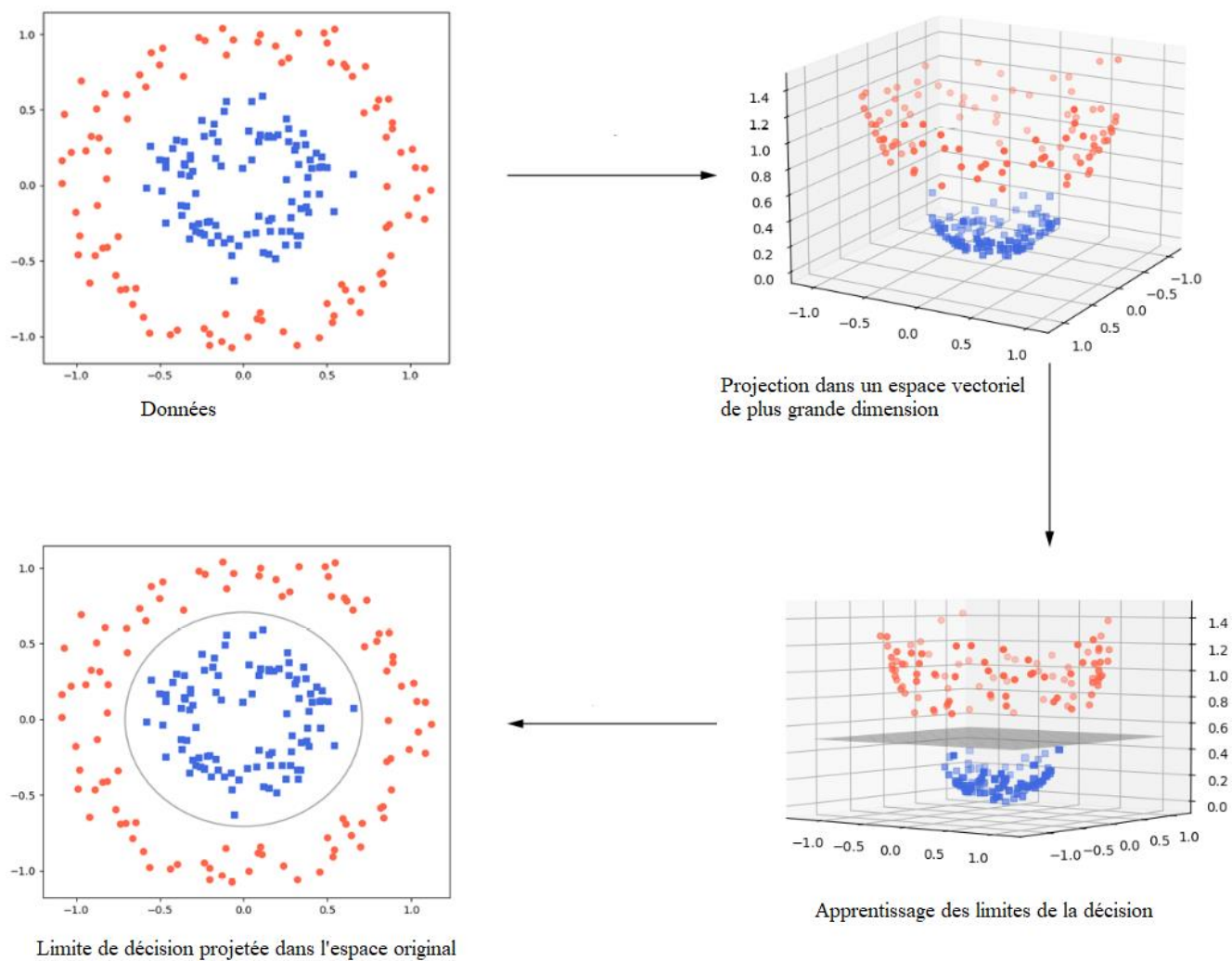


Fig. 7. Utilisation de SVM pour la classification.

Pour la réalisation de l'apprentissage supervisé sous python nous avons utilisé les packages python [23] tels que *Scikit-Learn* qui implémente une grande variété d'algorithmes d'apprentissage automatique, de prétraitement, de validation croisée et de visualisation à l'aide d'une interface unifiée ; *Streamlit* qui facilite la création et le partage d'applications Web personnalisées pour l'apprentissage automatique et la science des données ; *Pandas* pour l'analyse et la manipulation de données ; *NumPy* pour les calculs scientifiques car il fournit un objet tableau multidimensionnel, divers objets dérivés (tels que des tableaux masqués et des matrices) et un assortiment de routines pour des opérations rapides sur des tableaux. Le module *pickle* qui implémente des protocoles binaires de sérialisation et dé-sérialisation d'objets Python.

La sérialisation est le procédé par lequel une hiérarchie d'objets Python est convertie en flux d'octets. La dé-sérialisation est l'opération inverse, par laquelle un flux d'octets est converti en hiérarchie d'objets. *Matplotlib* est utilisé pour créer des visualisations statiques, animées et interactives de graphiques ; *Seaborn* pour la visualisation des données basée sur *matplotlib* . Il fournit une interface de haut niveau pour dessiner des graphiques statistiques attrayants et informatifs. *Pillow* est utilisé pour le traitement d'image et offre un support pour différents formats de fichiers tels que PPM, PNG, JPEG, GIF, TIFF et BMP.

5. Architecture du système

Un aperçu général du système utilisé est présenté à la figure 8.

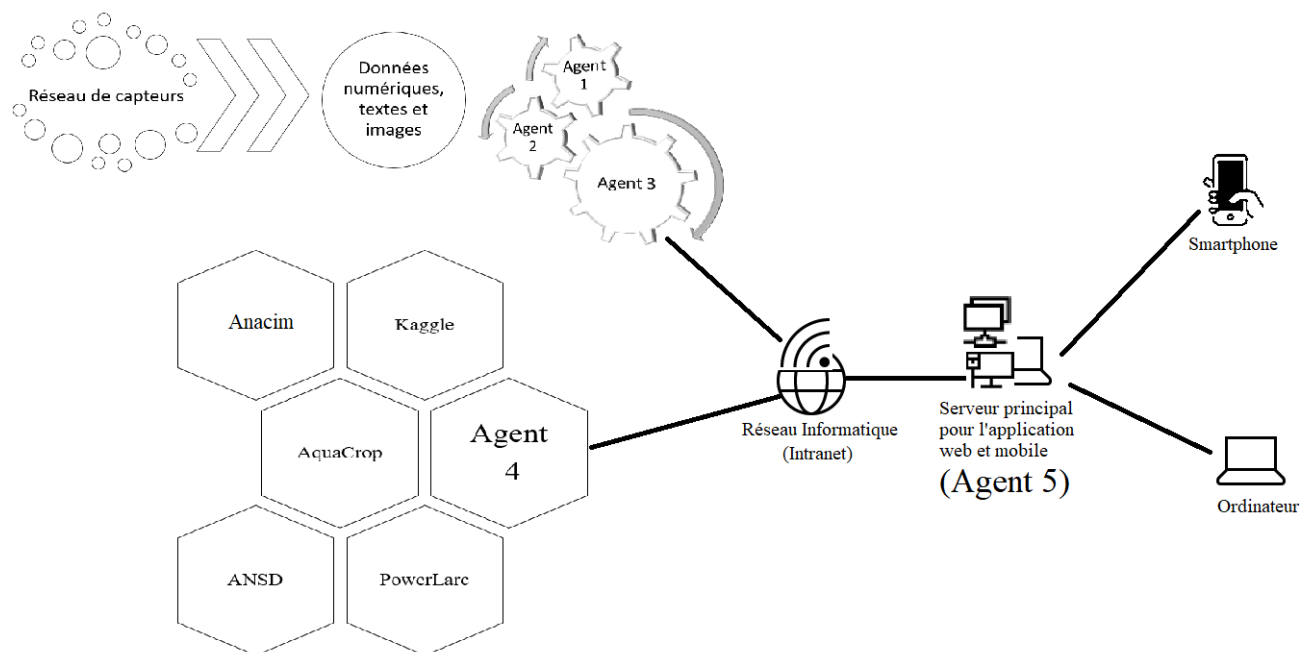


Fig. 8. Architecture de collecte de données et de déploiement des plateformes.

La description du système est faite ci-dessous.

- Le réseau de capteurs : pour la récupération des paramètres agroécologiques du champ d'application.

Réseau que nous avons déjà utilisé dans le cadre d'autres travaux comme dans [30].

- Agent 1 : Programme ML basé sur le CART pour les données étiquetées.

- Agent 2 : Programme ML basé sur le KNN pour les données étiquetées

- Agent 3 : Programme ML basé sur le SVM (SVC, NuSVC and LinearSVC) et qui utilise les données des agent 1 et agent 2.

- Agent 4 : Programme ML basé sur la régression linéaire, le CART et le KNN.

- Plateformes digitales : acquisition de données agricoles et météorologiques (Anacim⁴, ANSD⁵, Aquacrop⁶, Powerlarc⁷ et Kaggle⁸).

- Intranet : Réseau Ethernet sans fil pour la communication entre les différents agents qui tourne sur microcontrôleur ESP CAM (agent 1 et agent 2) et ordinateur (agent 3, agent 4 et agent 5)

- Agent 5 : Serveur d'application pour l'hébergement de la plateforme Web et pour la génération de la version mobile avec MIT App Inventor.

Le fonctionnement de l'architecture de déploiement et le principe de fonctionnement des plateformes sont présentés respectivement sur les figures 9 et 10.

Dans la suite, nous décrivons le fonctionnement de nos applications web et mobile.

6. Résultats et discussions

6.1. L'application mobile

APP INVENTOR [21, 22] est un outil de développement des applications en ligne pour les smartphones Android et permet à chacun de créer son application personnelle. La plateforme de développement est offerte à tous les utilisateurs possédant un compte Gmail. L'environnement de APP INVENTOR [21, 22] contient trois fenêtres qui sont proposées pendant le développement.

- Une pour la création de l'interface homme machine : elle donne l'interface de l'application (App Inventor Designer).
- Une pour la programmation par elle-même : elle permet, par l'assemblage des blocs, de créer le comportement de l'application (App Inventor Block Editor).
- Une pour l'émulateur : qui permettra de tester l'application. L'émulateur permet de remplacer un terminal réel pour vérifier le bon fonctionnement du programme.

La connexion d'un terminal Android (téléphone, tablette, etc.) permet ensuite d'y installer et d'utiliser l'application. De manière succincte, nous expliquons les principales fonctionnalités de l'application mobile de notre Guide Cultural. La page d'accueil de ce guide est présentée à la figure 11.

⁴<https://www.anacim.sn/spip.php?article67>; ⁵<https://www.ansd.sn/enquete>; ⁶<https://www.fao.org/aquacrop/software/fr/>; ⁷<https://power.larc.nasa.gov/data-access-viewer/>; ⁸<https://www.kaggle.com/datasets>.

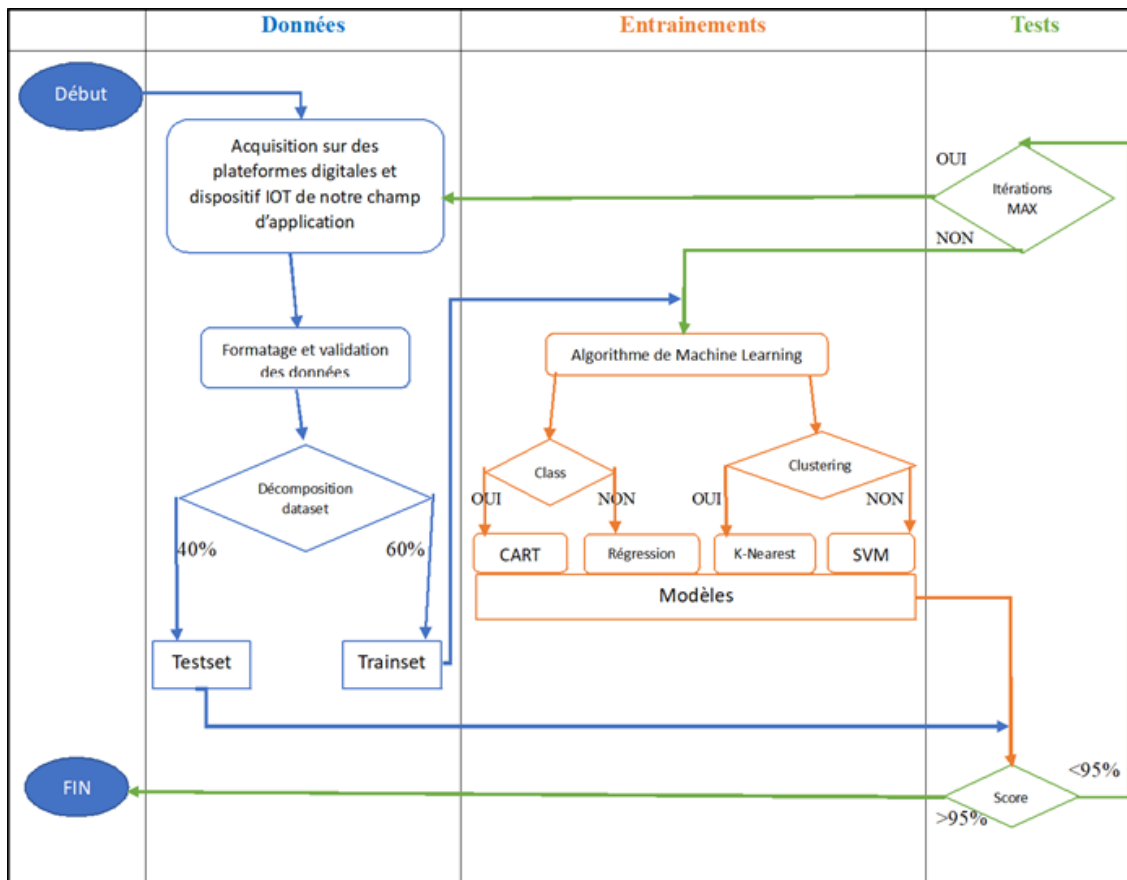
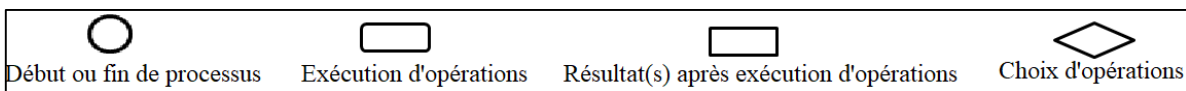


Fig. 9. Fonctionnement de notre architecture de déploiement.

Légende :



Les couleurs indiquent à quel moment est réalisé l'étape (acquisition et traitements de données, entraînements ou tests). Le texte à l'intérieur de

chaque forme donne plus de précisions sur la tâche ou le(s) résultat(s).

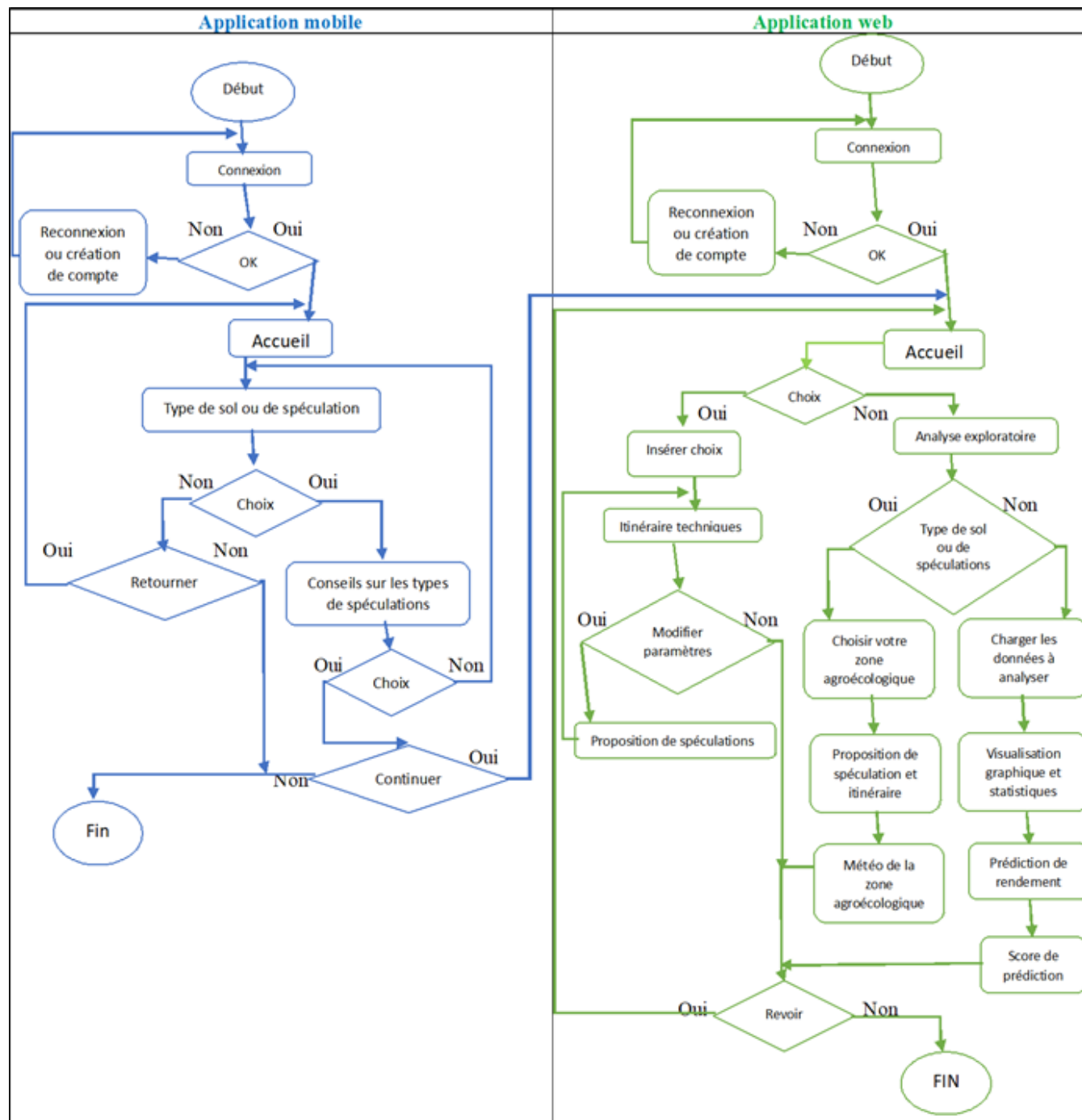
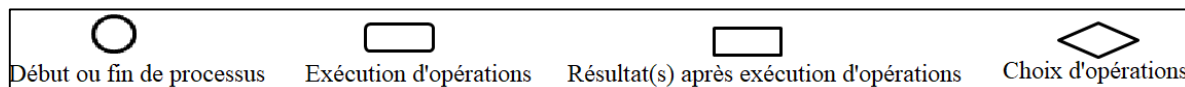


Fig. 10. Fonctionnement et interaction entre notre plateforme Web et mobile.

Légende :



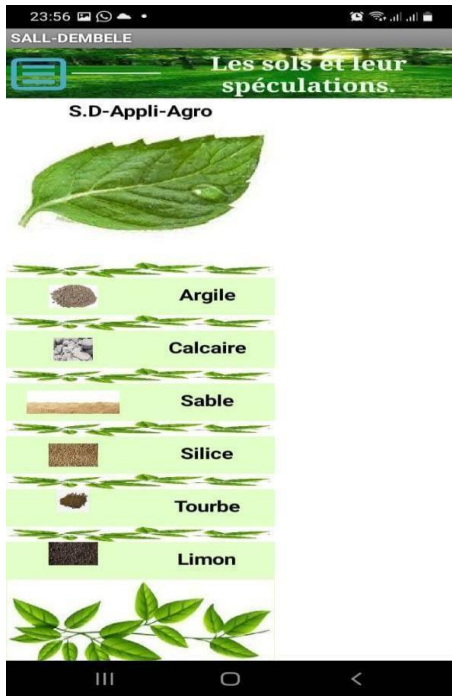


Fig. 11. Page d'accueil de Guide Cultural.

L'interface de la figure 11 montre les types de sol. En cliquant sur le bouton « argile », le résultat obtenu est indiqué à la figure 12.



Fig. 12. La page du bouton argile.

Une fois dans cette page, on peut choisir de se renseigner sur l'itinéraire d'une culture. Prenons l'exemple de la tomate en cliquant sur « tomate ». Cette page défile et nous montre l'intégralité de l'itinéraire technique de la tomate (Figure 13).



Fig.13. L'itinéraire de la tomate.

6.2. L'application web

Notre application web est conçue grâce au langage python [23]. Il sert d'outil de prédiction pour les producteurs agricoles. L'interface d'accueil est présentée à la figure 14.

La page d'accueil nous montre à gauche, l'ensemble des options selon les besoins récurrents des producteurs agricoles. Le choix du bouton « Graphique », conduit à la figure 15.

L'interface ainsi obtenue montre deux options, l'une pour choisir les types de graphes et l'autre pour choisir une caractéristique qui peut être un

intrant (pH, Azote, Phosphore, Potassium) ou bien un facteur climatique (Température, humidité, précipitations).

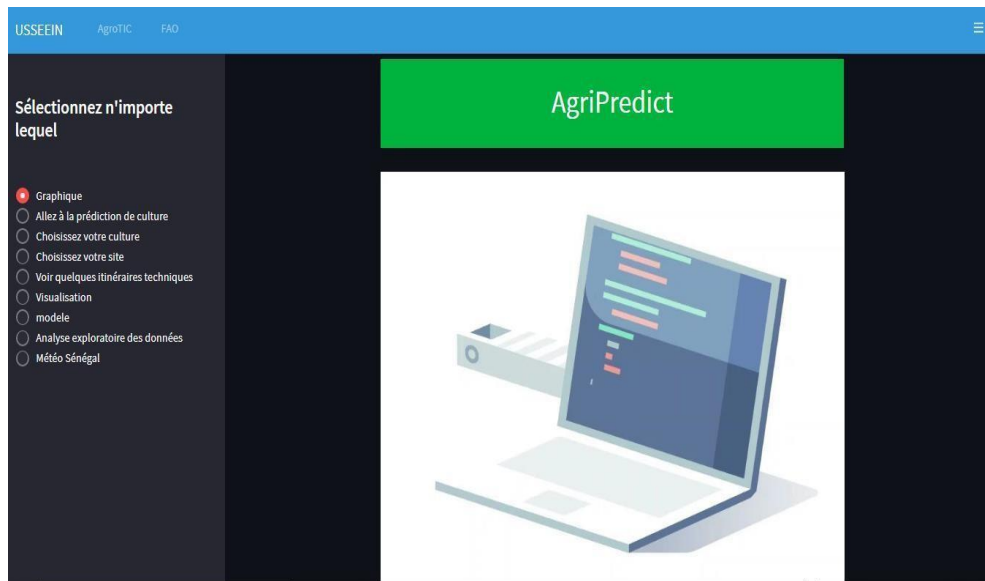


Fig. 14. Page d'accueil AgriPredict.

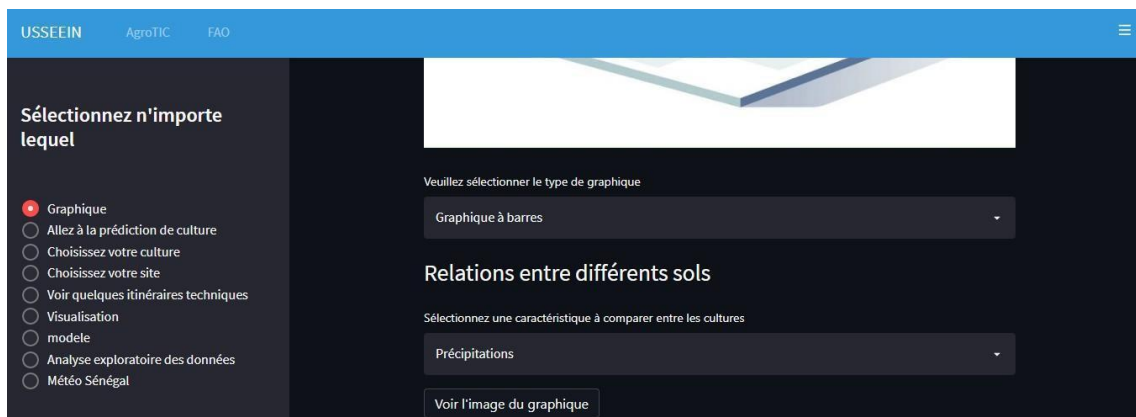


Fig. 15. Choix de graphe sur AgriPredict.

Pour notre exemple, nous avons choisi le diagramme de barre et la précipitation.

Les résultats confirment certaines connaissances empiriques sur les besoins en eau (Figure 16).

Le diagramme de la figure 16 montre le rendement de différentes cultures selon leur besoin en eau. Nous constatons que le riz a un fort besoin en eau par rapport aux autres cultures étudiées. Prenons le même exemple que précédemment en choisissant cette fois le diagramme en barre et la température.

Le résultat est présenté sur la figure 17.

Ce résultat montre que la papaye a le besoin le plus élevé en température comparé aux autres cultures. D'après les figures 16 et 17, nous pouvons dire que le rendement des cultures dépend de différents paramètres et que chaque culture a besoin d'un ensemble de caractéristiques environnementales pour son développement. La croissance, le développement de la culture et par conséquent son rendement peut être gravement compromis si les paramètres se situent en dehors de leurs seuils.

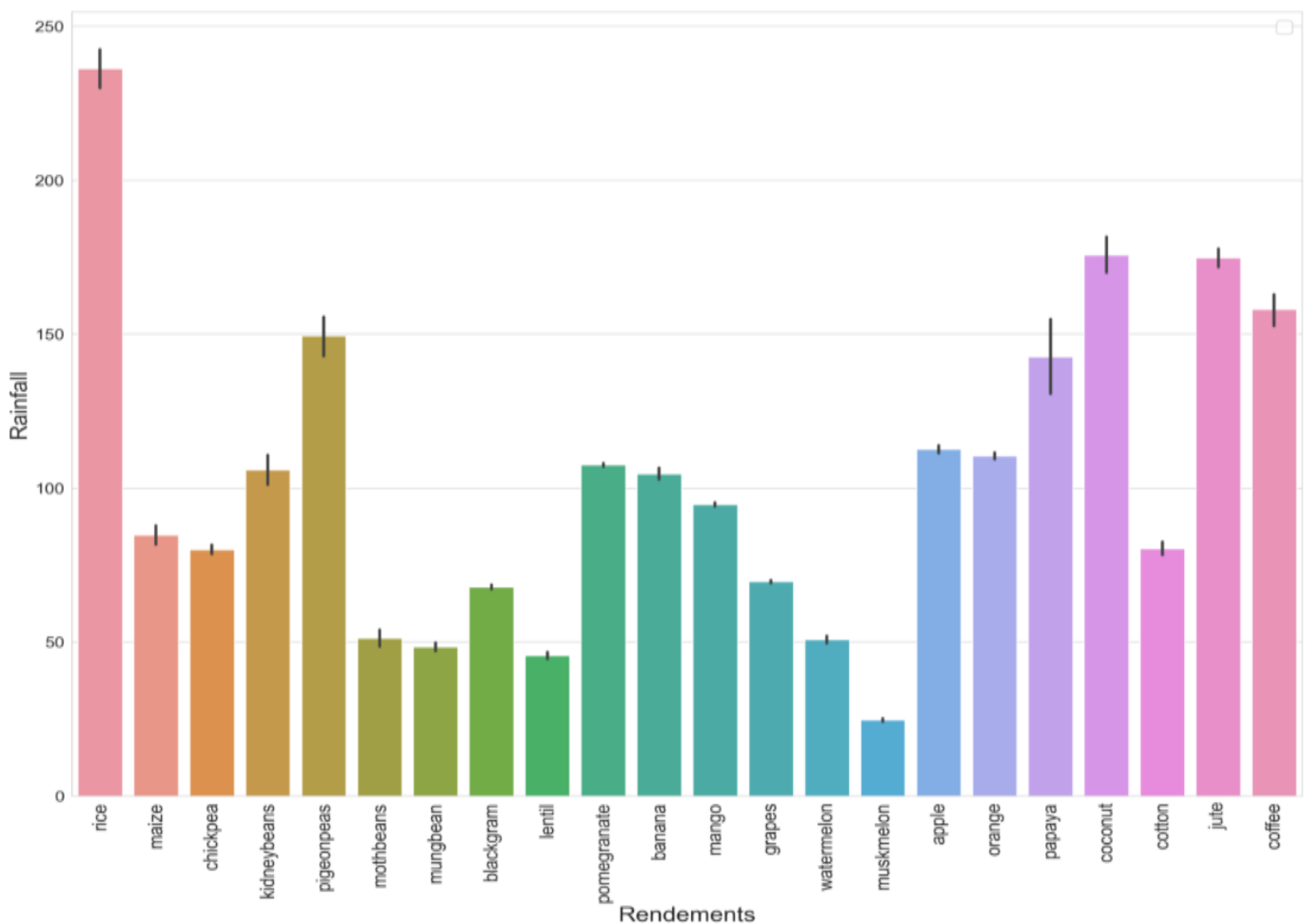


Fig. 16. Rendement des cultures par rapport à la pluviométrie.

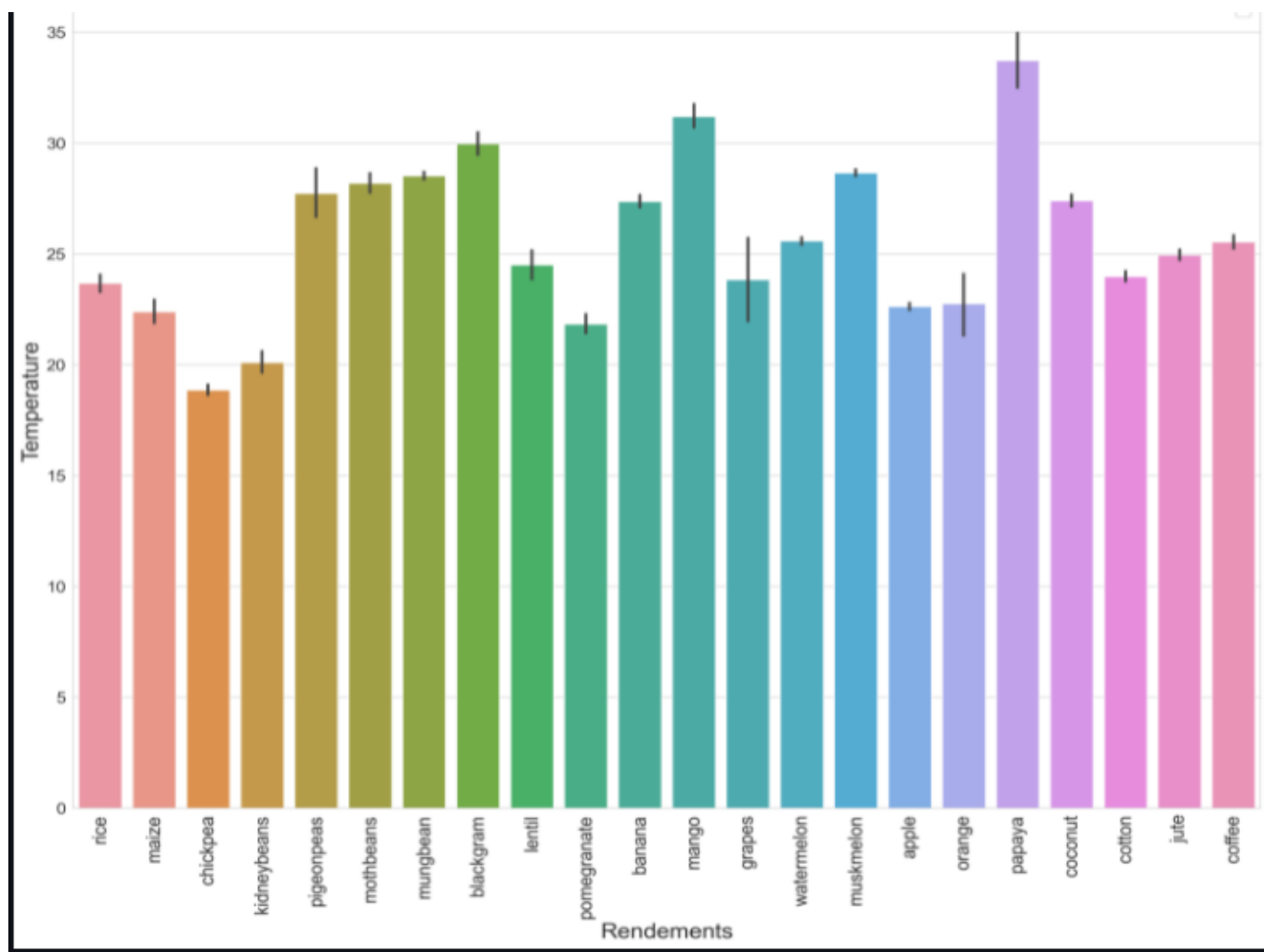


Fig. 17. Rendement des cultures selon la température.

Considérons les autres boutons de notre application. Cliquons sur le bouton « Allez à la prédiction de culture ».

La figure 18 montre que nous avons la possibilité de glisser le curseur pour définir les valeurs des paramètres. En cliquant sur « Connaitre la culture », on aboutit au résultat présenté à la figure 19.

Avec les paramètres que nous avons choisis dans la figure 18, la culture adaptée est le haricot pour avoir un meilleur rendement. Faisons le sens inverse en choisissant la culture grâce au bouton « Choisissez votre culture » (Figure 20).

Il apparaît une option permettant de choisir le type de culture et nous avons choisi le haricot. Le résultat obtenu est présenté à la figure 21. Cette figure montre les différents paramètres nécessaires pour aboutir à un meilleur rendement de la culture d'haricot. Cliquons sur le bouton « Choisissez votre site » et choisissons la région de Ziguinchor sur l'option « Veuillez sélectionner votre site » pour savoir la culture la mieux adaptée (Figure 22). Le résultat obtenu est présenté à la figure 23.



Fig. 18. Choix de la prédiction de culture.

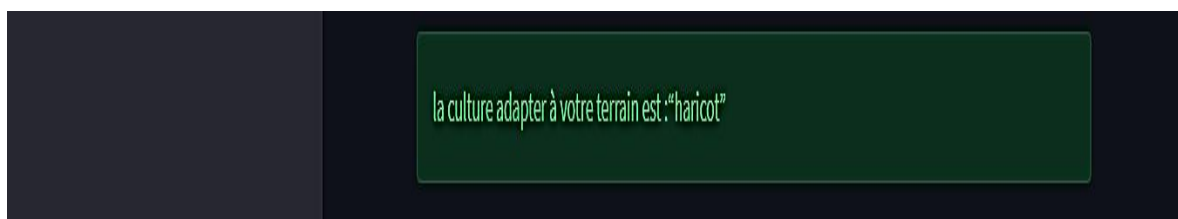


Fig. 19. Résultat après le choix des paramètres.

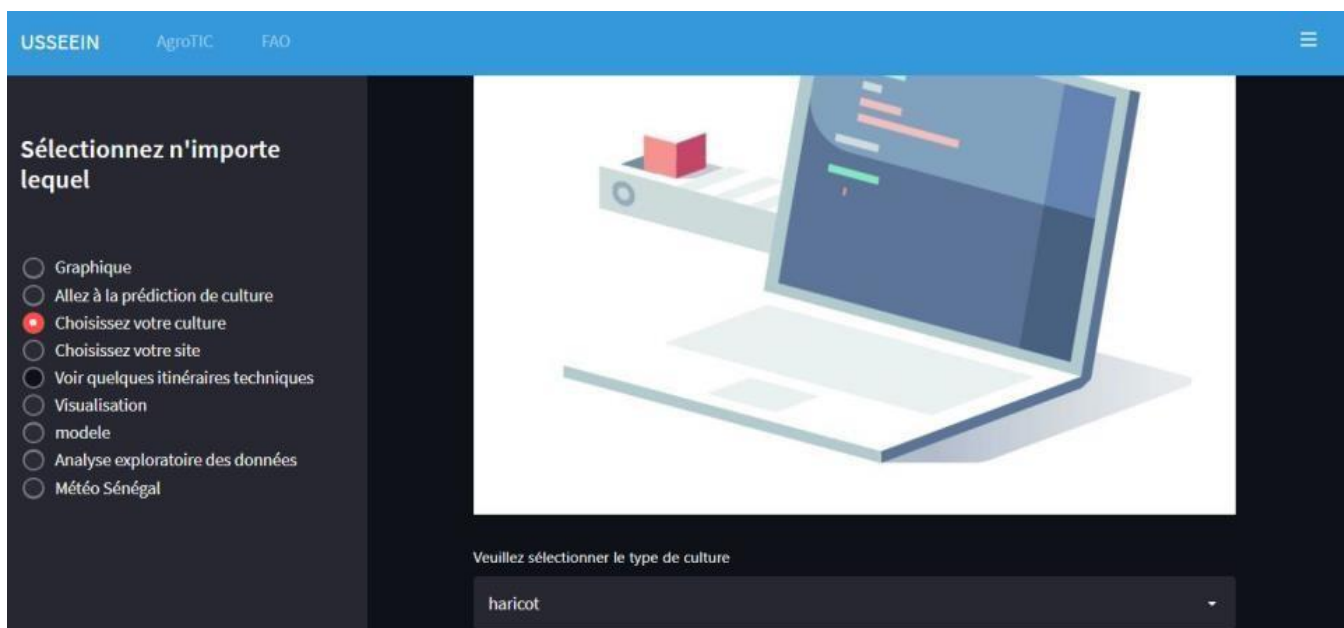


Fig. 20. Choix de la culture.

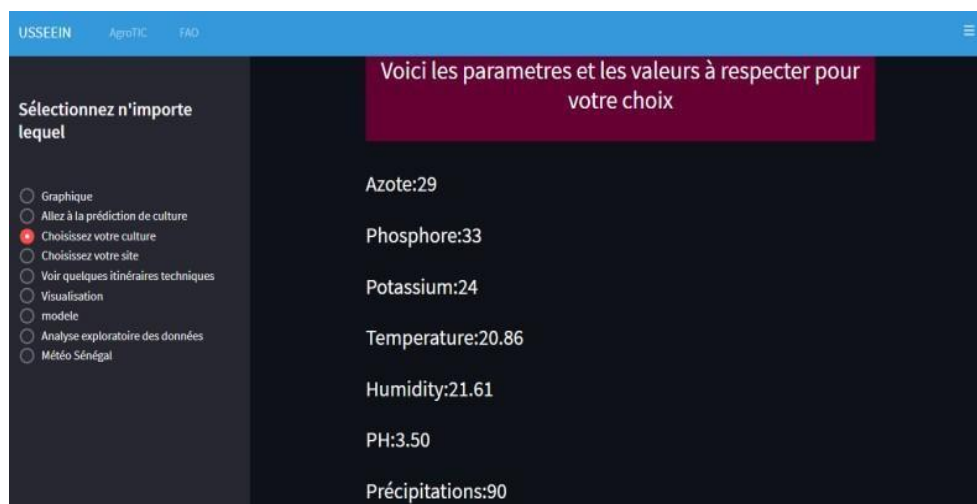


Fig. 21. Résultat après choix de la culture.

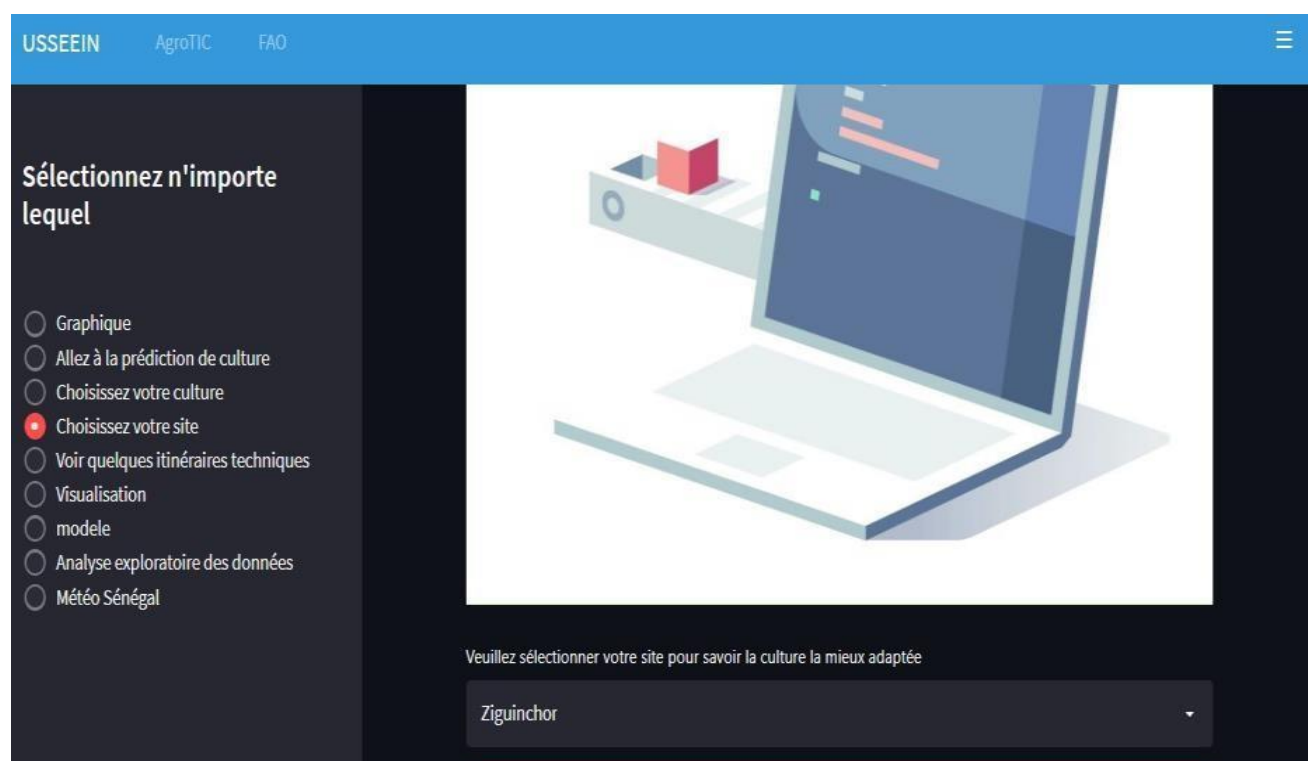


Fig. 22. Choix du site.

A la figure 23 sont présentées les différentes cultures qui sont adaptées dans la région de Ziguinchor pour avoir un meilleur rendement. Passons au bouton suivant qui permet de voir l'itinéraire des cultures en cliquant sur « Voir quelques itinéraires techniques » (Figure 24).

A partir de l'option « Veuillez sélectionner le type de culture », la pastèque a été choisie. Le résultat affiché est présenté à la figure 25.

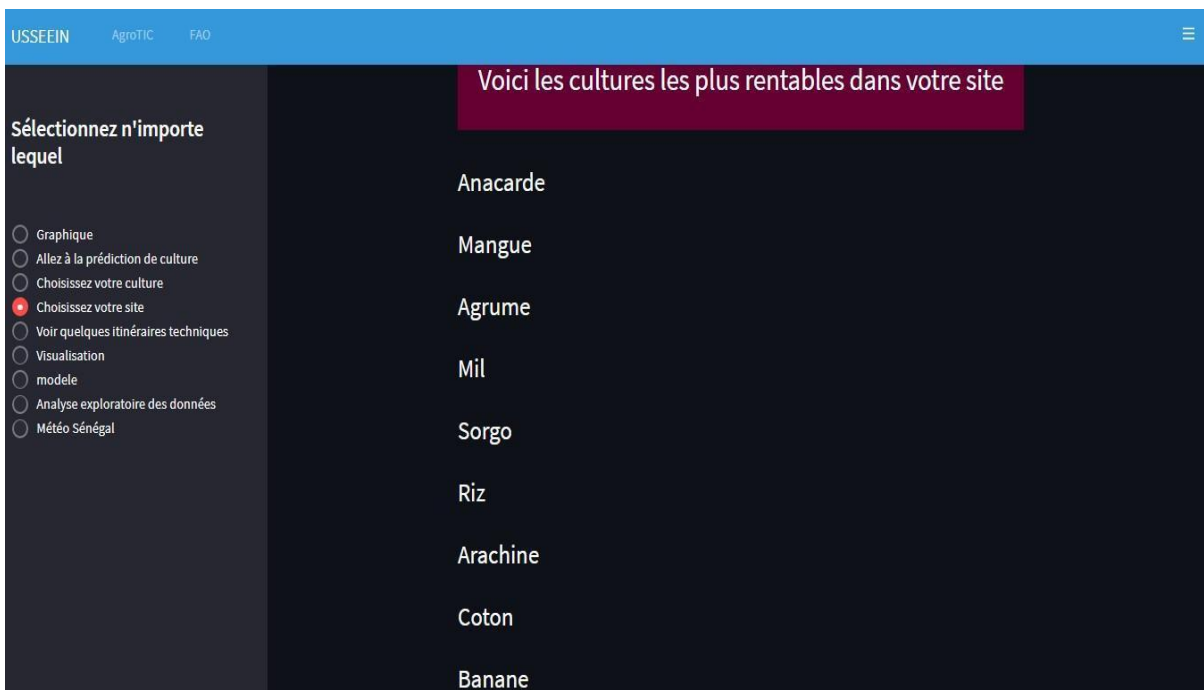


Fig. 23. Les cultures adaptées dans la région de Ziguinchor.

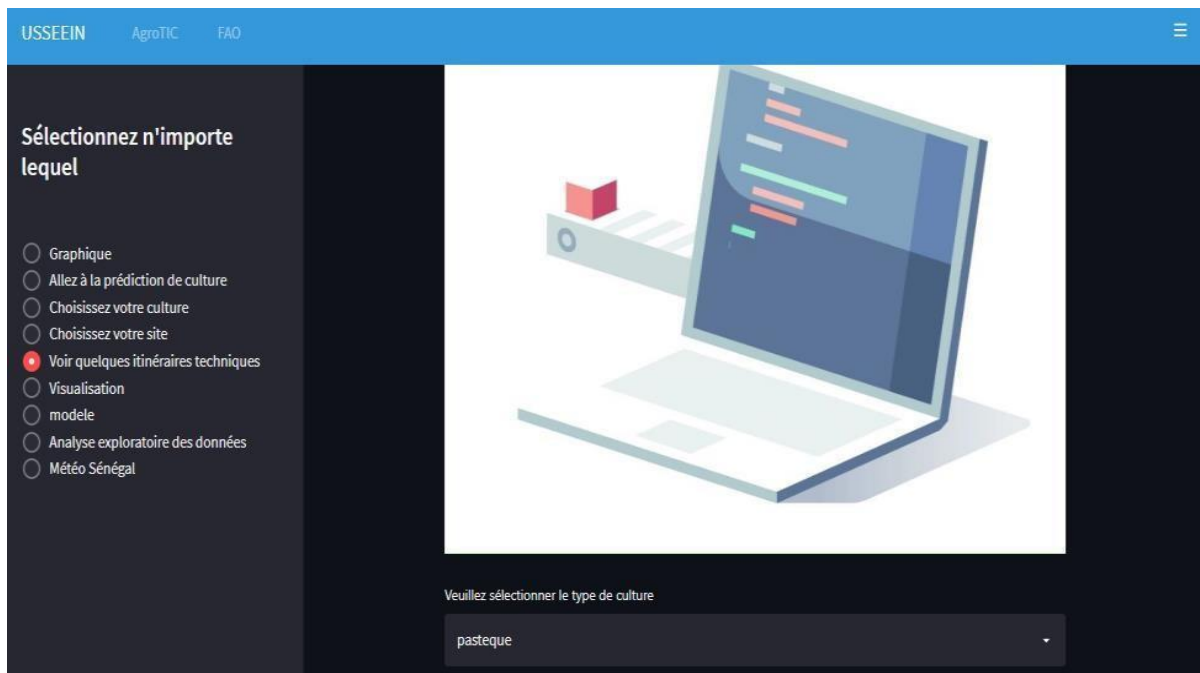


Fig. 24. Itinéraires techniques des cultures.

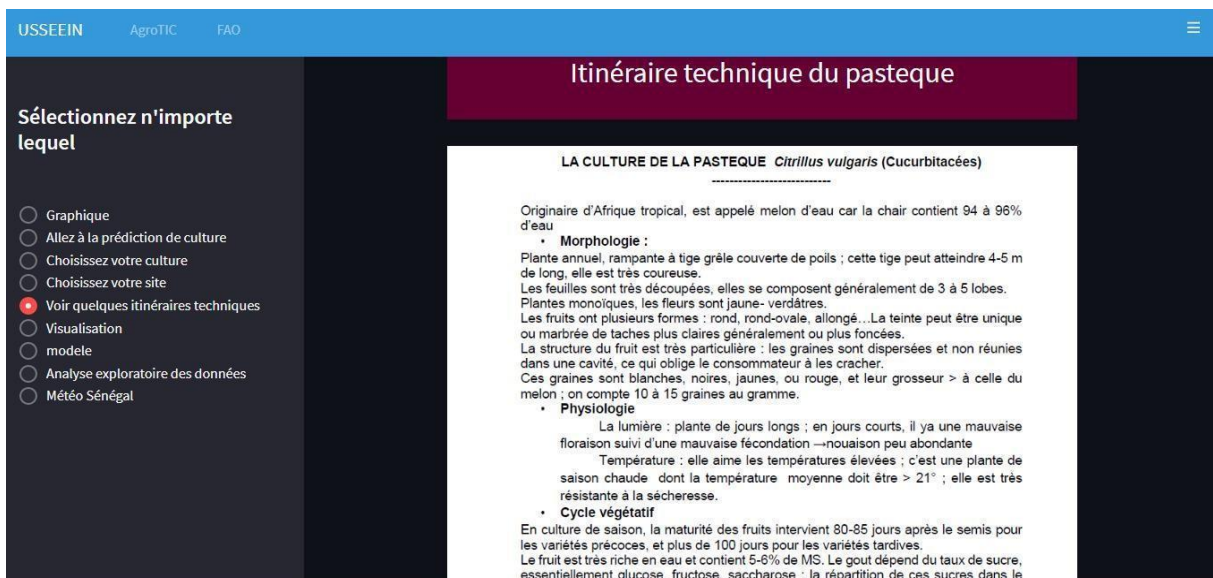


Fig. 25. Itinéraire technique de la pastèque.

La figure 25 nous montre l’itinéraire technique de la culture de la pastèque pour une meilleure pratique agricole. Poursuivons avec notre application AgriPredict, cette fois en cliquant sur le bouton « Visualisation » tel que montré à la figure 26. Une requête de téléchargement de l’ensemble des données s’affiche alors à partir de l’interface présentée à la figure 27.

Le fichier de données étant sélectionné, on peut accéder à son contenu grâce au bouton « Ouvrir ». Le résultat obtenu est présenté à la figure 28. Il apparaît l’ensemble des paramètres pour chaque type de culture. Il est ainsi possible de sélectionner le ou les colonnes qui nous intéressent comme indiqué dans la figure 29.

La visualisation des paramètres choisis obtenue initialement peut être nettement améliorée grâce à

la carte thermique. Cette carte thermique est présentée à la figure 30. Elle a l’avantage de permettre une meilleure analyse des différents paramètres. A droite du corrélogramme, la légende de couleurs montre les coefficients de corrélation et les couleurs correspondantes. On remarque dans cette légende que plus on s’approche de la couleur jaune, plus on a une meilleure corrélation des différents paramètres. Ainsi, le potassium (K) et le phosphore (P) ont une meilleure corrélation contrairement à l’azote et l’humidité de l’air qui présente une faible corrélation.

En simplifiant les paramètres pour une meilleure visualisation du corrélogramme pour mettre en évidence les paramètres corrélés et ceux non corrélés, on aboutit aux figures 31 et 32.

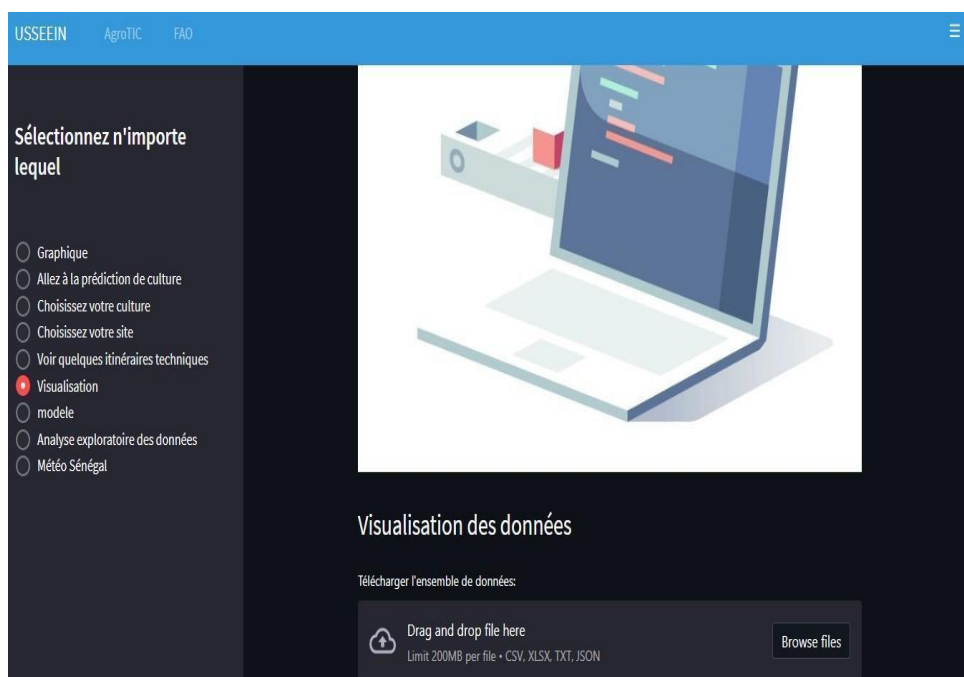


Fig. 26. Bouton de visualisation.

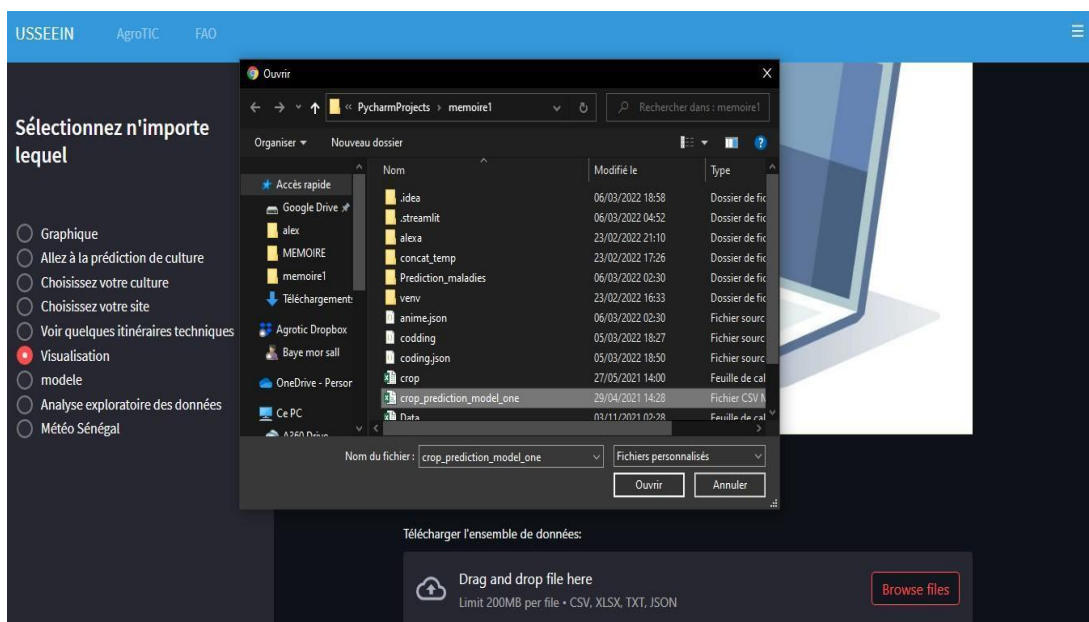


Fig. 27. Téléchargement des données.

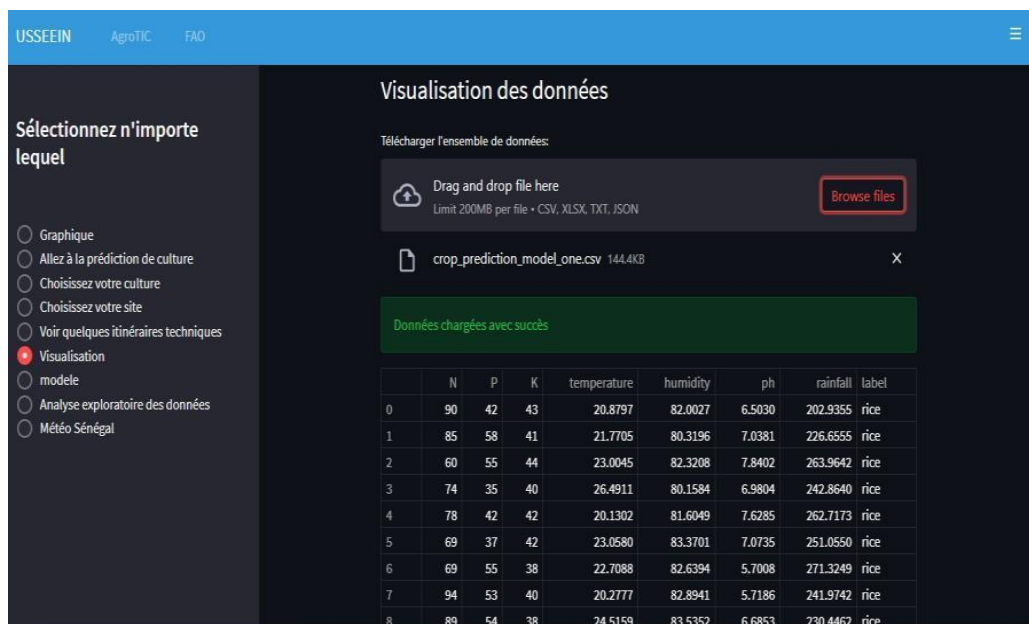


Fig. 28. Visualisation des données.

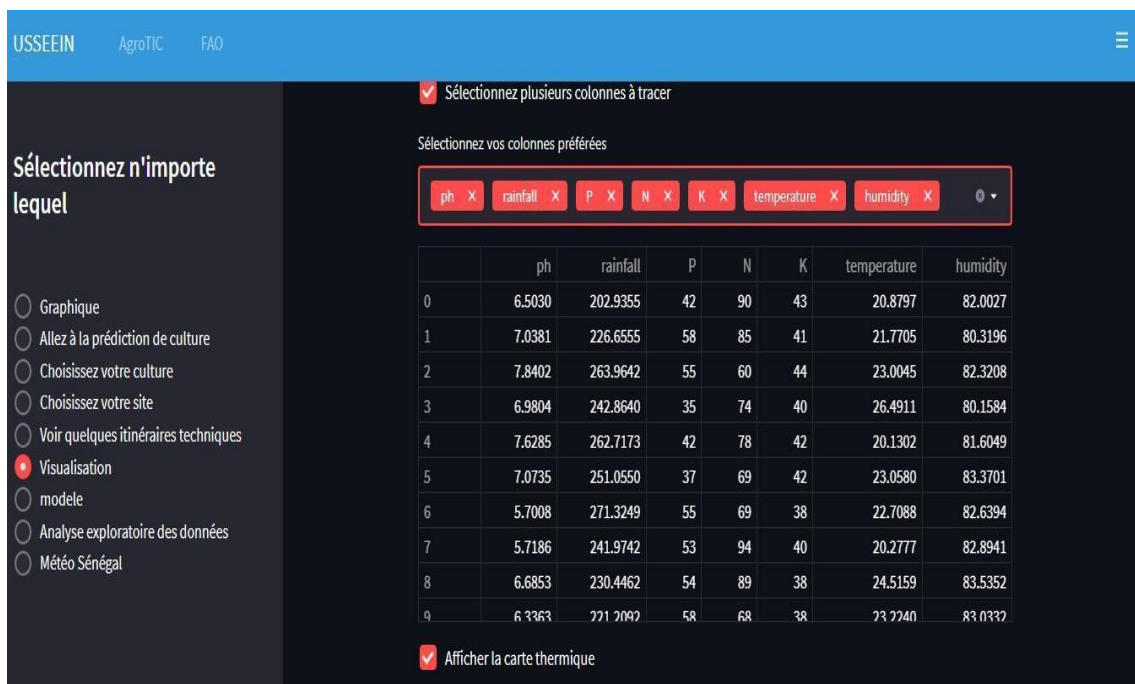


Fig. 29. Visualisation des paramètres choisis.

Le corrélogramme de la figure 31 montre que le potassium (K) et le phosphore (P) ont une meilleure corrélation qui est de 0,75. Par contre, le corrélogramme de la figure 32 montre une faible corrélation entre le pH et l'humidité de l'air qui est de -0.0085.

Passons à présent au bouton « Modèle » qui nous permet de faire un choix du modèle avec l'algorithme KNN en considérant les cultures

comme cible. L'exemple est indiqué dans la figure 33. Après avoir fait le choix du modèle avec l'algorithme KNN, il va nous prédire les cultures les plus favorables avec les différents attributs choisis. Le résultat est présenté à la figure 34. Cette figure montre avec une précision de 65,9% que les cultures sont favorables avec les attributs choisis.



Fig. 30. Carte thermique des paramètres.

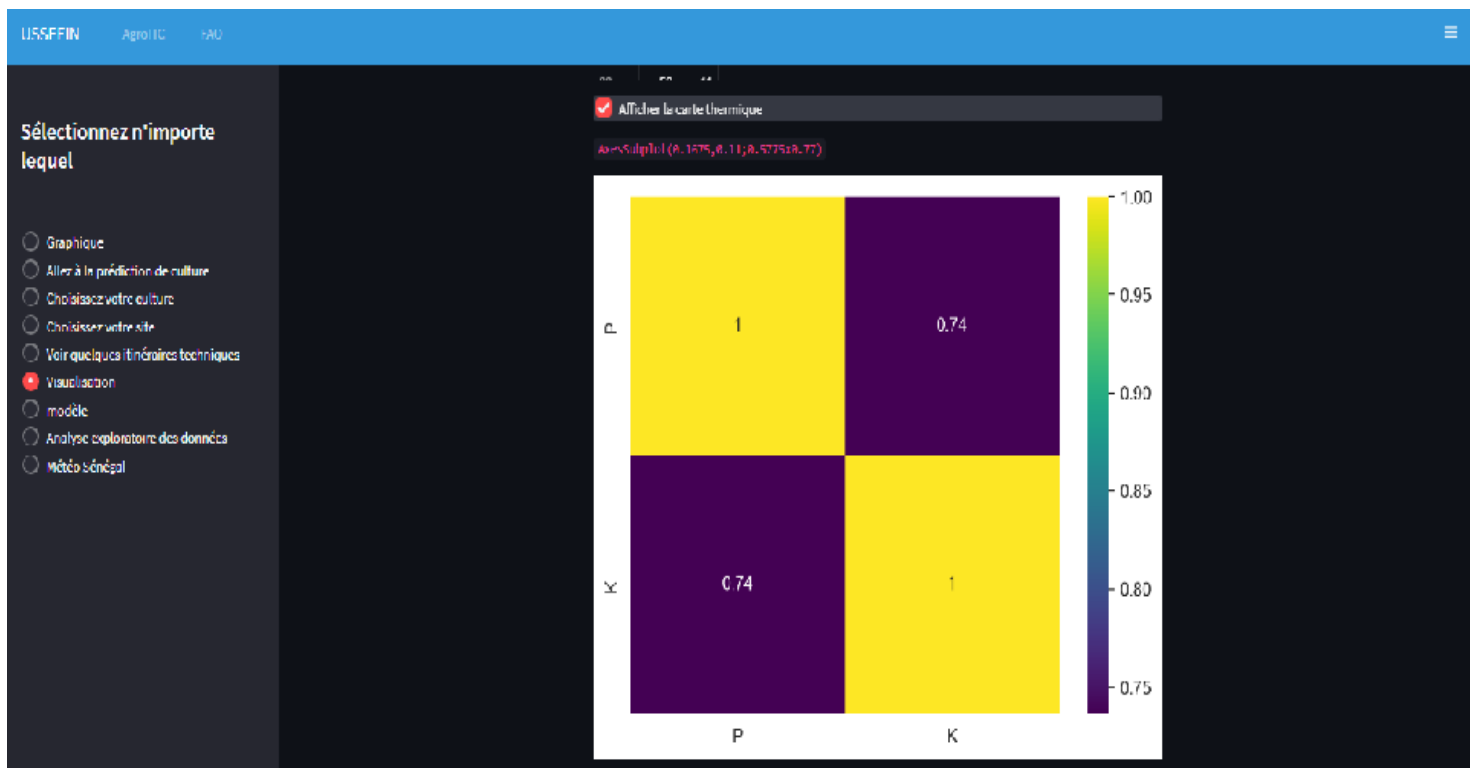


Fig. 31. Corrélogramme de deux paramètres corrélés.



Fig. 32. Corrélation de deux paramètres non corrélés.

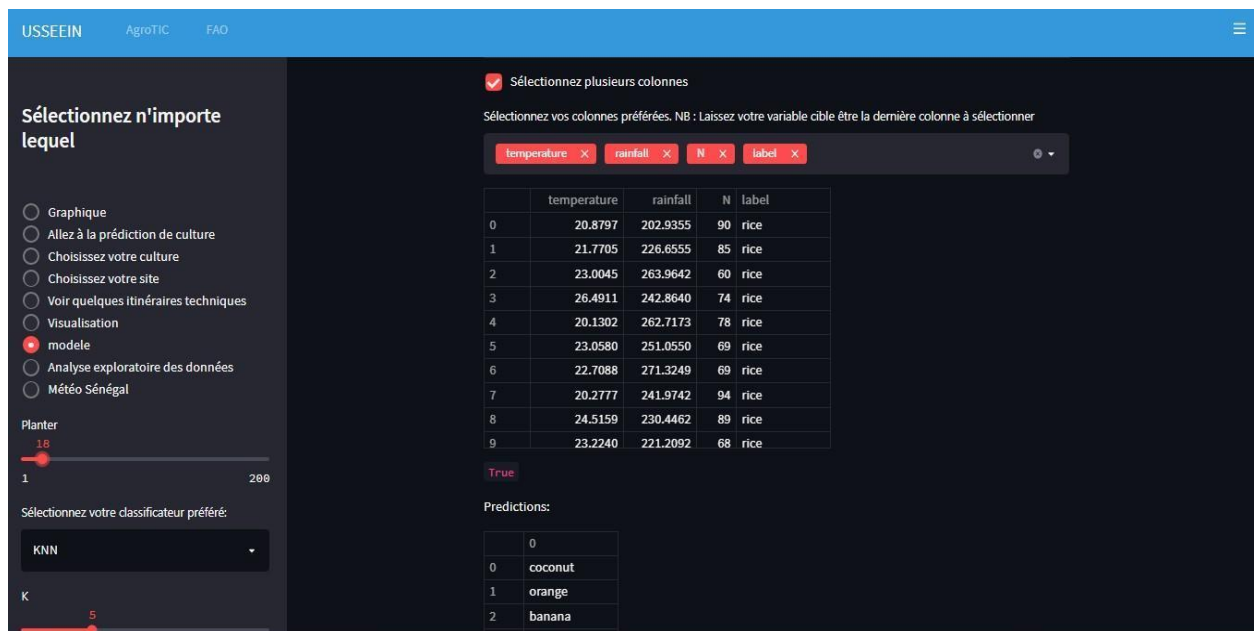


Fig. 33. Choix du modèle avec l’algorithme KNN pour la prédiction.



Fig. 34. Prédiction des cultures.

Analysons maintenant l’algorithme de régression linéaire (RL) pour prédire les cultures avec les paramètres choisis. Le résultat obtenu est présenté dans la figure 35. Avec les mêmes attributs que précédemment, on a une précision de 45,90% pour que les cultures soient favorables. Passons à présent au bouton « Analyse exploratoire des

données » tel que présenté à la figure 36. Ce choix permet d’afficher le résumé des attributs choisis pour une meilleure visualisation des données et enfin d’en faire une analyse. Le résultat est indiqué dans la figure 37.

Le bouton « Météo Sénégal » conduit à l’interface indiquée dans la figure 38. Cela est pertinent lors de la prise de décision dans le processus de choix des spéculations agricoles d’une zone précise. Il est ainsi possible de voir la météo dans une région de notre choix au Sénégal. Prenons l’exemple de la zone agro écologique de Dakar. Le résultat est présenté dans la figure 39. Il apparaît la météo de la région de Dakar à une date et heure précises, ainsi que d’autres paramètres utiles en agronomie.

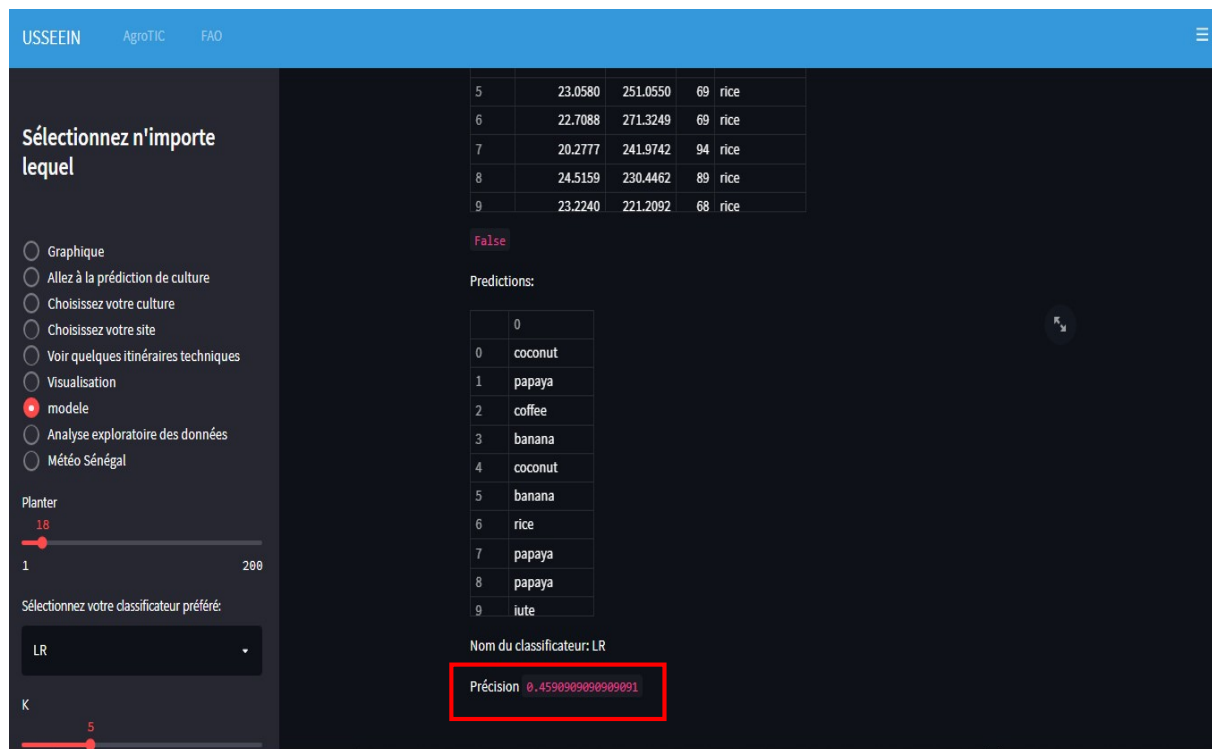


Fig. 35. Prédiction des cultures avec l'algorithme de régression linéaire.

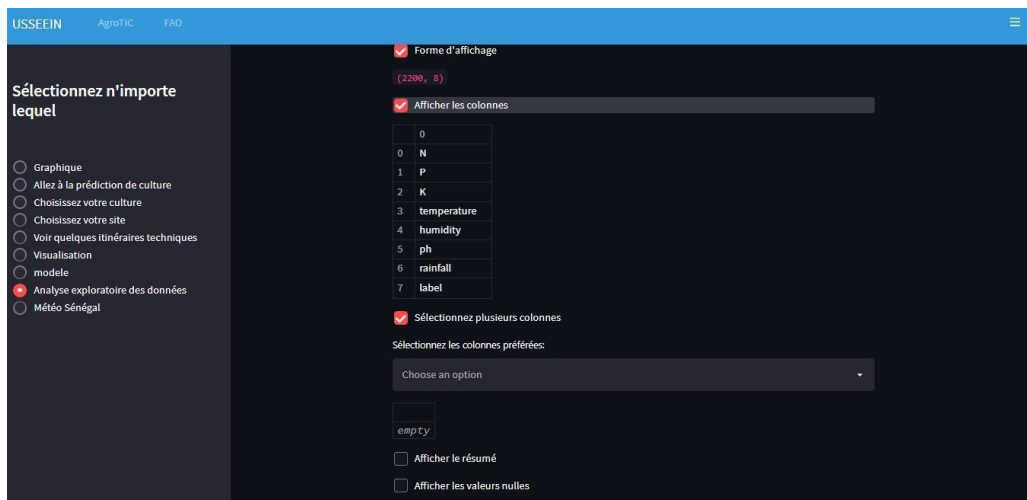


Fig. 36. Choix pour l'analyse exploratoire des données.

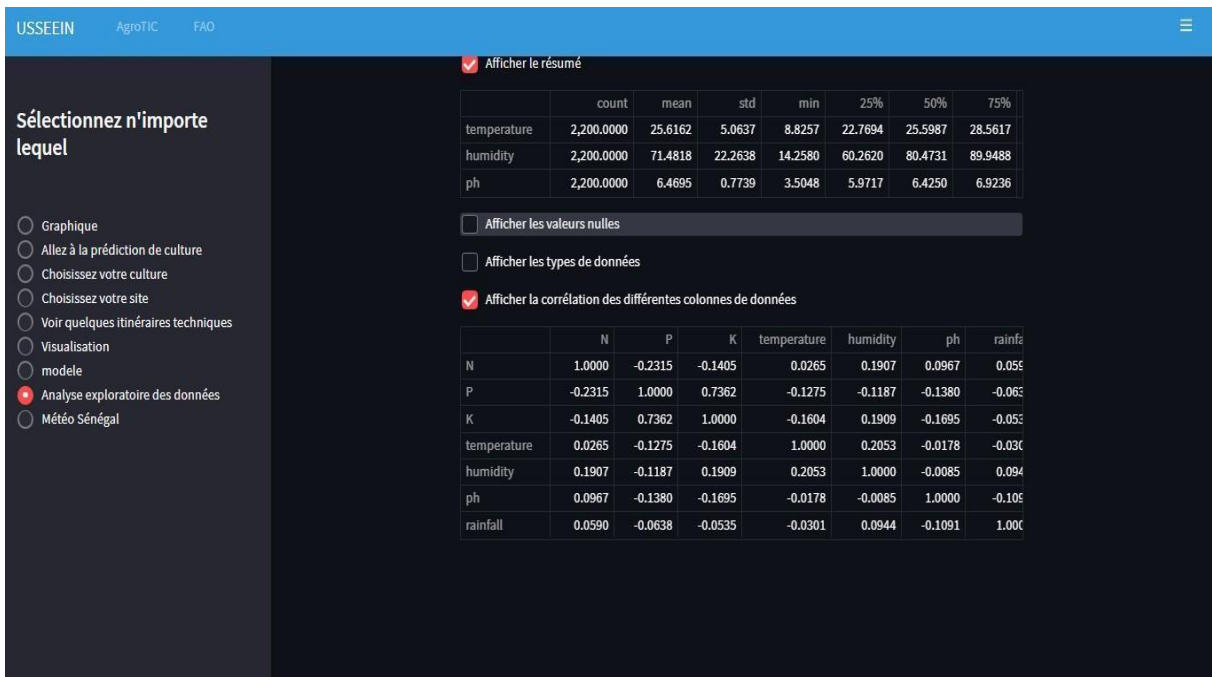


Fig. 37. Résumé des données choisis.

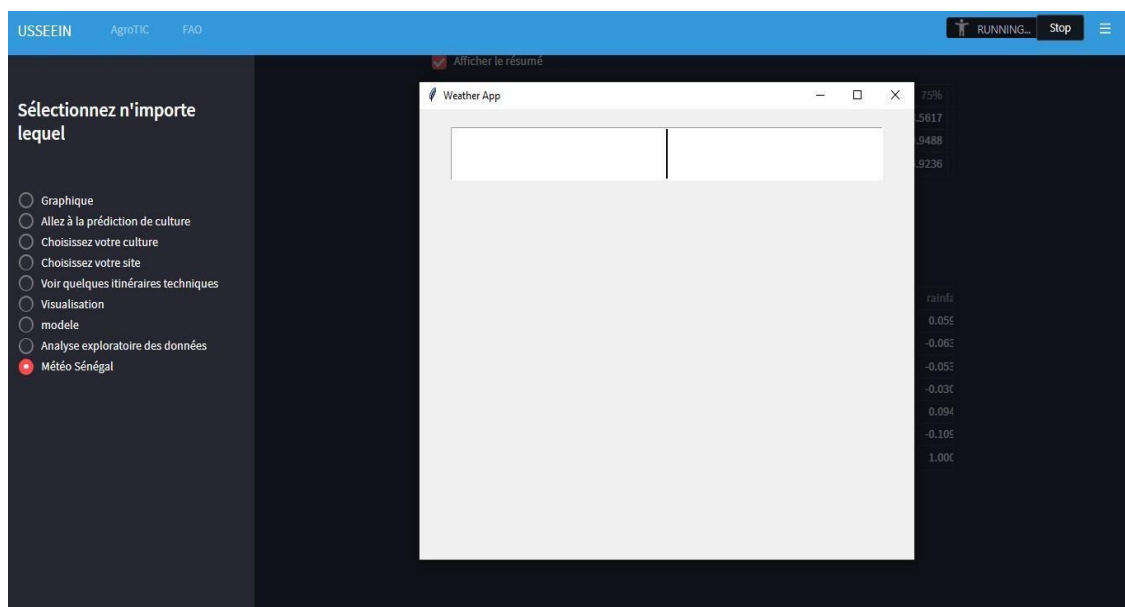


Fig. 38. L'interface du bouton météo Sénégal.

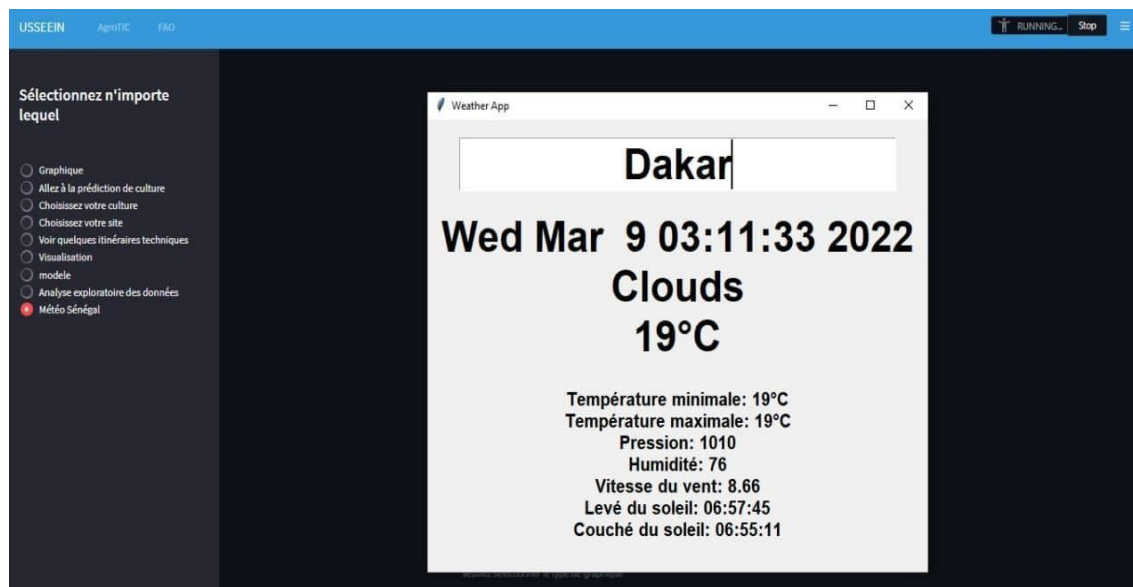


Fig. 39. Météo de la région de Dakar.

7. Conclusion

Sur la base des paramètres des entrées climatiques et des intrants, le présent travail a fourni une démonstration de l'utilisation des algorithmes basés sur la régression et la classification pour prédire les cultures. Les algorithmes sont k-nearest neighbors (k-MN), machine à vecteurs de support (SVM) et CART. Étant donné que notre système proposé est un système basé sur le Web, les variables d'entrées et les modules peuvent être facilement modifiés car de nouvelles fonctionnalités peuvent être ajoutées en fonction des besoins. Cela est aussi valable pour l'application mobile développée. Nos solutions donnent également des réponses rapides et précises aux agriculteurs. Nos travaux futurs vont examiner l'apprentissage automatique hybride tel que la régression multiple, la régression logistique, et les algorithmes d'apprentissage en profondeur (Deep Learning)

tel que le réseau de neurones à convolution profonde (DCNN) et la longue mémoire à court terme (LSTM) qui pourrait fournir une solution rapide et précise aux producteurs. Les travaux futurs incluront l'examen des grands ensembles de données issus de différents pays pour prédire le rendement des cultures à l'avance, la détection des maladies des plantes et les prédictions de la qualité des fruits.

Références bibliographiques

- [1] K.S.M. Anbananthen, S. Subbiah, D. Chelliah, P.Sivakumar, V. Somasundaram, K.H. Velshankar, M.K.A.A. Khan, *An intelligent decision support system for crop yield prediction using hybrid machine learning algorithms*, F1000Research 10 (2021) 1143. doi: 10.12688/f1000research.73009.1
- [2] S. Holzapfel, M. Brüntrup, *SDG 2 (Zero Hunger) in the context of the German Sustainable Development Strategy: are we leaving the starving behind?* Briefing paper (2017). <https://www.idos-research.de/en/briefing-paper/article/sdg-2-zero-hunger-in-the-context->

of-the-german-sustainable-development-strategy-are-we-leaving-the-starving-behind/ (consulté le 20 septembre 2022).

[3] D. Ramesh, B.V. Vardhan, *Analysis of crop yield prediction using data mining techniques*, Int. J. Res. Eng. Technol. 4(1) (2015) 470-473. doi: 10.15623/ijret.2015.0401071

[4] A. Patil, R. Medar, V. Desai, *Crop Yield Prediction Using Machine Learning Techniques*, Int. J. Sci. Res. Sci. Eng. Technol. 7(3) (2020) 312-315. doi: 10.32628/IJSRSET20736

[5] M. Jadhav, N. Kolambe, S. Jain, S. Chaudhari, *Farming Made Easy using Machine Learning*, 2nd Int. Conf. Emerg. Technol. INCET (2021). doi: 10.1109/INCET51464.2021.9456351

[6] A. Singh, N. Janu, S. Trivedi, M. Jain, *Precision Agriculture and Machine Learning*, IEEE World Conf. Appl. Intell. Comput. AIC (2022). doi: 10.1109/AIC55036.2022.9848898

[7] R. Medar, V. S. Rajpurohit, S. Shweta, *Crop Yield Prediction using Machine Learning Techniques*, IEEE 5th International Conference for Convergence in Technology (I2CT) (2019) 1-5. doi: 10.1109/I2CT45611.2019.9033611.

[8] S. Veeragandham, H. Santhi, *A Review on the Role of Machine Learning in Agriculture*. <https://www.semanticscholar.org/paper/Deep-Learning-for-Smart-Agriculture-Kavitha/0a272722fe4838cce5af0bb907310bf76927406d> (Consulté le: 20 septembre 2022)

[9] A. Kavitha, *Deep Learning for Smart Agriculture*, Int. J. Eng. Res. 9(5) (2021) 3.

[10] A.D. Salwe, L.S. Khedekar, *A Survey On Smart Agriculture Crop Yield Prediction Using Machine Learning Deep Learning And Cloud Computing*, JETIR 8(8) (2021) a864-a867.

[11] E. Manjula, S. Djodiltachoumy, *A Model for Prediction of Crop Yield*, Internat. J. Comput. Intel. Inform. 6(4) (2017) 8.

[12] S. Khaki, L. Wang, S. V. Archontoulis, *A CNN-RNN Framework for Crop Yield Prediction*, Front. Plant Sci. 10 (2020) 1750. doi: 10.3389/fpls.2019.01750

[13] H. Almahdi, *Predicting Crops Yield: Machine Learning Nanodegree Capstone Project*. <https://towardsdatascience.com/predicting-crops-yield-machine-learning-nanodegree-capstone-project-e6ec9349f69> (Consulté le 20 septembre 2022)

[14] M. Meena, P.K. Singh, *Crop Yield Forecasting Using Neural Networks, Swarm, Evolutionary, and Memetic Computing*, Cham. (2013) 319-331. doi: 10.1007/978-3-319-03756-1_29

[15] S. Dharmaraja, V. Jain, P. Anjoy, H. Chandra, *Empirical Analysis for Crop Yield Forecasting in India*, Agric. Res. 9(1) 132-138. doi: 10.1007/s40003-019-00413-x

[16] P.S. Birthal, T. Khan, D.S. Negi, S. Agarwal, *Impact of Climate Change on Yields of Major Food Crops in India: Implications for Food Security*, Agric. Econ. Res. Rev. 27(2) (2014) 145. doi: 10.5958/0974-0279.2014.00019.6

[17] A. Prema, D.M. Sujatha, R. Jegadeesan, V.S. Harsha, G. Snehitha, M. Chethana, *Crop Yield Prediction Based On Indian Agriculture Using Machine Learning*, INCET 7(932) (2021) 14. <https://api.semanticscholar.org/CorpusID:247448163>

[18] AgriApp : Smart Farming App - Apps on Google Play, Playstore. <https://play.google.com/store/apps/details?id=com.criyagen&hl=en&gl=SN> (consulté le 20 septembre 2022)

[19] POWER | Data Access Viewer. <https://power.larc.nasa.gov/data-access-viewer/> (consulté le 20 septembre 2022)

[20] S. K. Rakshith, *A Survey on Crop Prediction using Machine Learning Approach*, Int. J. Res. Appl. Sci. Eng. Technol. 7(4) (2019) 3231-3234. doi: 10.22214/ijraset.2019.4542

[21] MIT App Inventor <https://appinventor.mit.edu/> (consulté le 21 septembre 2022)

[22] MIT App Inventor <http://ai2.appinventor.mit.edu/#5715475661979648> (consulté le 21 septembre 2022)

[23] Welcome to Python.org, Python.org.
<https://www.python.org/>
(consulté le 21 septembre 2022)

[24] L'apprentissage supervisé - Machine Learning, Mr. Mint: Apprendre le Machine Learning de A à Z.
<https://mrmint.fr/apprentissage-supervise-machine-learning> (consulté le 21 septembre 2022)

[25] S. Sharma, G. Rathee, H. Saini, Big Data Analytics for Crop Prediction Mode Using Optimization Technique, Fifth Int. Conf. Parallel Distrib. Grid Comput. PDGC (2018).
doi: 10.1109/PDGC.2018.8746001.

[26] L'apprentissage non supervisé - Machine Learning, Mr. Mint: Apprendre le Machine Learning de A à Z.
<https://mrmint.fr/lapprentissage-non-supervise-machine-learning> (consulté le 21 septembre 2022)

[27] P.F. Faye, S. Aknine, M. Sene, O. Shehory, Dynamic Coalitions Formation in Dynamic Uncertain Environments, IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT) 2 (2015) 273-276. doi: 10.1109/WI-IAT.2015.86.

[28] T. Rédac, Reinforcement Learning : Définition et application, Formation Data Science| DataScientest.com.
<https://datascientest.com/reinforcement-learning> (consulté le 21 septembre 2022)

[29] P.F. Faye, S. Aknine, M. Sene, O. Sheory, Stabilizing Agent's Interactions in Dynamic Contexts, IEEE 28th International Conference on Advanced Information Networking and Applications (2014) 925-932.
doi: 10.1109/AINA.2014.113

[30] P.F. Faye, M. Senghor, G.A. Ndione, Stable Trade Coordinations in Smart Agriculture, IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), Niagara Falls, ON, Canada (2022) 215-222.
doi: 10.1109/WI-IAT55865.2022.00038.

[31] A. Géron, Machine Learning avec Scikit-Learn - Mise en oeuvre et cas concrets, Dunod, 2^{ème} édition, 2019, 320 pages, ISBN-10 : 210079065X, ISBN-13 : 978-2100790654