RÉPUBLIQUE DU SÉNÉGAL

Un peuple - un but - une foi

¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR, DE LA RECHERCHE ET DE L'INNOVATON

¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤

L'excellence ma référence

¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤

Mention : Management des systèmes d'informations Automatisés

Département : Économie Gestion

UFR : Science Économique Sociale

¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤

SUJET :

Prédiction des niveaux de stock par machine-learning pour une gestion optimale des approvisionnements

¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤

Présenté par :

Anna DIOP

Directeur de mémoire

Professeur Edouard Ngor SARR

Codirecteur de mémoire

Professeur Amon Anike DEH

Membres du jury :

Professeur Serigne DIOP	UASZ	**Président**
Docteur Abel DIATTA	UASZ	**Examinateur**
Docteur Fulgence MANSAL	UAM	**Examinateur**

Année universitaire : 2025–2026

Soutenu publiquement le 27/07/2026

Dédicace

Je dédie ce travail à :

- Mon père Alioune DIOP ;
- Ma mère Khady DIALLO ;
- Ma tante Yaye Anna DIALLO.

Remerciements

J'adresse mes sincères remerciements :

À mes directeurs de mémoire le Prof Edouard Ngor SARR et le Prof Amon Anike DEH pour la rigueur, les conseils, les enseignements, la disponibilité et l'accompagnement tout au long de ce processus ;
Aux membres du jury, d'avoir accepté d'évaluer ce travail et pour leurs observations et conseils ;

À tout le corps enseignant de la filière MIO (management informatisé des organisations), je formule mes sincères remerciements pour la rigueur de votre encadrement et la richesse de vos enseignements ;

À Mr Ibou NGOM pour son aide, ces conseils et son expertise ;
À mes frères Pape Abdoulaye DIOP, Mouhamadou Moustapha DIOP et Amadou Wéllé DIALLO, merci pour vos encouragements et votre soutient, merci d'avoir toujours été là ;
À mes sœurs Ndéye Dado DIOP et Aminata DIOP qui m'ont toujours conseillée et soutenu durant ces années d'étude et durant toute ma vie, merci du fond du cœur ;
À mon maître coranique Mouhamed Mansour Kane DIALLO, merci de toujours me guider et surtout merci de m'avoir appris le coran et ma religion ;
À Birane SENE merci pour ton soutien, tes encouragements, tes conseilles et ta présence, merci de m'accompagner ;
À mes très chers Papa Alioune DIALLO et Mouhamed COLY, je ne remercierai jamais accès le ciel de vous avoir mis sur mon chemin. Je vous suis énormément reconnaissante pour tout ce que vous faites pour ma personne, merci de croire en moi et rendez-vous au sommeil ;
À Modou GUEYE merci pour toutes ces années de partage ;
À ma famille de Ziguinchor, La famille MENDY, merci de m'avoir ouvert vos portes, de m'avoir chaleureusement accueillit, je vous témoigne ma sincère reconnaissance ;
À mes amis : Nguissaly SY, Aissatou Lamine BARRY, Maimouna DARY, Elhadj Talla Niang FALL et Aminata DIAGNE, merci pour toutes ces années d'amitié qui ne perdure ;
À mes belles rencontres de Ziguinchor : Dr Abdou BOMOU, Abdoukhadre BALDE (BKB), Mouhamed DIALLO, Abdallah BA, Aly TOP, Mika NDIAYE, Ismaïla KEITA, Absa FALL, André DASYLVA, Raimond Célestin DIOP, Awa DEME, Aissata HANE, Dieumb NDONGO, Moussa DIOUF, Saliou BA, Mbissane KASSE, Ramatoulaye DIALLO, merci pour tous ces moments précieux ;

À Abdoulaye SY, Matar NAHAM, Déthié AW, Alimatou DIALLO, merci ;

À tous mes camarades de classe, merci pour toutes ces années de partage ;

À toute la famille NGONGO, merci pour votre hospitalité.

Résumé

L'objectif de ce mémoire est de présenter une manière d'optimiser ses stocks et son processus d'approvisionnement. La prédiction des stocks est cruciale pour éviter les ruptures de stock et les surstocks. De ce fait, pour une gestion optimale des approvisionnements nous avons utilisé des outils de prédiction basés sur le machine learning supervisé que sont le Random Forest et le XGBoost. Ils contribuent à améliorer et affiner les prédictions en analysant de grandes quantités de données historiques et en identifiant les cycles de demande, les délais de livraison et les événements externes. Ces données secondaires proviennent d'une entreprise de distribution basée au Sénégal. Ainsi, après une collecte, analyse et traitement de données, nous montrerons comment optimiser le processus d'approvisionnement d'une grande distribution. Pour ce faire la méthode utilisée est le constructivisme pour permettre à toutes entreprises de distribution de pouvoir réduire ces couts de stockage ainsi que ses ruptures de stock. Nos résultats montrent que la combinaison des deux techniques de ML permet d'améliorer la précision des prédictions de stock conduisant ainsi à une bonne gestion des approvisionnements. Donc, ce travail pourra aider les gestionnaires de stock et approvisionnement à stratégiquement mieux faire leur travail.

Mots clés : Stocks-Approvisionnements, Gestion-Stock, Approvisionnements, Machine-learning.

Abstract

The objective of this thesis is to present a method for optimizing inventory and supply chain processes. Inventory prediction is crucial for avoiding stockouts and overstocking. Therefore, for optimal supply chain management, we used predictive tools based on supervised machine learning, namely Random Forest and XGBoost. These tools help improve and refine predictions by analyzing large amounts of historical data and identifying demand cycles, delivery times, and external events. This secondary data comes from a distribution company based in Senegal. Thus, after data collection, analysis, and processing, we will demonstrate how to optimize the supply chain process of a large retailer. To do this, we use constructivism, enabling all distribution companies to reduce their storage costs and stockouts. Our results show that combining these two machine learning techniques improves the accuracy of inventory predictions, leading to effective supply chain management. Therefore, this work can help inventory and supply managers to strategically improve their performance.

Keywords : Inventory-Supply, Inventory Management, Supply Chain, Machine Learning.

Sommaire

Liste des figures

Liste des tableaux

Sigles et abréviations

- **AI** : Artificial Intelligence (Intelligence Artificielle)
- **ANN** : Artificial Neural Networks (Réseaux de Neurones Artificiels)
- **API** : Application Programming Interface (Interface de Programmation d'Application)
- **ARIMA** : Auto-Regressive Integrated Moving Average (Modèle Autorégressif Intégré de Moyennes Mobiles)
- **AWS** : Amazon Web Services (Fournisseur de services cloud)
- **Bias / MBE** : Mean Bias Error (Erreur de Biais Moyenne)
- **BPNN** : Back-Propagation Neural Network (Réseau de Neurones à Rétropropagation)
- **CI/CD** : Continuous Integration / Continuous Deployment (Intégration Continue / Déploiement Continu)
- **CMPC** : Coût moyen pondérée du capital
- **CMV** : Coût des Marchandises Vendues
- **CNN** : Convolutional Neural Networks (Réseaux de Neurones Convolutifs)
- **CRBM** : Conditional Restricted Boltzmann Machine (Machine de Boltzmann Restreinte Conditionnelle)
- **CRM** : Customer Relationship Management (Gestion de la Relation Client)
- **DRCI** : Délais de rentabilité du capital investi
- **EDA** : Exploratory Data Analysis (Analyse Exploratoire des Données)
- **ENSAJ** : École Nationale des Sciences Appliquées d'El Jadida
- **EOQ** : Economic Order Quantity (Quantité Économique de Commande)
- **ERP** : Enterprise Resource Planning (Progiciel de Gestion Intégré)
- **ETL** : Extract, Transform, Load (Extraire, Transformer, Charger)
- **FCRBM** : Factored Conditional Restricted Boltzmann Machine (Machine de Boltzmann Restreinte Conditionnelle Factorisée)
- **GCP** : Google Cloud Platform (Fournisseur de services cloud)
- **GPA** : Gestion Partagée des Approvisionnements
- **GRU** : Gated Recurrent Unit (Unité Récurrente à Portes)
- **GSA** : Gestion des Stocks et Approvisionnements
- **HMM** : Hidden Markov Models (Modèles de Markov Cachés)
- **IA** : Intelligence Artificielle

- **IASO** : Improved Ant Swarm Optimization (Optimisation par Essaim de Fourmis Améliorée)
- **IDE** : Integrated Development Environment (Environnement de Développement Intégré)
- **IP** : Indice de profitabilité
- **KNN** : K-Nearest Neighbors (K-Plus Proches Voisins)
- **L1 / L2** : Régularisations de type Lasso L1 et Ridge L2 (Techniques de pénalisation des coefficients)
- **LSTM** : Long Short-Term Memory (Réseau de Neurones Récurrent à Mémoire à Long et Court Terme)
- **LTI** : Laboratoire de Technologies de l'Information
- **MAE** : Mean Absolute Error (Erreur Absolue Moyenne)
- **MAPE** : Mean Absolute Percentage Error (Erreur Percentuelle Absolue Moyenne)
- **MedAE** : Median Absolute Error (Erreur Absolue Médiane)
- **ML** : Machine Learning (Apprentissage Automatique)
- **MRP** : Material Requirements Planning (Planification des Besoins en Composants/Matières)
- **MSE** : Mean Squared Error (Erreur Quadratique Moyenne)
- **NAR** : Nonlinear Auto-Regressive (Modèle Autorégressif Non Linéaire)
- **NARX** : Nonlinear Auto-Regressive with Exogenous inputs (Modèle Autorégressif Non Linéaire avec Entrées Exogènes)
- **NNLS** : Non-Negative Least Squares (Moindres Carrés Non Négatifs)
- **NumPy** : Numerical Python (Bibliothèque de calcul numérique)
- **OA** : Ordre d'Approvisionnement
- **OF** : Ordre de Fabrication
- **OLS** : Ordinary Least Squares (Moindres Carrés Ordinaires)
- **OOF** : Out-Of-Fold (Prédictions hors-échantillon générées lors de la validation croisée)
- **PCC** : Pearson Correlation Coefficient (Coefficient de Corrélation de Pearson)
- **PDP** : Plan Directeur de Production
- **POS** : Point Of Sale (Point de Vente / Terminal de Caisse)
- **PSO** : Particle Swarm Optimization (Optimisation par Essaim Particulaire)
- **PyTorch** : Bibliothèque Python de calcul tensoriel et de Deep Learning
- **R^2 / R^2** : Coefficient de détermination (Mesure de la qualité de la prédiction)

- **RBM** : Restricted Boltzmann Machine (Machine de Boltzmann Restreinte)
- **RF** : Random Forest (Forêt Aléatoire)
- **RMSE** : Root Mean Squared Error (Racine de l'Erreur Quadratique Moyenne)
- **RMSLE** : Root Mean Squared Logarithmic Error (Racine de l'Erreur Logarithmique Quadratique Moyenne)
- **RNN** : Recurrent Neural Networks (Réseaux de Neurones Récurrents)
- **ROP** : Reorder Point (Point de Réapprovisionnement)
- **RRMSE** : Relative Root Mean Squared Error (Erreur Quadratique Moyenne Relative)
- **SAPSO** : Simulated Annealing Particle Swarm Optimization (Optimisation par Essaim Particulaire avec Recuit Simulé)
- **Scikit-learn** : Bibliothèque d'apprentissage automatique pour Python
- **SI** : Swarm Intelligence (Intelligence en Essaim)
- **SMAPE** : Symmetric Mean Absolute Percentage Error (Erreur Percentuelle Absolue Moyenne Symétrique)
- **SS** : Safety Stock (Stock de Sécurité)
- **SVM** : Support Vector Machine (Machine à Vecteurs de Support)
- **TRI** : Taux de rentabilité interne
- **VAN** : Valeur actuelle nette
- **VMI** : Vendor Managed Inventory (Stocks Gérés par le Fournisseur)
- **VS** : Versus (Face à / Contre)
- **VS Code** : Visual Studio Code
- **WMS** : Warehouse Management System (Système de Gestion d'Entrepôt)
- **XGB / XGBoost** : Extreme Gradient Boosting

Introduction générale

Au cours des dernières décennies, les entreprises ont évolué dans un environnement économique de plus en plus concurrentiel, marqué par la mondialisation des échanges, la diversification des produits et l'évolution rapide des attentes des consommateurs. Les clients exigent désormais une disponibilité permanente des produits, la qualité au rendez-vous et un niveau de service toujours plus élevé [48, 49]. Dans ce contexte, les entreprises doivent assurer un équilibre entre la satisfaction de la demande et la maîtrise des coûts logistiques. La gestion des stocks ne se limite donc plus au simple stockage des marchandises, elle constitue un véritable levier de performance permettant d'améliorer la compétitivité, la rentabilité et la qualité du service offert aux clients [48, 49]. Par ailleurs, les récentes perturbations des chaînes d'approvisionnement, telles que les fluctuations des marchés ou encore les variations imprévisibles de la demande, ont mis en évidence les limites des approches traditionnelles de gestion des stocks [50, 51]. Ces événements ont montré l'importance de disposer d'outils capables d'anticiper les besoins futurs afin de réduire les risques de rupture ou de sur-stockage. Ainsi, la gestion des stocks est un défi majeur pour les entreprises de tous secteurs. C'est un enjeu stratégique pour les entreprises, car elle impacte directement sur la satisfaction client mais aussi sur la trésorerie de l'entreprise. En effet, un stock suffisant assure et répond rapidement aux besoins de la clientèle, tandis qu'une insuffisance de stock entraîne une rupture de stock, des clients mécontents et une perte des ventes [10]. Le sur stockage génère des coûts inutiles. Donc, une optimisation des approvisionnements est cruciale afin d'éviter aux entreprises tous ces désagréments. Pendant longtemps, les entreprises ont comptés sur des méthodes classiques. En effet, traditionnellement, la gestion des stocks repose sur des modèles statiques comme la Quantité Économique de Commande (EOQ) ou la Planification des Besoins en Composants (MRP), les méthodes ARIMA [33] etc. Ces méthodes présentent toutefois des limites dans le contexte industriel moderne et lorsque les demandes deviennent irrégulières. Elles peinent à intégrer des variables externes complexes comme les périodes de fête, les débuts et fins de mois ou les changements brusques de comportement des consommateurs [33]. Elles s'appuient souvent sur des moyennes historiques et des tendances linéaires qui ne capturent pas les tendances réelles du marché [33]. Par ailleurs, la transformation numérique des entreprises a conduit à une augmentation importante des données disponibles sur les ventes, les produits, les fournisseurs et les comportements des consommateurs. Ces données constituent une source d'information précieuse qui peut être exploitée afin d'améliorer la qualité des prévisions et d'optimiser les décisions d'approvisionnement. C'est ici que l'apport de l'apprentissage

automatique (le Machine Learning) devient stratégique. Ainsi, avec le ML, nous aimerions mettre en place des stratégies et des outils pour gérer de manière efficace et efficiente les flux de marchandises et ainsi avoir connaissance sur les besoins en approvisionnement. D'un point de vue pratique, piloter les flux dans le domaine stock et approvisionnement, consiste à prendre des décisions à chaque étape de la chaîne (depuis les fournisseurs jusqu'au client final) et pour chaque entité (matière première, composant, produit intermédiaire ou produit fini), permettant de déterminer quand et en quelle quantité lancer une activité telle que l'activité d'approvisionnement, de transport, de fabrication ou d'assemblage [4]. Toutes fois, dans le contexte africain et plus particulièrement au Sénégal, rare sont les entreprises de distribution qui utilisent des solutions basées sur la prédiction par machine learning pour leur gestion des stocks et approvisionnements.

Face à ce constat, il est ainsi donc crucial de se poser la question selon laquelle : ***En quoi l'usage du machine learning peut aider à obtenir des prédictions beaucoup plus précises et permettre de prendre des décisions plus éclairées dans le processus d'approvisionnement de la grande distribution ?*** En d'autres termes :

- Quelle est l'efficacité des algorithmes de machine learning sur la prédiction ?
- Peut-on espérer une gestion plus intelligente des approvisionnements avec l'influence du machine learning ?

Face à cette problématique, nous proposons de mettre en place une stratégie de prédiction des niveaux de stock et un mode d'approvisionnement qui tient en compte les périodes de fête, la saisonnalité et autres.

L'objectif général de cette étude est de montrer l'utilité de l'usage du machine learning dans l'optimisation du processus d'approvisionnement dans la grande distribution. Nous nous assurerons donc qu'ils puissent grâce à ce dernier :

- De montrer l'efficacité du machine learning sur l'amélioration de l'exactitude des prédictions ;
- De montrer l'amélioration sur la gestion des approvisionnements avec le machine learning.

Cette étude profite à toutes entreprises de distribution. En effet cela leur permettra de :

- Réduire les couts de stockage et les ruptures de stock ;
- Assurer la précision des prédictions et renforcer leur compétitivité ;
- Assurer la fiabilité de l'entreprise et une meilleure satisfaction client ;
- Adaptation rapide aux changements.

Elle peut aussi profiter à la communauté scientifique et pour les chercheurs en :

- Contribuant à la littérature scientifique ;
- Exploitant de nouvelles méthodologies ;
- Développant de solutions innovantes.

Afin de mener à bien notre travail, nous procèderons par une approche mixte combinant une méthode qualitative et quantitative. Le choix de cette méthode est justifié du fait qu'elle nous permet d'analyser et de comprendre le comportement des clients en nous facilitant leur satisfaction et même d'anticiper leurs besoins. En effet, on procèdera à une collecte et traitement des données pour obtenir des résultats statistiquement significatifs.

Notre plan se décomposera en quatre chapitres :

- **Chapitre 1 : Approche conceptuelle et revue de la littérature théorique :** une première partie qui, consacrée aux préalables de l'étude, nous permettra de présenter les concepts clés du sujet. En effet nous étalerons les bases du stock et de l'approvisionnement et parallèlement l'importance de leurs bonnes gestions ;
- **Chapitre 2 : État de l'art et positionnement :** dans cette partie nous allons explorer les dernières recherches et différentes méthodes de gestion des stocks et de prédiction des approvisionnements déjà mis en place puis nous positionner par rapport à nos attentes ;
- **Chapitre 3 : Méthodologie et Conception de la solution :** ce chapitre présente la démarche adoptée pour concevoir la solution de prédiction des niveaux de stock. Il décrit les principales étapes, allant de la collecte et la préparation des données jusqu'au choix et à la mise en œuvre du modèle de prédiction. L'objectif est de transformer les données historiques en prévisions fiables en s'appuyant sur des méthodes adaptées et performantes ;
- **Chapitre 4 : Présentation et analyse des résultats :** un quatrième chapitre consacré à la présentation des résultats issus des expérimentations réalisées. Il met en évidence

les performances des modèles développés. Par ailleurs, il propose une analyse des résultats obtenus, permettant de formuler des recommandations pour l'optimisation des stratégies d'approvisionnement.

Chapitre 1 : Approche conceptuelle et revue de la littérature théorique

1.1. Introduction

La maîtrise des niveaux de stock constitue aujourd'hui un enjeu stratégique pour les entreprises cherchant à optimiser leurs coûts, améliorer leur réactivité et renforcer la performance globale de leur chaîne d'approvisionnement. Dans ce contexte, les avancées de l'intelligence artificielle, et plus particulièrement du Machine Learning, offrent de nouvelles opportunités pour anticiper la demande, réduire les ruptures et mieux planifier les approvisionnements. Afin de comprendre le cadre conceptuel de cette étude et de définir les exigences du projet, la clarification des mots-clés essentiels, ainsi qu'à la présentation du cahier des charges. Cette étape vise à expliciter les concepts fondamentaux tels que le machine learning et ces algorithmes d'apprentissage, la gestion des stocks et des approvisionnements et la prédiction. Ces termes constituent les bases théoriques nécessaires à la compréhension du projet. Cette clarification permet de lever toute ambiguïté et d'établir un cadre conceptuel rigoureux.

1.2. Approche conceptuelle

1.2.1. Le machine-learning

1.2.1.1. Définition

Le Machine Learning ou apprentissage automatique est un domaine de l'IA qui permet aux ordinateurs d'apprendre à partir de données sans être explicitement programmés. Au lieu de suivre des instructions étape par étape, les algorithmes de Machine Learning analysent des ensembles de données pour identifier des tendances, des relations et des modèles, et utilisent ces informations pour faire des prédictions ou prendre des décisions [7, 16, 18]. Un programme informatique est considéré comme « apprenant » s'il améliore ses performances à une tâche donnée grâce à l'expérience. Cela signifie que plus le modèle est exposé à des données, plus il devient précis dans ses prédictions ou décisions [15].

L'objectif principal des techniques de Machine Learning est de produire un modèle qui peut être utilisé pour effectuer des tâches telles que la classification, la prédiction, l'estimation et autres [16].

1.2.1.2. Les techniques d'apprentissage du machine learning

Le Machine Learning est divisé en plusieurs types d'apprentissages, dont l'apprentissage supervisé, l'apprentissage non supervisé, l'apprentissage par renforcement et semi-supervisé.

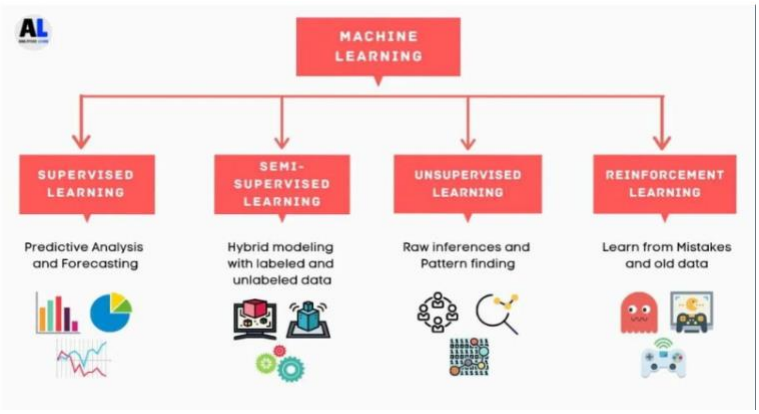


Figure 1. Les types de machine-learning[1]

Source : Web

1.2.1.2.1. L'apprentissage supervisé

L'apprentissage supervisé consiste à utiliser des ensembles de données étiquetées pour entraîner les algorithmes à classer les données ou à prédire des résultats avec précision. Au fur et à mesure que le modèle reçoit les données d'entrée, il ajuste ses paramètres par pondération jusqu'à atteindre un niveau optimal grâce à un processus de validation croisée. Cette méthode permet aux entreprises de résoudre divers problèmes à grande échelle, tels que le filtrage des courriels indésirables dans des dossiers autres que la boîte de réception[2]. En de termes plus simples, l'apprentissage supervisé c'est comme apprendre à un ordinateur ce qu'est une chose en lui montrant l'image de la chose et ses aspects. Il existe principalement deux types d'apprentissage supervisé :

- Classification : l'objectif est de prédire une classe ou une catégorie. Par exemple, classer un courriel comme spam ou non spam ;
- Régression : l'objectif est de prédire une valeur numérique. Par exemple, prédire le prix d'une maison.

La figure 2 représente de manière schématique le fonctionnement du processus d'apprentissage supervisé, une technique fondamentale du Machine Learning. Dans la partie supérieure, l'algorithme reçoit un ensemble de données d'entraînement composées d'exemples déjà étiquetés. Chaque observation (ici, des images d'animaux) est associée à une catégorie connue : certaines sont identifiées comme Canard (étiquette positive) et d'autres comme pas Canard (étiquette négative). À partir de ces exemples, le système apprend à reconnaître les

[1] Quels sont les types d'apprentissage automatique ? - En détail - AnalyticsLearn consulté le 22/11/2024
[2] https://www.ibm.com/fr-fr/topics/supervised-learning?mhsrc consulté le 22/11/2024

caractéristiques distinctives qui permettent de différencier les canards des autres animaux, telles que la forme du bec, la texture du plumage ou encore la posture. Ce processus constitue la phase d'apprentissage, au cours de laquelle le modèle construit une représentation interne de la réalité observée. Dans la partie inférieure, le modèle entraîné est mis à l'épreuve : on lui présente une nouvelle image inconnue, qu'il doit analyser à l'aide des connaissances acquises. Grâce aux régularités qu'il a apprises, le modèle est alors capable de formuler une prédiction et de classer l'image comme appartenant à la catégorie Canard. Ainsi, l'apprentissage supervisé repose sur un principe de généralisation : à partir d'un ensemble d'exemples connus, le modèle apprend à prédire le résultat pour des situations nouvelles.

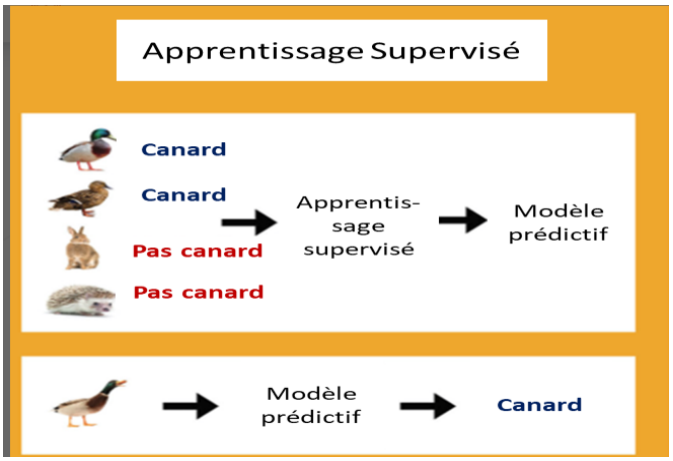


Figure 2. Apprentissage supervisé[3]

Source : Web

1.2.1.2.2. L'apprentissage non supervisé

L'apprentissage non supervisé est une méthode d'apprentissage automatique où l'algorithme travaille avec des données non annotées, c'est-à-dire des données pour lesquelles aucune informations ou étiquettes cibles ne sont fournies. L'objectif est d'identifier des patterns, des relations ou des structures cachées au sein des données sans intervention humaine ou guide externe [21, 22]. Autrement dit, c'est mettre à la disposition d'un individu un ensemble d'objets sans rien lui dire et qu'il commence par lui-même à les classés par catégories (formes, couleur, taille, etc.). Ainsi, comme technique d'apprentissage non supervisé on peut citer le **Clustering**[4] [15, 22]. L'image ci-dessous illustre le principe de l'apprentissage non supervisé de manière simple et visuelle. On y voit plusieurs animaux alignés sur la gauche (trois canards, un lapin et un hérisson). Ces données d'entrée ne sont pas accompagnées d'étiquettes ou de catégories

[3] https://www.le-cortex.com/media/articles/lintelligence-artificielle-comment-ca-marche consulté le 22/11/2024
[4] Clustering : Il vise à regrouper des données en ensemble en fonction de leur similarité.

prédéfinies, ce qui signifie que l'algorithme doit découvrir lui-même des regroupements ou structures cachées dans les données. Le schéma montre, à droite, le résultat de l'apprentissage non supervisé, où les animaux ont été regroupés en trois clusters distincts. Le premier cluster regroupe les trois canards, le deuxième contient le lapin et le troisième le hérisson.

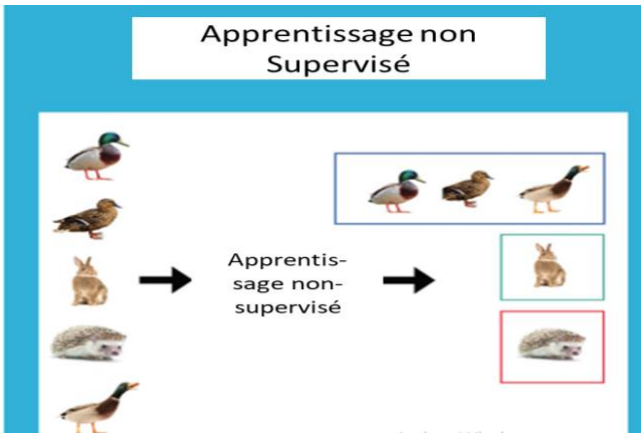


Figure 3. Apprentissage non supervisé[5]

Source : Web

1.2.1.2.3. L'apprentissage par renforcement

L'apprentissage par renforcement est une méthode d'apprentissage automatique où un agent logiciel apprend par essais et erreurs à prendre des décisions dans un environnement donné. L'objectif est de maximiser une récompense à long terme en choisissant les actions les plus avantageuses à chaque étape [21]. L'agent interagit avec son environnement en effectuant des actions. À chaque action correspond une récompense (positive ou négative) qui indique à l'agent si l'action était bonne ou mauvaise. Au fil du temps, l'agent apprend à dissocier les actions à refaire et ceux à évitées [21]. L'image qui suit montre le processus qui est une boucle fermée et constante qui commence avec le robot (l'agent), l'entité qui apprend et prend une décision. En observant l'environnement, l'agent détermine la meilleure action à exécuter, par exemple : acheter ou ne rien faire. L'exécution de cette action modifie l'environnement (la zone QR code), qui représente le monde dans lequel l'agent évolue (comme le marché des ventes). Cette modification se traduit par un nouvel état de l'environnement. L'environnement fournit alors une récompense (un signal de feedback) à l'agent, qui est positive si l'action est bonne (par exemple : une commande qui a évité une rupture de stock) ou négative si elle est mauvaise. L'agent utilise cet état et la récompense pour ajuster son modèle, c'est la stratégie interne

[5] https://www.le-cortex.com/media/articles/lintelligence-artificielle-comment-ca-marche consulté le 22/11/2024

représentée par le réseau neuronal. Ce modèle s'améliore au fil des cycles pour s'assurer que l'agent choisit l'action qui maximisera la récompense cumulée (le profit ou la performance) sur le long terme. En résumé, l'agent interagit, observe les conséquences, reçoit un score et apprend à devenir plus performant à chaque tour.

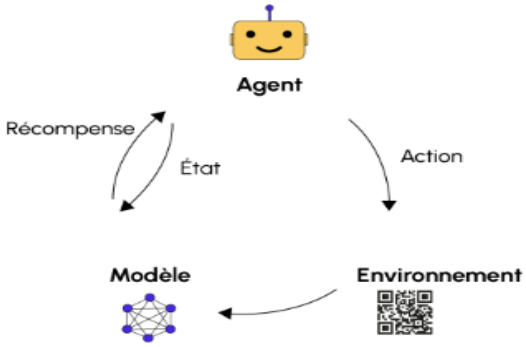


Figure 4. Apprentissage par renforcement [6]

Source : Web

1.2.1.2.4. L'apprentissage semi-supervisé

L'apprentissage semi-supervisé est une approche d'apprentissage automatique qui tire parti à la fois de données étiquetées et non étiquetées. Il offre une solution efficace lorsque l'étiquetage exhaustif des données est coûteux. En combinant les forces de l'apprentissage supervisé (qui requiert des données étiquetées) et non supervisé (qui exploite des données sans étiquettes), cette technique permet de construire des modèles plus robustes et performants, même avec des quantités limitées de données étiquetées [21]. Et comme exemple de cette méthode, on peut citer entre autres le **self-training** qui est un algorithme d'apprentissage qui commence par entraîner un modèle sur un ensemble de données initialement étiquetées. Ensuite, l'apprentissage semi-supervisé utilise ce modèle pour prédire les étiquettes des données non étiquetées. Les prédictions les plus confiantes sont ajoutées à l'ensemble d'entraînement initial, et le modèle est ré-entraîné. Ce processus se répète jusqu'à ce que le modèle obtienne des résultats satisfaisants ou qu'il atteigne sa limite d'apprentissage[7]. À l'image de la figure 5, le processus commence par un petit ensemble de données étiquetées (la banane et la pomme), dont l'identité est déjà connue par le modèle. Ces données étiquetées sont utilisées pour donner au

[6] https://datascientest.com/reinforcement-learning consulté le 22/11/2024
[7] https://www.intelligence-artificielle-school.com/ecole/technologies/apprentissage consulté le 22/11/2024

modèle une base de connaissance. L'ensemble des données d'entrée comprend : ces exemples étiquetés, mais aussi un autre groupe de données dont l'étiquette est inconnue (le raisin). Le modèle utilise ce qu'il a appris de la pomme et de la banane pour essayer de prédire l'identité du raisin. Une fois cette prédiction faite, le modèle peut potentiellement utiliser ces données non étiquetées nouvellement classifiées (même si l'étiquette n'est pas garantie exacte) pour affiner sa compréhension et devenir plus performant. Cette méthode est utilisée pour tirer parti de la grande quantité de données non étiquetées disponibles afin d'améliorer la précision et l'efficacité du modèle, tout en évitant le coût élevé de l'étiquetage manuel de la totalité des informations.

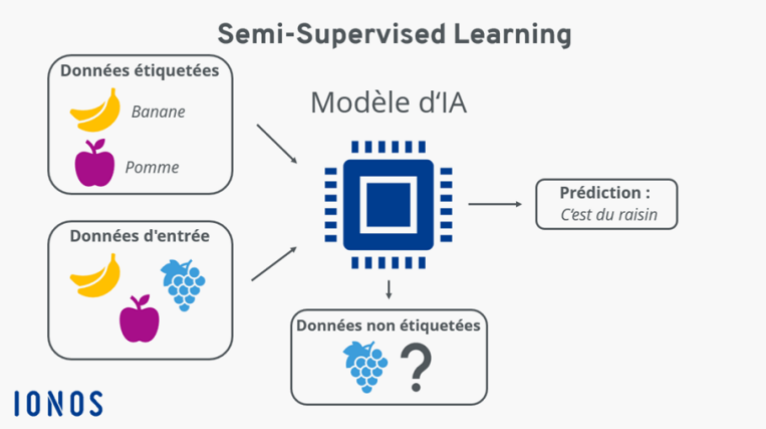


Figure 5. Le mode semi-supervisé[8]

Source : Web

1.2.1.2.5. Le Deep learning

Le deep learning, un sous domaine du machine learning est une approche qui utilise des réseaux de neurones artificiels avec de nombreuses couches pour apprendre, c'est le traitement du langage naturel. Cette technique permet de générer des algorithmes sophistiqués capables de faire des prédictions de performer et d'améliorer la productivité des entreprises [27]. Le deep learning nécessite de grandes quantités de données pour l'entraînement. Plus il y a de données disponibles, plus le modèle peut être performant [20]. Le deep learning peut demander un calcul plus important comparé à des méthodes d'apprentissage supervisé. Il est complexe et nécessite une puissance de calcul considérable. L'apprentissage profond a permis des avancées notables avec des réseaux neuronaux artificiels produisant des algorithmes supérieurs à ceux conçus par des ingénieurs humaines. Ces avancées sont appliquées dans des domaines tels que la

[8] https://www.ionos.fr/digitalguide/web-marketing/search-engine-marketing/semi-supervised-learning/ consulté le 06/11/2025

reconnaissance faciale et vocale, l'étiquetage d'images, le traitement automatisé du langage, la vision par ordinateur, la reconnaissance automatique d'images [28]. Le schéma représente l'architecture d'un Réseau Neuronal Artificiel Profond. Le réseau est composé d'une série de couches de calcul. Tout commence avec la couche Input (Entrée), qui reçoit les données brutes ou caractéristiques à analyser. Ces données voyagent ensuite à travers de multiples couches cachées (Hidden Layers), dont le grand nombre est ce qui caractérise la profondeur et la puissance du modèle. Chaque cercle dans ces couches est un neurone qui effectue un petit calcul et transmet le résultat au neurone suivant. Les flèches bleues représentent les connexions et les poids que le modèle ajuste durant son entraînement pour apprendre les motifs complexes. Ce traitement en cascade permet d'extraire des informations de plus en plus abstraites des données initiales. Enfin, l'information atteint la couche Output (Sortie), qui fournit la prédiction finale du modèle. En résumé, le réseau est un processus de calculs qui transforme progressivement les données d'entrée en une réponse utile.

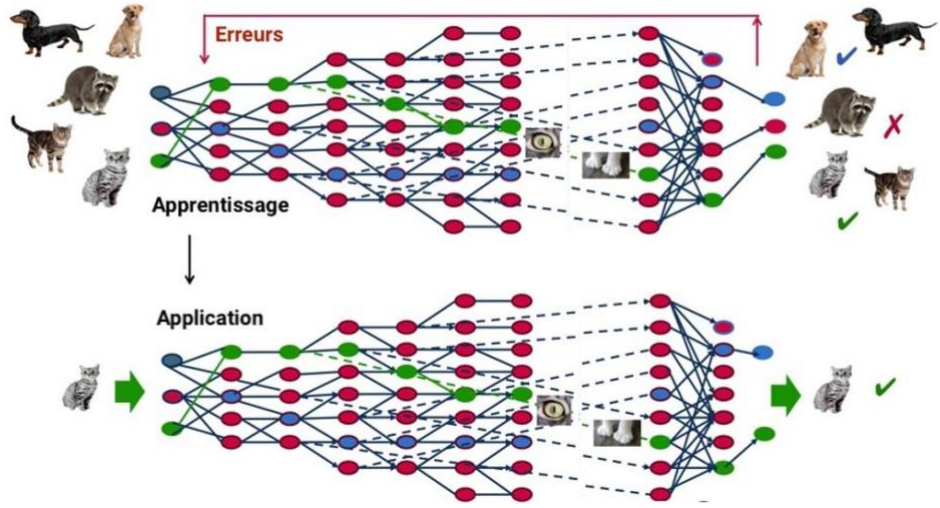


Figure 6. Reconnaissance visuelle par Deep learning[9]

Source : Web

Le tableau ci-dessous est un récapitulatif des différents types d'apprentissages, leurs descriptions, leurs atouts et limites. L'apprentissage supervisé offre une grande précision, mais repose sur la disponibilité de données préalablement étiquetées. À l'inverse, l'apprentissage non supervisé cherche à découvrir des structures ou des régularités cachées dans des données non annotées. L'approche semi-supervisée combine ces deux méthodes afin de limiter l'effort d'étiquetage tout en maintenant de bonnes performances. L'apprentissage par renforcement,

[9] https://www.futura-sciences.com/tech/definitions/intelligence-artificielle-deep-learning-17262/ consulté le 06/11/2025

quant à lui, se base sur un système d'essais-erreurs où le modèle apprend à partir de ses actions et des récompenses reçues. Le deep learning, grâce à ses réseaux neuronaux profonds, permet de traiter efficacement des données volumineuses et complexes, bien qu'il exige une puissance de calcul importante. Enfin, le transfert d'apprentissage consiste à réutiliser un modèle déjà entraîné sur une tâche similaire afin de réduire les besoins en données et d'accélérer le processus d'apprentissage [34].

Tableau de comparaison des différentes approches de machine Learning.

Source : Conception Hauteur

Type	Description	Atouts	limites
Apprentissage supervisé	Méthode qui utilise des données étiquetées pour entraîner un modèle à prédire des résultats pour des données non étiquetées.	Très précis pour les tâches spécifiques (classification, régression). Adapté pour des problèmes bien définis avec beaucoup de données étiquetées.	Dépend fortement de la qualité et de la quantité des données étiquetées. Coût élevé pour l'étiquetage manuel.
Apprentissage non supervisé	Technique qui apprend des structures cachées ou des relations dans des données non étiquetées.	Utile pour explorer des données non structurées. Permet de découvrir des motifs et des regroupements non évidents.	Ne fournit pas de sortie directement interprétable. Moins précis sans étiquettes explicites.
Apprentissage par renforcement	Modèle basé sur l'interaction entre un agent et un environnement, où l'agent apprend par essais-erreurs en recevant des récompenses ou des pénalités.	Très efficace pour des tâches séquentielles comme les jeux, la robotique ou la navigation permet d'apprendre des stratégies optimales dans des environnements dynamiques.	Longs temps d'entraînement, nécessite une forte puissance de calcul et des environnements bien simulés.
Apprentissage semi-supervisé	Combine un petit ensemble de données étiquetées avec un grand ensemble de données non étiquetées pour améliorer l'apprentissage.	Réduit le besoin de données étiquetées. Bonne performance dans les contextes où l'étiquetage est coûteux ou difficile.	Complexité accrue dans l'implémentation. Dépend de la qualité des données étiquetées disponibles.
Deep Learning ou Apprentissage Automatique	Sous-domaine du machine learning utilisant des réseaux neuronaux profonds pour modéliser des problèmes complexes, comme la vision ou le traitement du langage naturel.	Excellente performance pour les données volumineuses et complexes. Capacité à extraire des caractéristiques pertinentes sans intervention humaine.	Fortement dépendant de la quantité et de la qualité des données. Nécessite des ressources informatiques élevées et peut manquer d'explicabilité.

1.2.2. Gestion des approvisionnements

1.2.2.1. Définition

L'approvisionnement est un processus crucial pour le bon fonctionnement de toute entreprise. Il implique l'acquisition des biens et des services nécessaires à la réalisation des activités de l'entreprise, que ce soit pour la production, la distribution ou la vente. Il ne s'agit pas simplement de passer des commandes, mais d'un processus complexe qui requiert une analyse approfondie des besoins, une sélection rigoureuse des fournisseurs, une négociation des conditions et un suivi des livraisons [13]. L'objectif de l'approvisionnement est d'établir des relations durables avec les fournisseurs pour optimiser les coûts, garantir la qualité et réduire les délais de livraison.

1.2.2.2. Les aspects clés de l'approvisionnement

L'approvisionnement est un processus complexe qui s'inscrit dans un contexte plus large de gestion des flux et des opérations. Les aspects les plus importants à tenir en compte pour sa bonne gestion sont :

- L'identification des besoins : l'approvisionnement commence par une identification précise des besoins de l'entreprise en termes de biens et de services. Il s'agit de déterminer la nature, la quantité, la qualité et les délais requis pour chaque article ;
- La sélection des fournisseurs : une fois les besoins identifiés, il est important de choisir les fournisseurs les plus aptes à les satisfaire. La sélection des fournisseurs doit tenir compte de critères tels que la qualité des produits, les prix, les délais de livraison et la fiabilité du fournisseur ;
- La négociation des conditions : la négociation avec les fournisseurs est une étape importante pour obtenir les meilleures conditions possibles en termes de prix, de délais, de quantités, de qualité et de services ;
- La passation des commandes : après la négociation des conditions, les commandes sont passées aux fournisseurs. Les commandes doivent être claires et précises pour éviter tout malentendu ;
- Le suivi des livraisons : le suivi des livraisons est crucial pour s'assurer que les biens et services sont livrés dans les délais et selon les conditions convenues ;
- La réception et le contrôle des marchandises : à la réception des marchandises, il est important de vérifier la conformité des produits livrés par rapport à la demande.

1.2.2.3. Les principales méthodes d'approvisionnement

1.2.2.3.1. L'approvisionnement à la commande

C'est une méthode qui consiste à lancer les achats de matières premières ou de composants uniquement après réception d'une commande client. Autrement dit, l'entreprise ne constitue pas de stock préalable, mais adapte ses approvisionnements à la demande réelle [3]. Le gestionnaire d'approvisionnement analyse ainsi les besoins spécifiques générés par chaque commande, puis passe une commande ciblée auprès du fournisseur. Cette approche offre une grande flexibilité et permet de répondre de manière personnalisée aux besoins des clients. Elle favorise également une réduction significative des coûts de stockage, puisque les marchandises ne sont pas immobilisées inutilement dans l'entrepôt. Cependant, cette méthode présente aussi des limites importantes, notamment le risque d'allongement des délais de livraison. En effet, le client doit attendre non seulement le temps de production, mais aussi celui d'approvisionnement des matières premières. Ainsi, l'approvisionnement à la commande est particulièrement adapté aux produits à forte personnalisation ou aux demandes ponctuelles pour lesquelles il serait coûteux de maintenir un stock permanent. Lorsqu'il est bien maîtrisé et que les délais d'approvisionnement sont compatibles avec les attentes des clients, ce système se révèle efficace, économique et flexible [3].

1.2.2.3.2. Le réapprovisionnement de stock

C'est un système de gestion qui maintient un stock d'articles et le complète dès qu'un seuil est atteint. Ainsi, nous notons :

- La méthode à point de commande : cette méthode est une approche de gestion des stocks très répandue et simple à mettre en œuvre, particulièrement adaptée aux articles à faible coût et à consommation régulière. Le principe fondamental consiste à surveiller en continu le niveau de stock pour un article donné. L'action de réapprovisionnement est déclenchée dès que le niveau de stock atteint un seuil prédéfini, appelé le Point de Commande (S). Une fois ce seuil atteint, une quantité fixe (Quantité de Réapprovisionnement Q), également désignée par Taille de Lot, est commandée. Cette politique de gestion nécessite donc la définition de deux paramètres cruciaux. Le premier paramètre est le point de commande (S) qui représente la valeur de stock qui déclenche l'approvisionnement (ou la commande). Son rôle principal est d'éviter les ruptures de stock. Théoriquement, il correspond à la quantité de stock qui sera consommée pendant le délai de réapprovisionnement (le temps écoulé entre la commande et la réception). Lorsque la demande ou le délai d'approvisionnement est

incertain (aléatoire), ce seuil est généralement majoré par l'ajout d'un stock de sécurité pour garantir un niveau de service satisfaisant face aux variations imprévues. Et un deuxième paramètre qu'est la Quantité de Réapprovisionnement (Q), représente la quantité fixe qui est commandée lorsque le point de commande est atteint. Elle est généralement optimisée pour minimiser les coûts de passation de commande et les coûts de stockage (souvent calculée à l'aide du modèle de la Quantité Économique de Commande ou *Economic Order Quantity - EOQ*). En résumé, la politique (S, Q) garantit que la commande est passée juste à temps pour recevoir la livraison avant que le stock ne tombe à zéro [3, 5, 10] ;

- Le « Recomplétement » périodique : le système de recomplétement périodique, également appelé gestion calendaire, consiste à réapprovisionner les stocks à intervalles réguliers prédéfinis. Cette méthode permet une planification anticipée des commandes et une synchronisation avec les calendriers d'acheminement. Contrairement aux systèmes basés sur un stock minimum ou un seuil d'alerte, la gestion calendaire repose sur un niveau de « recomplétement » ou stock maximum que chaque commande vise à atteindre. La quantité commandée varie donc à chaque période, car elle dépend du stock réellement disponible au moment du réapprovisionnement. Ce niveau de « recomplétement » est calculé de manière à couvrir la consommation prévue durant le délai d'approvisionnement. En pratique, la période de commande souvent exprimée en jours, semaines ou mois coïncide avec les inventaires périodiques, ce qui facilite la gestion et le suivi des stocks [3, 4, 5].

1.2.2.3.3. L'approvisionnement sur prévision ou méthode MRP (Material Requirements Planning)

Il repose sur l'anticipation de la demande finale pour planifier les besoins en composants nécessaires à la production. À partir du Plan Directeur de Production (PDP), les besoins en produits finis sont établis, puis décomposés en besoins selon la nomenclature des produits. Cette approche vise à assurer une planification optimale des approvisionnements et de la production, en évitant à la fois les excès de stock et les ruptures. La méthode MRP est une méthode « push » (flux poussé), fondée sur le principe du *make to stock*, c'est-à-dire produire pour le stock avant la réception des commandes clients. Elle distingue la demande indépendante (celle des produits finis, issue des prévisions de ventes) de la demande dépendante (celle des composants nécessaires à leur fabrication). Le système fonctionne par périodes discrètes appelées *time buckets*, dans lesquelles sont programmés les ordres de fabrication (OF) et les ordres

d'approvisionnement (OA). Le MRP demeure un outil essentiel pour la planification de la production et des approvisionnements, à condition d'intégrer des paramètres tenant compte des incertitudes afin de mieux équilibrer le risque de rupture et le coût du stockage. Le processus MRP se déroule en trois étapes principales [3, 4, 10] :

- L'explosion de nomenclature : les besoins bruts en composants sont calculés à partir des prévisions de production et de la structure des produits ;
- Le calcul des besoins nets : les besoins réels sont obtenus en tenant compte des stocks disponibles et des commandes déjà en cours ;
- La planification des livraisons : les quantités et les dates de livraison sont déterminées de manière à équilibrer les coûts de stockage et de commande.

1.2.3. Gestion des stocks

1.2.3.1. Définition

La gestion des stocks désigne l'ensemble des méthodes et procédures permettant de maintenir des niveaux de stock optimaux pour chaque article, afin d'assurer la disponibilité des produits tout en maîtrisant les coûts. Elle vise à trouver un équilibre entre le coût de possession des stocks, le coût lié aux ruptures et la satisfaction des clients, mesurée par le taux de service. Cette gestion ne se limite pas au simple suivi des quantités. Elle s'inscrit dans un système plus large de Gestion des Stocks et d'Approvisionnements (GSA), qui comprend le magasinage (stockage et conservation des articles), la tenue des stocks (inventaires et classification), les approvisionnements (définition des politiques d'achat et passation des commandes) et les analyses et études permettant d'optimiser les flux. Les stocks sont constitués dès qu'un décalage existe entre la production et la demande et servent de tampon pour assurer la continuité de l'approvisionnement et de la distribution. Leur dimensionnement doit permettre d'éviter à la fois les surstocks (coûteux pour l'entreprise) et les ruptures (préjudiciables à la satisfaction client). L'état du stock peut être mesuré par le niveau de stock net, c'est-à-dire la quantité disponible après prise en compte des ruptures et par la position de stock, qui inclut les approvisionnements en cours. Enfin, la gestion des stocks repose sur des politiques définissant quand et en quelle quantité commander, assurant ainsi la coordination entre besoins opérationnels et objectifs stratégiques de l'entreprise [3, 5, 10, 13].

1.2.3.2. Les méthodes de gestion de stock

La gestion des stocks est un élément clé pour toute entreprise de distribution qui vise à optimiser les quantités de produits à stocker, afin de répondre à la demande tout en minimisant les coûts.

Ces méthodes peuvent être regroupées en deux grandes catégories : les méthodes classiques, fondées sur des principes de gestion traditionnels et déterministes, et les méthodes avancées issues de recherches récentes intégrant l'incertitude, la complexité multi-échelon et l'optimisation des décisions logistiques.

1.2.3.2.1. La méthode de la politique classique de gestion de stock

Elle constitue les fondements historiques de la gestion des stocks. Elle repose sur des modèles simples, souvent basés sur des hypothèses de demande stable et de délais constants. Ces approches visent principalement à déterminer la quantité et le moment optimal de commande à partir de paramètres économiques et opérationnels connus [3, 8, 13]. Parmi les plus utilisées figurent :

- Le modèle du point de commande (S, Q) qui consiste à lancer une commande dès que le stock atteint un seuil défini, appelé point de commande. Ce modèle permet d'éviter les ruptures tout en maintenant un stock de sécurité ;
- Le modèle de la quantité économique de commande (EOQ ou Wilson) qui détermine la quantité optimale à commander en équilibrant les coûts de possession et de passation de commande ;
- La méthode de recomplétement périodique où les commandes sont passées à intervalles réguliers, selon une quantité ajustée pour atteindre un niveau de stock cible ;
- Les méthodes de classification (ABC/XYZ) qui permettent de différencier les articles selon leur importance économique ou leur variabilité, afin d'appliquer une politique adaptée à chaque catégorie.

Ces modèles se distinguent par leur simplicité d'application et leur pertinence dans des environnements stables, mais montrent des limites face à la variabilité croissante de la demande, à la mondialisation des flux et à la complexité des réseaux logistiques [3, 13]. Ces constats ont conduit au développement de méthodes plus dynamiques et intégratives, dites avancées.

1.2.3.2.2. La méthode avancée de gestion de stock

Cette méthode s'inscrit dans une logique d'optimisation globale et de pilotage collaboratif. Elle vise à améliorer la performance de la chaîne logistique en intégrant les incertitudes, la décentralisation des décisions et la complexité des systèmes multi-échelons [2, 4, 5, 6, 10]. Elle mobilise des approches analytiques et informatiques sophistiquées, parmi lesquelles :

- La planification des besoins en composants (MRP) : cette méthode permet de déterminer les quantités et les dates de commande en fonction des prévisions de vente

et des caractéristiques des produits. Elle assure une meilleure coordination entre production et approvisionnement ;

- La gestion collaborative des approvisionnements (GPA/VMI) : dans cette approche, le fournisseur prend en charge la gestion des stocks du client sur la base d'informations partagées. Elle permet de réduire les coûts et d'améliorer le taux de service ;
- Les modèles multi-échelons : ils prennent en compte la structure hiérarchique des réseaux logistiques et les interdépendances entre les différents niveaux (fournisseurs, entrepôts, distributeurs). Ces modèles visent à optimiser les stocks à chaque échelon pour améliorer la performance globale [2, 5] ;
- Les modèles stochastiques : ils permettent de gérer les incertitudes de la demande et des délais d'approvisionnement. Ces modèles, tels que le Newsboy ou les approches de programmation probabiliste, aident à définir des niveaux de stock de sécurité adaptés aux aléas du système [2, 10] ;
- Les approches algorithmiques et heuristiques : elles sont utilisées pour résoudre les problèmes complexes de gestion des stocks lorsque les méthodes exactes sont inapplicables. Ces algorithmes d'approximation offrent des solutions proches de l'optimum dans les contextes industriels réels [8] ;
- Les politiques en boucle fermée : elles intègrent la récupération, la réparation et le recyclage des produits dans la planification des stocks, afin d'optimiser les flux directs et inverses dans une perspective de durabilité [5] ;
- Les approches Lean et Kanban : elles reposent sur le pilotage par la demande et l'amélioration continue des flux. Le système Kanban, par exemple, déclenche automatiquement les réapprovisionnements à partir de signaux visuels, réduisant ainsi les gaspillages et les excès de stock [3, 6, 13].

Ces méthodes avancées favorisent une gestion proactive, flexible et collaborative, capable de répondre aux incertitudes du marché et aux exigences croissantes de performance logistique.

Synthèse comparative des méthodes de gestion de stock

Source : Conception Hauteur

Catégories	Caractéristiques principales	Objectifs dominants
Méthodes classiques	Modèles déterministes, demande stable, simplicité de mise en œuvre (EOQ, point de commande, périodique, ABC).	Minimiser les coûts et simplifier la gestion.
Méthodes avancées	Modèles stochastiques, multi-échelons, collaboratifs et orientés optimisation (MRP, GPA, Kanban, modèles probabilistes, boucles fermées).	Optimiser la performance globale et renforcer la résilience du système.

1.2.3.3. Le stock et ces types

Le stock est un ensemble de produits ou d'éléments disponibles mise de côté pour faire face à une demande future. Il permet de pallier les imprévus, comme une augmentation soudaine de la demande ou un retard de livraison. En ayant des produits en réserve, on s'assure de pouvoir répondre aux besoins des clients en temps et en heure. De plus, les stocks permettent d'optimiser la production en lissant les pics de demande et en profitant des économies d'échelle lors des achats en grande quantité [4]. Les stocks peuvent être classés selon leur rôle fonctionnel au sein du processus logistique et de production. Cette typologie, largement abordée dans la littérature en gestion industrielle et logistique, permet de mieux comprendre la nature, la fonction et l'importance stratégique de chaque catégorie de stock au sein de la chaîne d'approvisionnement [3, 8, 13]. Ainsi, on distingue généralement les catégories suivantes :

- Les stocks de matières premières : ils regroupent l'ensemble des composants, pièces, produits bruts ou matériaux nécessaires à la fabrication des produits finis. Ces stocks assurent la continuité du processus de production en évitant les interruptions liées à des retards d'approvisionnement ou à des fluctuations de la demande en intrants. Ils jouent un rôle stratégique dans la stabilité de la production et la flexibilité de la chaîne logistique, notamment dans les environnements où les délais d'approvisionnement sont longs ou incertains [3,13] ;
- Les stocks en cours de fabrication (ou semi-finis) : il s'agit des produits partiellement transformés se trouvant entre les différentes étapes du processus de production. Ces stocks représentent un tampon entre les postes de travail et permettent d'absorber les déséquilibres ou retards dans la chaîne de production. Ils sont essentiels pour maintenir la fluidité des flux internes et réduire les temps d'attente entre les opérations, notamment dans les systèmes de production multi-échelons [4, 5] ;

- Les stocks de produits finis : ils désignent les produits complètement fabriqués, conditionnés et prêts à être livrés aux clients ou aux distributeurs. Ces stocks sont au cœur de la performance commerciale de l'entreprise, car ils permettent de répondre rapidement aux besoins des clients et de maintenir un niveau de service élevé. Leur dimensionnement optimal est crucial pour éviter à la fois le sur stockage qui immobilise des capitaux et les ruptures, sources d'insatisfaction client [3, 13] ;
- Les stocks de sécurité : ce sont des réserves constituées pour faire face aux aléas de la demande et aux incertitudes des délais d'approvisionnement. Ils représentent une marge de précaution permettant d'assurer la continuité des activités malgré les imprévus. Ces stocks jouent un rôle essentiel dans les systèmes logistiques soumis à une forte variabilité et leur dimensionnement fait appel à des modèles probabilistes avancés permettant d'équilibrer coût et fiabilité [2, 6] ;
- Les stocks de transit : ils correspondent aux produits en cours de déplacement entre différents sites logistiques (entrepôts, usines, points de distribution ou magasins). Ces stocks existent en raison des délais de transport et du temps nécessaire pour que les marchandises passent d'un maillon de la chaîne à un autre. La prise en compte des stocks de transit est primordiale dans les systèmes multi-échelons ou internationaux, où les flux physiques sont étendus et complexes [8].

1.2.4. La prédiction

1.2.4.1. Définition

La prédiction est un processus complexe qui vise à estimer les événements futurs en s'appuyant sur l'analyse des données passées et la compréhension des tendances actuelles. Elle est essentielle pour une gestion efficace des opérations [9]. Elle permet une meilleure planification et anticipation des événements à venir. Mais il est important de garder à l'esprit son caractère incertain et de prendre des mesures pour atténuer les risques associés. Ainsi, l'utilisation du machine learning permettra après une analyse d'une grande quantité de données d'identifier les patterns complexes et donc d'améliorer la précision des prévisions.

1.2.4.2. Les méthodes de prédiction utilisées pour anticiper les niveaux de stock

La prédiction des niveaux de stock consiste à estimer, pour chaque période et pour chaque emplacement de la chaîne (usine, entrepôt, point de vente), la quantité de stock nécessaire pour satisfaire la demande tout en limitant les coûts (stockage, rupture, commandes) [6]. La qualité de ces prédictions conditionne directement les politiques de réapprovisionnement, le dimensionnement des stocks de sécurité et la performance globale de la chaîne logistique [6,

7]. L'objectif principal est de déterminer des cibles de stock (stock de cycle, stock de sécurité, stock saisonnier) qui garantissent un taux de service donné tout en maîtrisant le coût total [6, 13]. Les enjeux opérationnels sont d'intégrer l'incertitude (demande et délais), la structure multi-échelon du réseau et la contrainte de calcul pour les grands réseaux industriels [2, 8]. Ces enjeux expliquent pourquoi la prédiction des niveaux de stock associe des méthodes de prédiction de la demande et des modèles d'optimisation ou heuristiques pour traduire ces prédictions en niveaux de stock actionnables [6, 8]. Les méthodes de prédiction utilisées pour anticiper les niveaux de stock se répartissent en plusieurs catégories, chacune répondant à des contextes spécifiques liés à la stabilité de la demande, à la qualité des données ou à la complexité des phénomènes à modéliser. Elles s'étendent des approches statistiques traditionnelles aux modèles hybrides intégrant des techniques avancées d'apprentissage automatique :

- Séries temporelles classiques : les méthodes de prédiction tel que les moyennes mobiles, les lissages exponentiels (simple, Holt, Holt–Winters) et le modèles ARIMA sont des outils de base pour des demandes relativement stables et bien historiées. Ils servent de base pour dimensionner le stock de cycle [6, 9] ;
- Méthodes adaptatives : ce sont des algorithmes de lissage adaptatif qui ajustent automatiquement les paramètres selon l'évolution récente de la série (réduction du biais en période de rupture de tendance). Ces approches historiques (ex. méthodes proposées dès les années 60) restent pertinentes pour les prédictions à court terme en stock management [9] ;
- Méthodes de régressions : c'est quand des variables explicatives (promotions, météo, prix) influencent la demande. Dans ce cas la régression multiple permet d'affiner la prédiction et d'anticiper des pics ou creux [7] ;
- Méthodes avancées et hybrides : c'est une combinaison de plusieurs méthodes pour capter les non-linéarités, les saisonnalités complexes et les interactions. Ces modèles améliorent la précision de la prédiction quand les données sont abondantes et bruitées [7, 2] ;
- Approches processuelles et organisationnelles : c'est la formalisation et l'automatisation du processus de prédiction (collecte, nettoyage, arbitrage entre prédictions et commandes clients) afin d'améliorer la qualité des entrées pour le calcul des stocks. Ces aspects méthodologiques sont centraux dans les travaux récents sur la modélisation du processus de prédiction [7].

Pour chacun des outils de prédiction, la précision de ce dernier et la régularité des données déterminent la pertinence. Les méthodes quantitatives exigent des historiques fiables tandis que les méthodes qualitatives demeurent utiles en cas de nouveauté ou de rupture de série [7, 9].

1.2.4.3. Traduction des prévisions de niveaux de stock

Pour utiliser les prédictions de demande dans la gestion des stocks, il faut les transformer en niveaux de stock adaptés. Cette étape permet de déterminer combien stocker, comment se protéger contre les imprévus et comment assurer la disponibilité des produits [6, 10].

- Stock de cycle : calculé à partir de la demande moyenne sur la période de commandes (ou lot économique), il sert à couvrir la consommation normale [6] ;
- Stock de sécurité : dimensionné en fonction de la variance de la demande (ou des délais) et du niveau de service visé, les modèles stochastiques et la programmation stochastique multi-échelon permettent d'ajuster les stocks de sécurité selon la position dans le réseau et les corrélations d'aléas [2, 10] ;
- Intégration MRP / PDP : pour les environnements manufacturiers, la prédiction des ventes est éclatée en besoins nets de composants via le MRP. La précision des prédictions impacte directement les niveaux de stock à chaque nœud [6] ;
- Prise en compte des délais longs (approvisionnement lointain) : lorsqu'un approvisionnement est lointain, l'erreur de prédiction croît avec l'horizon, il faut alors augmenter la prudence (stocks de sécurité plus élevés) ou revoir la stratégie (regroupement, commandes anticipées). L'ajustement dynamique des stocks de sécurité en fonction de l'erreur de prévision est une recommandation opérationnelle issue des travaux sur approvisionnement lointain [2, 6, 10].

1.2.4.4. Spécialité multi-échelon et solution algorithmique

Dans les chaînes d'approvisionnement étendues, plusieurs méthodes avancées permettent d'optimiser les stocks à l'échelle de tout le réseau, d'améliorer la disponibilité des produits et de réduire les coûts globaux. Ainsi nous en citons :

- L'optimisation multi-échelon : elle consiste à optimiser les stocks de tous les sites en même temps (entrepôts, magasins, dépôts). Au lieu de gérer chaque site séparément, on les coordonne, cela réduit le stock total nécessaire tout en maintenant un bon niveau de service. Exemple : un stock central plus élevé permet de diminuer les stocks des magasins tout en évitant les ruptures [2] ;

- Les algorithmes d'approximation et heuristiques : pour les grands réseaux où la résolution exacte est intraitable, des algorithmes d'approximation ou heuristiques garantissent des solutions de bonne qualité avec un coût de calcul raisonnable. Quand le réseau est trop grand pour être optimisé exactement (calcul trop long), on utilise des méthodes plus rapides qui donnent de bonnes solutions sans être parfaites. Exemple : utiliser une heuristique pour décider des niveaux de stock dans 50 entrepôts au lieu de résoudre une équation complexe impossible à calculer [8] ;
- Gestion des transferts latéraux (transshipment) : cette technique consiste à transférer des produits entre sites d'un même réseau pour éviter les ruptures. Exemple : si un magasin manque d'un produit mais un autre en a trop, on rééquilibre les stocks en transférant rapidement l'excédent [11].

1.2.4.5. Bonnes pratiques opérationnelles

Dans les chaînes d'approvisionnement, il y'a des approches qui offrent un cadre analytique et opérationnel permettant de mieux coordonner les différents sites, de maîtriser l'incertitude et de soutenir des décisions d'approvisionnement plus efficaces. Ainsi il est bien de :

- Piloter le processus de prédiction (rôles, horizon, revue périodique) pour garantir la qualité et la traçabilité des données servant à dimensionner les stocks [7] ;
- Choisir la méthode de prédiction en fonction du contexte (données historiques, saisonnalité, impact d'événements externes) et combiner les méthodes quantitatives et jugements d'experts que quand nécessaire [7, 9] ;
- Intégrer l'incertitude et la structure réseau : utiliser des modèles stochastiques et optimisation multi-échelon pour définir les stocks de sécurité adaptés à la position dans la chaîne [2] ;
- Employer des algorithmes d'approximation pour les grands problèmes afin d'obtenir des solutions utilisables en temps utile [8] ;
- Mettre en place une boucle d'amélioration continue : mesurer l'erreur de prédiction, ajuster les modèles et recalculer périodiquement les paramètres (seuils, tailles de lots, stock de sécurité) [3, 13].

1.3. Revue de la littérature théorique

1.3.1. Les théories sur la gestion des stocks

Au fil du temps, les théories de la gestion des stocks sont passées d'une logique de simple entreposage à une approche plus stratégique visant à optimiser les niveaux de stock, réduire les coûts et assurer un approvisionnement régulier de l'entreprise.

1.3.1.1. Théorie de la Quantité Économique de Commande (EOQ)

Le modèle de Wilson, également connu sous le nom de Quantité Économique de Commande (EOQ), est une théorie fondamentale de la gestion des stocks qui vise à déterminer l'équilibre optimal entre différents coûts pour minimiser la dépense totale. La théorie de Wilson repose sur la recherche d'un équilibre entre deux types de charges [42]:

- Les coûts de passation (ou d'acquisition) : Ce sont les frais fixes liés au traitement administratif et logistique de chaque commande, quel que soit le volume ;
- Les coûts de possession (ou de stockage) : Ce sont les frais liés à l'immobilisation financière et au magasinage des articles en stock (loyer, assurance, détérioration).

Le modèle démontre que la quantité optimale à commander est celle qui minimise la somme de ces deux coûts.

1.3.1.2. Méthode Juste-à-Temps (Just In Time) / Le Toyotisme

La théorie du Juste-à-Temps (JAT), du Système de Production Toyota (SPT) ou Toyotisme, a été développée par OHNO pour répondre aux besoins spécifiques de l'industrie japonaise après la seconde guerre mondiale. Contrairement au modèle de production de masse de Ford, le JAT vise l'efficacité par l'élimination absolue du gaspillage [43]. Le Juste-à-Temps signifie que, dans un flux de production, les pièces nécessaires arrivent à la ligne d'assemblage au moment où elles sont nécessaires et seulement dans la quantité nécessaire. Il gère le flux pour que chaque processus reçoive exactement ce dont il a besoin. L'objectif est d'approcher le stock à zéro, car le SPT considère les stocks non comme des actifs, mais comme une collection de problèmes et de coûts cachés (surproduction, défauts, personnel excédentaire) [43].

1.3.1.3. La Théorie des Contraintes (TOC)

La Théorie des Contraintes (TOC), conceptualisée par Eliyahu M. Goldratt, redéfinit la gestion des stocks en la déplaçant du « monde des coûts » vers le « monde du débit » (throughput). Contrairement à la comptabilité traditionnelle qui considère les stocks comme un actif, la TOC les définit comme un passif ou une dette, car ils représentent de l'argent coincé à l'intérieur du

système qui n'a pas encore été converti en profit. La TOC s'appuie sur trois indicateurs clés pour évaluer si une décision (comme l'achat de stock) rapproche l'entreprise de son but (gagner de l'argent). Voici les principes fondamentaux de la TOC appliqués à la gestion des stocks :

- Le Débit (Throughput) : le rythme auquel le système génère de l'argent par les ventes (et non par la production seule) ;
- Les Stocks (Inventory) : tout l'argent que le système a investi dans l'achat de choses qu'il a l'intention de vendre. Cela inclut les matières premières, les en cours (WIP) et les produits finis ;
- Les Dépenses opérationnelles (Operating Expense) : tout l'argent que le système dépense pour transformer les stocks en débit.

L'objectif stratégique est d'augmenter le débit tout en réduisant simultanément les stocks et les dépenses opérationnelles.

1.3.1.4. La théorie de l'effet coup de fouet « Bullwhip Effect »

L'effet coup de fouet (Bullwhip Effect) désigne le phénomène par lequel de faibles variations de la demande des clients finaux entraînent des fluctuations de plus en plus importantes des commandes passées aux fournisseurs. À mesure que l'on remonte la chaîne logistique, ces variations s'amplifient, ce qui peut provoquer des déséquilibres importants dans la production, les stocks et les approvisionnements [44]. Pour atténuer cet effet, les auteurs préconisent une meilleure coordination et transparence :

- Partage des données : donner au fabricant l'accès direct aux données de vente au détail (données POS ou sell-through) pour éviter les doubles prévisions ;
- Gestion centralisée : utiliser des systèmes de VMI (Vendor-Managed Inventory) où le fournisseur gère lui-même les stocks du détaillant ;
- Réduction des délais : raccourcir le temps de cycle de commande et de livraison pour stabiliser les besoins ;
- Stabilité des prix : adopter des stratégies de type EDLP (Every Day Low Price) pour lisser la demande ;
- Politiques d'allocation : rationner en fonction des ventes passées plutôt que des commandes actuelles pour décourager les commandes gonflées.

1.3.2. La théorie sur l'adoption des innovations

L'étude de l'adoption de l'innovation est un domaine de recherche mature qui s'appuie sur plusieurs modèles théoriques issus de la sociologie, de la psychologie et des systèmes d'information.

1.3.2.1. La Théorie de la Diffusion des Innovations (Everett Rogers)

Développée par Everett ROGERS, cette théorie définit la diffusion comme le processus par lequel une innovation est communiquée via certains canaux, au fil du temps, parmi les membres d'un système social. Elle repose sur quatre éléments fondamentaux : l'innovation, les canaux de communication, le temps et le système social [45]. Le taux d'adoption d'une idée nouvelle dépend de la perception qu'en ont les potentiels adoptants selon cinq attributs clés :

- Avantage relatif : le degré auquel une innovation est perçue comme étant meilleure que l'idée qu'elle remplace ;
- Compatibilité : la perception que l'innovation est cohérente avec les valeurs existantes, les expériences passées et les besoins des adoptants ;
- Complexité : le degré de difficulté perçu pour comprendre et utiliser l'innovation ;
- Possibilité d'essai : La capacité d'expérimenter l'innovation sur une base limitée avant de s'engager ;
- Observabilité : La visibilité des résultats de l'innovation pour les autres membres du système.

Les membres d'un système social ne sont pas tous égaux face au changement et sont classés selon leur caractère innovateur :

- Innovateurs (2,5 %) : audacieux, ils cherchent activement de nouvelles idées et acceptent les risques ;
- Adoptants précoces (13,5 %) : Leaders d'opinion respectés, ils servent de modèles pour le reste du système ;
- Majorité précoce (34 %) : délibérés, ils adoptent juste avant la moyenne ;
- Majorité tardive (34 %) : sceptiques, ils n'adoptent que par nécessité économique ou pression sociale ;
- Retardataires (16 %) : traditionnels, leur point de référence est le passé.

1.3.2.2. Le Modèle d'Acceptation de la Technologie (TAM)

Proposé par Fred Davis en 1989, le TAM (Technology Acceptance Model) se concentre sur l'acceptation des systèmes d'information en milieu organisationnel. Ce modèle stipule que deux croyances spécifiques sont les déterminants primordiaux de l'utilisation d'un système [46] :

- Utilité perçue : le degré auquel une personne croit que l'utilisation d'un système améliorerait sa performance au travail ;
- Facilité d'utilisation perçue : le degré auquel une personne croit que l'utilisation d'un système ne nécessiterait pas d'effort excessif. Cette notion est étroitement liée à la « complexité » définie par Rogers.

1.3.2.3. La Théorie Unifiée (UTAUT)

En 2003, Venkatesh et ses collègues ont synthétisé huit modèles dominants (dont la TRA, le TAM et l'IDT) pour créer l'UTAUT (Unified Theory of Acceptance and Use of Technology) [47]. Ce modèle identifie quatre déterminants directs de l'intention et de l'utilisation :

- Attente de performance : similaire à l'utilité perçue ;
- Attente d'effort : similaire à la facilité d'utilisation ;
- Influence sociale : le degré auquel un individu perçoit que des personnes importantes croient qu'il devrait utiliser le système ;
- Conditions facilitatrices : la croyance qu'une infrastructure organisationnelle et technique existe pour soutenir l'utilisation.

L'UTAUT introduit également quatre modérateurs qui influencent ces relations : le genre, l'âge, l'expérience et le caractère volontaire ou obligatoire de l'utilisation.

1.4. Conclusion

Au terme de ce premier chapitre consacré à la clarification des mots clés et à l'élaboration du cahier des charges, il apparaît que la compréhension précise des concepts fondamentaux constitue un préalable indispensable à toute démarche pour nos prédictions des niveaux de stock basées sur le ML. L'analyse des notions d'optimisation des approvisionnements, de gestion des stocks, ainsi que l'identification des différents types et niveaux de stock, permet de situer clairement les enjeux opérationnels auxquels répondra le modèle de prédiction.

Chapitre 2 : État de l'art, Positionnement

2.1. Introduction

La transformation numérique, portée par le machine learning, offre de nouvelles perspectives pour optimiser la gestion des stocks. En exploitant les capacités prédictives de l'intelligence artificielle, les entreprises peuvent affiner leurs prédictions de demande, réduire les ruptures de stock et le sur stockage et ainsi maîtriser leurs coûts logistiques. De plus, le machine learning permet une meilleure adaptation aux évolutions du marché et une prise de décision plus rapide. Notre état de l'art sur l'application du machine learning à la gestion des stocks va exposer plusieurs axes de recherche, chacun visant à améliorer les aspects de la chaîne d'approvisionnement. L'exploration bibliographique a révélé une grande diversité de modèles utilisés pour la prédiction, reflétant une évolution constante des techniques, des approches classiques de régression jusqu'aux réseaux de neuronaux avancés. Cette revue de la littérature structure l'ensemble des prédictions faites par machine learning afin d'étudier de manière comparative leurs fondements, leurs performances et leurs limites. L'objectif ultime de cette analyse est de sélectionner le modèle le plus performant et le mieux adapté aux caractéristiques de nos données (linéarité, saisonnalité etc…), pour ensuite l'adapter spécifiquement à la prédiction précise des niveaux de stock. Cette démarche méthodologique garantira que la solution technique retenue est la plus robuste pour une optimisation efficace de nos approvisionnements. Ainsi nous allons les présentés ci-dessous avec leurs caractéristiques.

2.2. État de l'art

La gestion des stocks est un élément stratégique de la chaîne d'approvisionnement, permettant de réduire les ruptures, limiter les surstocks**,** améliorer le service client et optimiser les coûts logistiques. La transformation numérique, notamment via le machine learning, a ouvert de nouvelles perspectives pour la prédiction des niveaux de stock et l'optimisation des approvisionnements [27, 33].

2.2.1. Les prédictions par machine learning

2.2.1.1. Prédictions faites par Apprentissage supervisé

L'apprentissage supervisé est largement utilisé pour la prédiction des niveaux de stock, car il repose sur des données historiques étiquetées, permettant au modèle d'apprendre la relation entre les variables explicatives et les niveaux de stock futurs. Une étude menée sur les magasins Walmart [23] avait pour objectif de prédire les ventes hebdomadaires afin d'optimiser les niveaux de stock. Les chercheurs ont comparé trois modèles de régression : Arbre de décision, Random Forest et K Neighbors Regressor. Les résultats ont montré que le modèle Random Forest offrait les meilleures performances avec un coefficient de détermination **R^2 = 0,937**, une

erreur absolue moyenne **MAE = 1937,81** et une erreur quadratique moyenne **MSE = 32 993 323,63**. Cette approche s'est avérée robuste pour capturer les tendances générales des ventes. Cependant, elle présente certaines limites, notamment l'absence d'intégration de variables externes telles que la météo, les promotions ou les événements économiques, qui peuvent influencer significativement la demande.

	R^2	MAE	MSE
Decision Tree Regressor	0.905	2377.970	49832386.628
Random Forest Regressor	0.937	1937.810	32993323.634
K Neighbors Regressor	0.594	8199.393	213246328.556

Figure 7. Résultat des 3 modèles de régression

Source : Littérature [23]

Un autre exemple d'application de l'apprentissage supervisé dans la gestion des stocks concerne la planification des stocks de sécurité [30]**.** Dans ce contexte, la demande dépend de plusieurs variables exogènes comme les prix, la météo et la saisonnalité. Deux méthodes ont été explorées que sont l'Ordinary Least Squares (OLS) et la Programmation Linéaire (PL) [30]. L'OLS s'est révélé efficace lorsque les échantillons de données sont relativement petits et lorsque l'objectif est d'atteindre un niveau de service précis, par exemple éviter les ruptures de stock. En revanche, la PL offre une solution plus robuste dans des contextes multi-variables complexes, permettant d'optimiser les niveaux de stock en tenant compte de contraintes multiples. Ces travaux démontrent que l'apprentissage supervisé classique peut fournir des prédictions fiables, mais que son efficacité dépend fortement de la qualité et de la diversité des données utilisées.

2.2.1.2. Prédiction par Deep Learning

Pour dépasser les limites des modèles supervisés classiques, le Deep Learning a été appliqué à la prédiction des ventes et des stocks, en particulier pour gérer des relations complexes et non linéaires dans les séries temporelles. Dans l'étude [24], un modèle LSTM multi-varié a été utilisé pour prédire les ventes futures. Contrairement aux modèles uni-variés, ce modèle intègre plusieurs variables explicatives telles que les prix, les promotions ou les conditions économiques, permettant ainsi de capturer des dépendances temporelles longues. Les résultats ont montré que le modèle LSTM multi-varié atteint une grande précision, ce qui en fait un outil efficace pour la planification des stocks et la prédiction de la demande. Cependant, cette

méthode requiert de grandes quantités de données pour l'entraînement et sa capacité de généralisation à de nouveaux produits ou contextes peut être limitée.

Store	Store type	LSTM - Initial	LSTM - Improved
749	a	627.03	494.44
85	b	617.93	683.38
519	c	826.60	732.08
725	d	584.66	541.01
165	a	342.22	347.57
335	b	1,772.99	949.28
925	c	1,065.23	986.86
1089	d	1,161.25	816.24

Figure 8. Comparaison entre les résultats du LSTM simple et multi-varié

Source : Littérature [24]

Dans 6 magasins sur 8, le LSTM amélioré obtient une erreur plus faible que le LSTM initial. Cela montre que les modifications apportées au modèle ont permis d'améliorer sa capacité de prédiction dans la majorité des magasins. En revanche, pour les magasins 85 et 165, le modèle initial reste légèrement plus performant, ce qui suggère que les améliorations ne profitent pas de manière uniforme à tous les contextes ou tous types de magasins.

Une approche plus avancée a été développée [31], combinant le modèle GRU (Gated Recurrent Unit) avec un algorithme d'optimisation méta-heuristique[10] IASO pour améliorer la précision des prédictions dans des contextes sensibles à la demande. Ce modèle a permis d'optimiser automatiquement les paramètres du réseau GRU et d'obtenir des performances très élevées : **R^2 = 0,97**, **MAE = 0,90**, **RMSE = 1,20**, **MSE = 1,44** et **MAPE = 6,3%**. L'ingénierie des caractéristiques et l'optimisation des paramètres ont contribué à ces résultats, soulignant l'efficacité de cette approche pour la prédiction fine des niveaux de stock. Néanmoins, le GRU optimisé par IASO reste computationnellement exigeant, ce qui peut poser des contraintes pour les entreprises disposant de ressources limitées.

[10] Méta-heuristique : un terme donné à une classe générale d'algorithme utilisée pour trouver des solutions aux problèmes d'optimisation lorsque les techniques exactes s'avèrent insuffisantes. **Source** https://www.planilog.com/support/metaheuristique/#:~:text=La%20m%C3%A9taheuristique%20est%20un%20terme,%E2%80%93%20etc. Consulté le 06/12/2025

Model	R^2	RMSE	MAE	MSE
Baseline GRU/IASO	0.95	1.38	1.05	1.90
GRU/IASO with Feature Engineering	0.97	1.20	0.90	1.44

Figure 9. Résultat des performances du modèle GRU IASO

Source : Littérature [31]

2.2.1.3. Prédiction par Méthodes hybrides et comparaison

Les méthodes hybrides combinent plusieurs modèles pour tirer parti de leurs forces respectives. Dans le domaine de l'e-commerce et pour la référence [26], les auteurs ont proposé un modèle hybride associant ARIMA pour modéliser la composante linéaire des séries temporelles et LSTM pour prédire les résidus non linéaires. Cette approche permet de capturer simultanément les tendances globales et les fluctuations complexes des ventes, offrant une amélioration significative de la précision des prédictions.

Les résultats ont montré une réduction du RMSE par rapport aux modèles classiques, ce qui en fait une solution efficace pour les produits saisonniers et fortement sensibles aux variations du marché.

	MSE	RMSE	MAE
LSTM	540.76758	13.2629	9.68830
ARIMA	466.05542	12.2340	9.21864
Baseline (Zhang)	438.51756	11.7378	8.76454
Proposed Methodology	415.44138	11.6794	8.88266
Proposed Methodology with Retail Price	412.74034	11.5222	8.73078

Figure 10. Résultat des modèles LSTM, ARIMA

Source : Littérature [26]

Ce tableau montre que l'ajout du prix de vente (« Retail Price ») comme variable explicative améliore les performances du modèle proposé. En effet, cette version présente les erreurs les plus faibles sur l'ensemble des métriques, ce qui indique une meilleure qualité des prédictions.

Cela suggère que le prix de vente apporte une information pertinente pour expliquer les variations de la demande et améliorer la précision des prédictions.

Par ailleurs, dans les Smart Grids, les auteurs [29] montrent que les modèles CRBM et FCRBM offrent des performances nettement supérieures aux approches classiques comme les ANN, HMM ou SVM pour la prédiction de séries temporelles énergétiques.

Le CRBM se distingue par sa capacité à modéliser efficacement la dynamique non linéaire des consommations à court terme, tandis que le FCRBM, intégrant des variables externes telles que les prix ou la météo, améliore significativement la précision sur des horizons de prédiction plus longs. Les auteurs concluent que le FCRBM est le modèle le plus robuste et le plus performant dans le contexte des Smart Grids, avec des erreurs de prédiction globalement beaucoup plus faibles que celles des modèles traditionnels et une meilleure stabilité face à la variabilité temporelle [29].

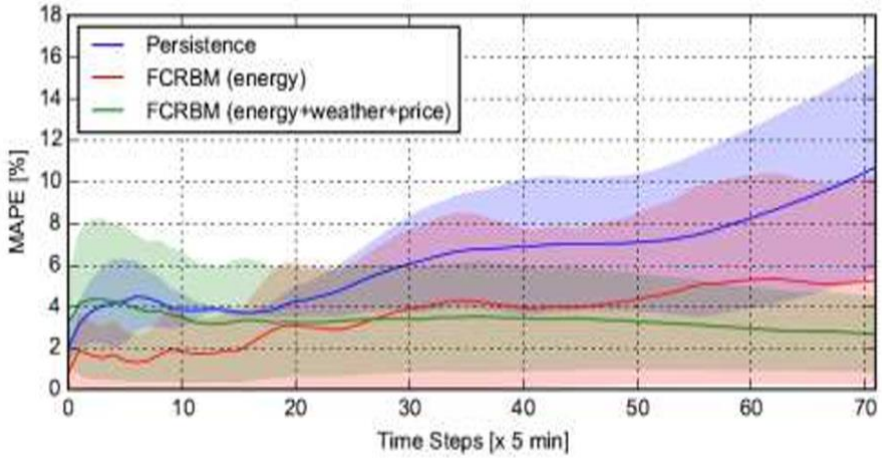


Figure 11. Exemple de prédiction de la demande électrique en fonction du MAPE

Source : Littérature [30]

Et voici une comparaison des approches d'apprentissage supervisé et non supervisé appliquées à la prédiction des prix des actions, en évaluant plusieurs modèles tels que la régression linéaire, le KNN, le SVM, les ANN et les LSTM [25]. Les résultats montrent que les modèles LSTM offrent les meilleures performances prédictives, surpassant nettement les méthodes classiques grâce à leur capacité à capturer les dépendances temporelles complexes.

Toutefois, malgré leur efficacité, les modèles d'apprentissage profond restent complexes, nécessitent de grandes quantités de données et présentent une faible interprétation, ce qui peut limiter leur généralisation à certains domaines ou types de données.

2.2.2. Synthèse de leurs Applications et études de cas

Tableau synthétique comparatif ***Source*** *: Conception hauteur*

Modèle	Type d'apprentissage	Avantages	Limites	Applications
Arbre de décision	Supervisé	Il est simple à comprendre, rapide à mettre en œuvre et facilement interprétable.	Il est sensible au surapprentissage, surtout lorsque les données sont bruitées.	Appliqué aux prévisions simples et aux analyses exploratoires de base.
Random Forest	Supervisé	Il est robuste et permet d'améliorer la précision en combinant plusieurs arbres.	Il prend difficilement en compte certaines variables externes complexes sans ajustements spécifiques.	Il est utilisé dans le secteur du retail pour améliorer la prédiction des ventes et des stocks.
KNN	Supervisé	Il est simple à utiliser et s'adapte facilement aux nouvelles données.	Il ne capture pas efficacement les tendances globales des séries temporelles.	Il est utilisé pour les prédictions à court terme sur des données peu complexes.
OLS	Supervisé	Il est simple, rapide et offre une bonne interprétation des relations entre variables.	Il devient peu performant lorsque les données impliquent plusieurs variables complexes ou non linéaires.	Il est utilisé pour estimer les stocks de sécurité dans des contextes simples.
PL	Supervisé	Il est robuste et permet de gérer plusieurs variables explicatives.	Il nécessite une phase d'optimisation pour obtenir de bonnes performances.	Il est appliqué à la gestion de stocks impliquant plusieurs facteurs explicatifs.
LSTM	DL	Il est capable de capturer les dépendances à long terme dans les données temporelles.	Il nécessite une grande quantité de données pour être efficace.	Il est utilisé pour la gestion de stocks multi-produits et les séries temporelles complexes.
GRU et IASO	DL	Ils optimisent les paramètres et améliorent l'apprentissage sur les données séquentielles.	Leur mise en œuvre est complexe et demande une expertise technique.	Ils sont utilisés pour les stocks sensibles nécessitant une grande précision.
ARIMA et LSTM	Hybride	Ce modèle combine les relations linéaires et non linéaires pour améliorer la précision.	Il est moins performant lorsque les données disponibles sont insuffisantes.	Il est utilisé dans l'e-commerce pour prévoir la demande et optimiser les stocks.
CRBM	Hybride	Il permet des prédictions en temps réel avec une bonne gestion des données temporelles	Sa mise en œuvre est complexe et nécessite des connaissances avancées.	Il est utilisé dans les Smart Grids et les systèmes nécessitant des prédictions en temps réel.

L'analyse des différentes approches de prédiction des niveaux de stock montre une large variété de modèles et de techniques, allant des modèles supervisés classiques aux méthodes hybrides combinant deep learning et statistiques. Chaque approche présente des avantages spécifiques, mais aussi des limites selon les contextes d'application.

Les modèles supervisés reposent sur des données historiques étiquetées pour apprendre la relation entre les variables explicatives et les niveaux de stock futurs. Parmi les modèles classiques, l'arbre de décision est simple, rapide et facilement interprétable. Cependant, il souffre de surapprentissage et n'est adapté qu'à des prévisions simples. Le K-Nearest Neighbors (KNN) est adaptatif et capable de prédire sur le court terme, mais il ne capture pas la tendance globale de la demande. Le modèle Random Forest se distingue par sa robustesse et sa précision [23], ce qui le rend particulièrement adapté au secteur retail (commerce en détail). Son principal inconvénient reste l'absence de prise en compte des variables externes telles que les promotions ou la météo, qui peuvent fortement influencer la demande. Pour la planification des stocks de sécurité, des méthodes statistiques comme l'OLS et la Programmation Linéaire (PL) ont été utilisées [30]). L'OLS est simple et interprétable, efficace pour des petits échantillons et des objectifs précis, tandis que la PL est plus robuste dans des contextes multi-facteurs, mais nécessite une optimisation des paramètres pour fonctionner correctement.

Le Deep Learning permet de modéliser des relations complexes et non linéaires dans les séries temporelles. Le LSTM multivarié intègre plusieurs variables explicatives comme les prix, les promotions et les conditions économiques [24]. Il capture les dépendances longues dans les séries temporelles et atteint une précision notable, ce qui en fait un outil performant pour la prévision multi-produits. Cependant, il nécessite de grandes quantités de données et peut être difficile à généraliser à d'autres contextes ou produits. L'approche GRU optimisé par IASO [31] améliore la précision en optimisant automatiquement les paramètres du réseau. Elle démontre une grande capacité pour la prédiction fine de stocks sensibles. Le principal inconvénient reste sa complexité computationnelle élevée, ce qui peut constituer un frein pour certaines entreprises.

Les méthodes hybrides combinent des modèles linéaires et non linéaires pour maximiser la précision. Par exemple, le modèle ARIMA et LSTM combine la linéarité d'ARIMA avec la capacité de LSTM à modéliser les résidus non linéaires. Cette approche est particulièrement efficace pour l'e-commerce et les produits saisonniers, offrant une amélioration notable des métriques de performance [26]. Le FCRBM [29], quant à lui, est adapté à des prévisions en

temps réel, notamment dans les Smart Grids ou pour des stocks sensibles. Mais sa complexité et la difficulté de mise en œuvre constituent ses principales limites.

Des travaux de comparaison [25] montrent que les méthodes classiques de machine learning (régression linéaire, KNN, SVM, ANN) offrent des performances limitées pour la prédiction des prix des actions en raison de la forte non-linéarité et de la nature temporelle des données financières. Les modèles de Deep Learning, en particulier les LSTM, obtiennent de meilleurs résultats grâce à leur capacité à capturer les dépendances temporelles à long terme et à intégrer des informations supplémentaires comme l'analyse de sentiment.

Contre tenu de cet état de l'art nous formulons les propositions de recherche suivantes :

Proposition de recherche 1 : le machine learning aide à une meilleure précision des prédictions ;

Proposition de recherche 2 : avec le machine learning on peut bien faire une gestion plus intelligente des approvisionnements.

2.2.3. Logiciels et outils de gestion intelligentes des stocks et approvisionnements

2.2.3.1. NetSuite ERP

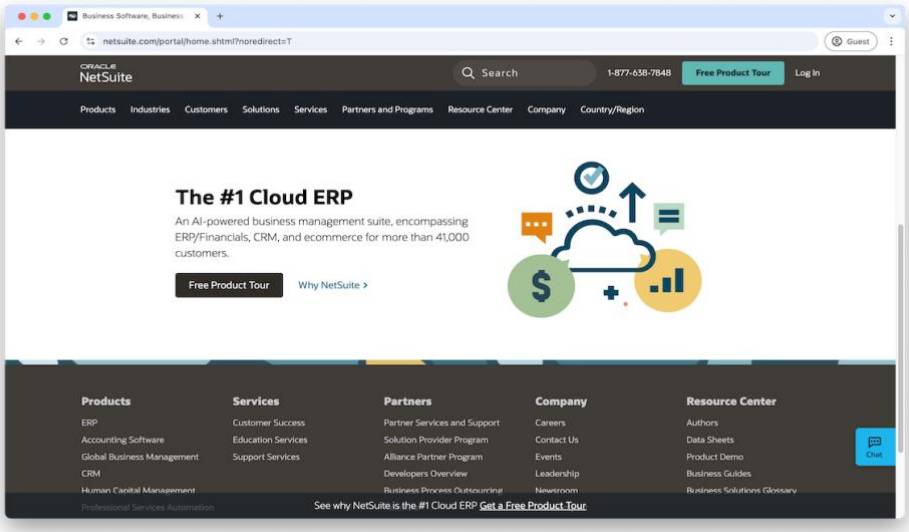


Figure 12. Site du logiciel NetSuite

Source : Web

NetSuite ERP est une solution de gestion d'entreprise conçue pour les organisations de taille moyenne et grande, notamment celles disposant de chaînes logistiques complexes impliquant plusieurs entrepôts et des flux intégrés entre ventes, stocks et comptabilité. Il s'agit d'un

système complet, accessible via un abonnement mensuel d'environ 999 dollars auquel s'ajoutent des frais par utilisateur. La plateforme offre une visibilité en temps réel des stocks sur différents sites, permet d'automatiser le réapprovisionnement selon des règles définies et intègre des fonctionnalités avancées telles que la planification de la demande, la gestion des entrepôts et des expéditions etc. Elle se distingue par sa capacité à centraliser l'ensemble des opérations de l'entreprise et à s'intégrer avec d'autres modules, ce qui améliore l'efficacité globale et réduit les erreurs. Toutefois, cette solution présente certaines limites, notamment un coût élevé et une mise en œuvre complexe nécessitant des compétences techniques et une organisation structurée. [11]

2.2.3.2. Zoho Inventory

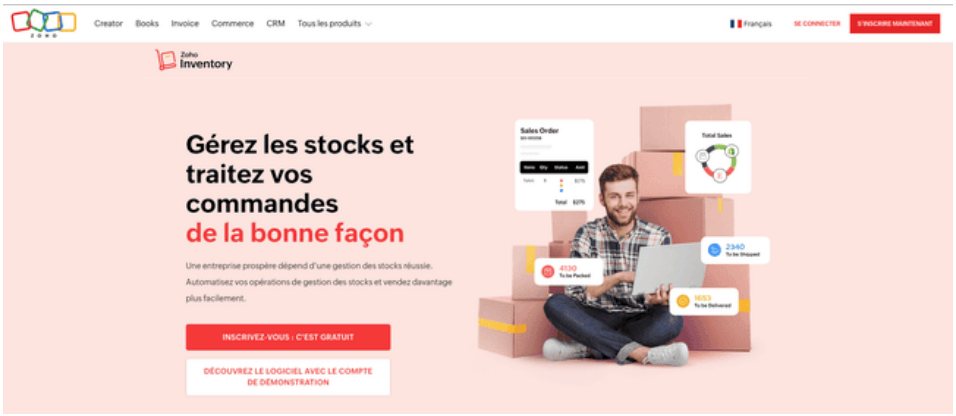


Figure 13. Page d'accueil Zoho inventaire

Source : Web

Zoho Inventory est une solution cloud intuitive destinée principalement aux PME, startups et petits commerces souhaitant une gestion de stock simple et efficace. Elle permet le suivi en temps réel des stocks, la gestion des entrées et sorties, des transferts entre entrepôts, ainsi que la traçabilité des produits via codes-barres, lots et numéros de série. Elle prend également en charge la gestion des commandes, des réceptions fournisseurs, des expéditions et des ventes, tout en proposant des alertes de réapprovisionnement et des possibilités d'automatisation. Grâce à ses intégrations avec des outils de comptabilité, de vente ou d'e-commerce, elle facilite les opérations multicanales. Son principal atout réside dans son coût abordable, avec des plans tarifaires flexibles adaptés aux besoins des entreprises, ce qui en fait une solution accessible et facile à prendre en main. Toutefois, elle reste moins adaptée aux structures ayant des besoins complexes ou des volumes très élevés, certaines fonctionnalités avancées étant disponibles

[11] https://stackverdict.com/best-inventory-management-software/?utm consulté le 08/12/2025

uniquement via des offres payantes et le nombre d'utilisateurs pouvant être limité dans les formules de base. [12]

Les différents tarifs

Source : Conception hauteur

Plan	Prix/Mois	Utilisateurs
Gratuit	0 euro	1
Standard	29 euro	2
Professionnelle	79 euro	2
Premium	129 euro	2
Entreprise	249 euro	7

2.2.3.3. Odoo Inventory (module de l'ERP Odoo)

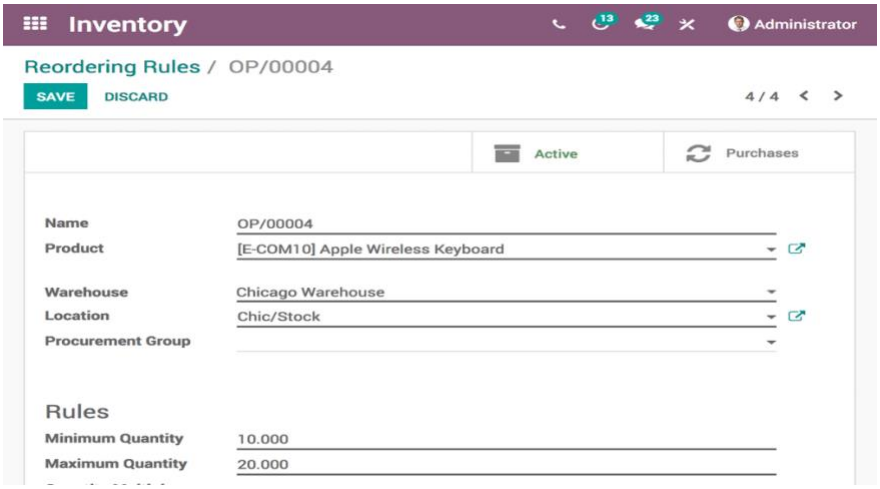


Figure 14. Logiciel d'inventaire Odoo [13]

Source : Web

Odoo est un ERP open-source hautement modulable, utilisé aussi bien par les petites que les grandes entreprises et dont le module Inventory permet de gérer efficacement les stocks, les entrepôts, les réceptions etc. Il offre des fonctionnalités complètes telles que la gestion multi-entrepôts et multi-emplacements, la traçabilité des produits via lots, numéros de série et codes-barres. Le système permet également d'automatiser les réapprovisionnements selon des règles personnalisables et de générer des commandes internes ou externes. Son principal atout réside

[12] https://www.bestdevops.com/top-10-inventory-management-software-tools-in-2025-features-pros-cons-compariso Consulté le 08/12/2025
[13] https://www.dynapps.ch/voorraadbeheer-odoo Consulté le 09/12/2025

dans sa grande flexibilité et sa capacité de personnalisation, offrant un bon compromis entre richesse fonctionnelle et coût pour les TPE/PME souhaitant évoluer. Cependant, sa mise en place peut devenir complexe et nécessiter des compétences techniques, notamment pour des besoins avancés ou une forte automatisation. De plus, certaines personnalisations ou modules supplémentaires peuvent engendrer des coûts additionnels et l'ajout progressif de modules peut rendre le système plus difficile à gérer, d'autant plus que la version gratuite reste limitée en fonctionnalités.[14]

2.2.3.4. Fishbowl Inventory

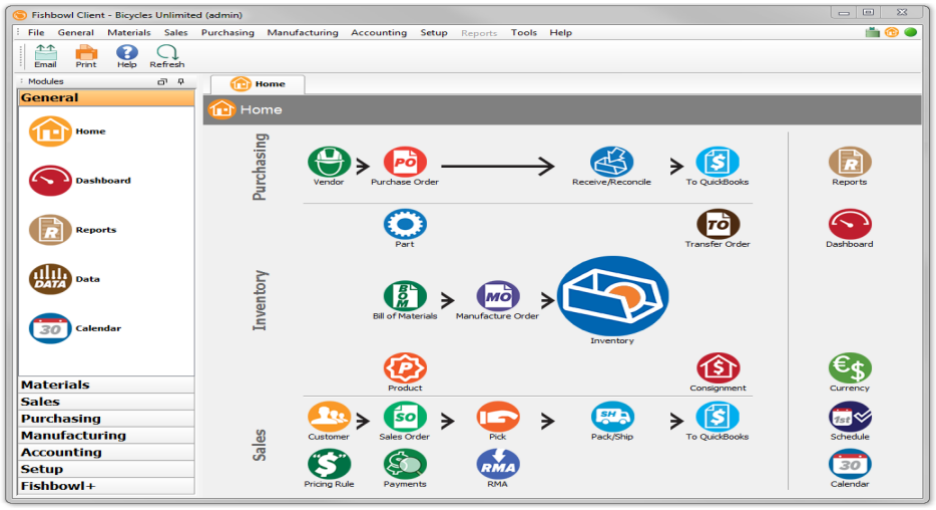


Figure 15. Page d'accueil FishbowI inventory [15]

Source : Web

Fishbowl Inventory est une solution orientée vers les entreprises manufacturières et les distributeurs ayant des volumes de stock importants et des besoins en gestion d'entrepôts et de production. Elle permet un suivi en temps réel des stocks, la gestion des entrées et sorties, ainsi que le pilotage multi-entrepôts. L'outil se distingue aussi par sa capacité de prendre en charge la production à travers la gestion des ordres de fabrication, ce qui le rend particulièrement adapté aux entreprises qui assemblent ou fabriquent des produits. Il propose en outre des fonctionnalités complètes de gestion des commandes, des transferts, des inventaires, ainsi que la traçabilité via les lots et numéros de série, avec une possibilité d'intégration à des systèmes comptables. Cette solution constitue ainsi un bon choix pour les entreprises industrielles de taille moyenne cherchant à combiner gestion de stock, production, achats et ventes dans un

[14] https://www.bestdevops.com/top-10-inventory-management-software-tools-in-2025-features-pros-cons-comparison Consulté le 08/12/2025

[15] https://israellopezconsulting.com/wp-content/uploads/2014/11/2014-11-21-07_07_30-HomeScreen.png consulté le 09/12/2025

même environnement. Toutefois, elle présente certaines limites, notamment une complexité de paramétrage qui la rend moins adaptée aux petites structures, un coût relativement élevé par rapport à certaines solutions, ainsi qu'une installation et une configuration exigeantes nécessitant souvent une formation approfondie pour les utilisateurs.[16]

2.2.3.5. Cin7

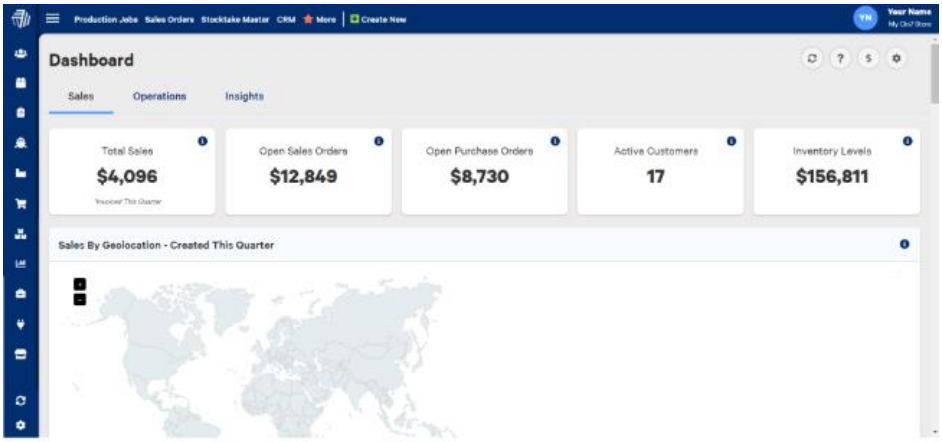


Figure 16. Tableau de bord de la page d'accueil Cin7[17]

Source : Web

Cin7 est une solution cloud de gestion des stocks et de distribution multicanale conçue pour unifier et centraliser les opérations des entreprises évoluant à la fois dans la vente en ligne, le e-commerce et les magasins physiques. Ses fonctionnalités principales incluent la gestion multi-dépôts, le transfert entre entrepôts, le suivi automatique des commandes ainsi que la synchronisation en temps réel des flux de vente, évitant ainsi toute incohérence entre les différents canaux. Bien que cet outil se distingue également par son support de la production et sa traçabilité automatisée, son coût évolutif basé sur le nombre d'utilisateurs et d'entrepôts activés peut s'avérer élevé. De plus, sa configuration initiale demande un investissement en temps non négligeable, ses capacités de reporting restent limitées, et la richesse de ses fonctions peut être disproportionnée pour des entreprises de petite taille ou aux besoins très simples.[18]

[16] https://www.bestdevops.com/top-10-inventory-management-software-tools-in-2025-features-pros-cons-comparison Consulté le 08/12/2025

[17] https://help.omni.cin7.com/hc/en-us/articles/9052702412815-Cin7-Omni-homepage-dashboard Consulté le 09/12/2025

[18] https://www.bestdevops.com/top-10-inventory-management-software-tools-in-2025-features-pros-cons-comparison Consulté le 08/12/2025

2.2.3.6. Lightspeed Retail

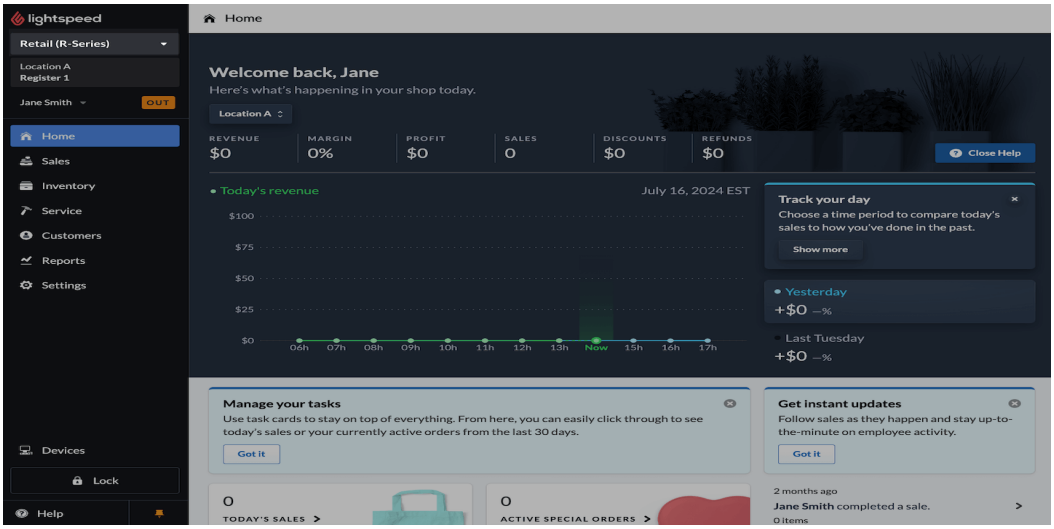


Figure 17. Page d'accueil Lightspeed retail [19]

Source : Web

Lightspeed Retail est une solution cloud de point de vente (POS) et de gestion des stocks, spécialement conçue pour simplifier les opérations quotidiennes des boutiques, des magasins physiques et des commerces de détail multicanaux. Ce système se distingue par sa facilité d'utilisation, son interface pensée pour le retail et sa synchronisation automatique en temps réel de l'inventaire après chaque vente. Il intègre de plus des fonctionnalités pratiques telles que la gestion des catalogues fournisseurs, la création de bons de commande et le calcul du coût des marchandises vendues (CMV) via une interface accessible sur mobile ou tablette. Toutefois, cet outil s'avère moins adapté pour les entreprises gérant des entrepôts multiples ou des flux logistiques complexes, et ses fonctionnalités restent limitées en comparaison d'un ERP ou d'un système de gestion d'entrepôt complet, excluant notamment les besoins avancés de planification et de production.[20]

[19] https://retail-support.lightspeedhq.com/hc/fr/articles/8808070440987-Utilisation-de-l-arri%C3%A8re-boutique-Commerce-%C3%A9lectronique-s%C3%A9rie-E Consulté le 08/12/2025

[20] https://www.shopify.com/fr/blog/logiciels-de-gestion-de-stock-le-top-10-2023 Consulté le 09/12/2025

2.2.3.7. inFlow Inventory

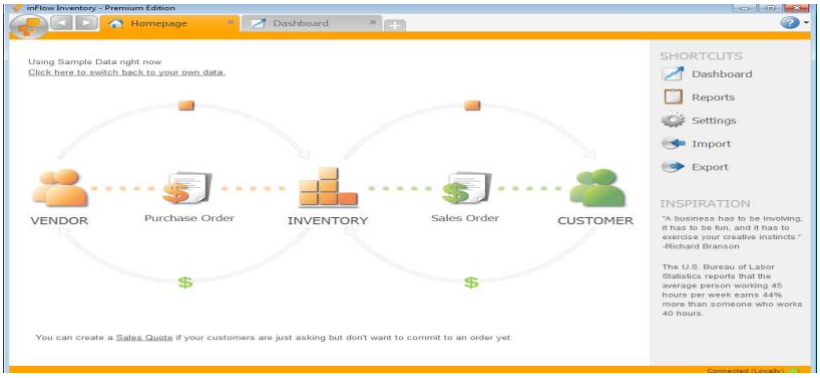


Figure 18. Page d'accueil InFlow Premium [21]

Source : Web

InFlow Inventory est une solution de gestion des stocks polyvalente et abordable, spécialement conçue pour les petites et moyennes entreprises qui souhaitent abandonner la méthode traditionnelle au profit d'un outil plus structuré et sans investissement budgétaire majeur. Ce système se distingue par sa prise en main facile et son interface simple, regroupant des fonctionnalités essentielles telles que le suivi des entrées et sorties, les alertes de seuil critique, les transferts de produits, ainsi que la gestion des bons de commande, des réceptions et des expéditions. S'il représente un excellent compromis pour les petites structures aux flux modérés n'ayant pas l'utilité d'un ERP complet, il montre rapidement ses limites face à des chaînes logistiques complexes. InFlow Inventory souffre en effet d'un manque d'évolutivité en cas de forte croissance de l'entreprise, n'intègre pas de gestion multidevises et s'avère peu adapté à la gestion de sites multiples ou aux flux de production volumineux.[22]

2.2.3.8. ERPNext

ERPNext est un progiciel de gestion intégré (ERP) open-source, distribué sous licence GPL-3.0, qui se distingue par sa modularité et sa capacité à centraliser l'ensemble des activités des PME, qu'il s'agisse de la manufacture, de la distribution, des services ou du commerce. Cette solution complète intègre des modules variés couvrant la gestion multi-entrepôts, le suivi des stocks, la planification des achats, la gestion des commandes ainsi que le pilotage de la production et de la comptabilité.

L'un de ses principaux atouts réside dans sa flexibilité et son coût très avantageux, étant gratuit lorsqu'il est auto-hébergé, ce qui permet aux structures en croissance d'adapter l'outil à leurs

[21] https://onpremise.inflowinventory.com/support/v2/overview/ Consulté le 09/12/2025
[22] https://www.bestdevops.com/top-10-inventory-management-software-tools-in-2025-features-pros-cons-comparison Consulté le 09/12/2025

besoins spécifiques avec un budget limité. Toutefois, ERPNext offre un niveau de support d'entreprise inférieur à celui des solutions commerciales et dépend fortement de sa communauté. De plus, son installation, sa maintenance et son exploitation pour des flux intensifs ou des architectures complexes exigent des compétences techniques pointues.[23]

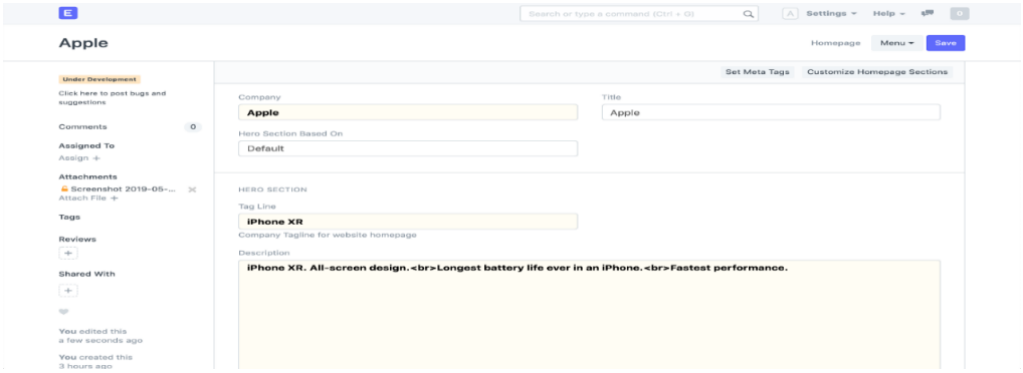


Figure 1. Aperçu du logiciel ERPNext [24]

Source : Web

2.2.3.9. Sumtracker

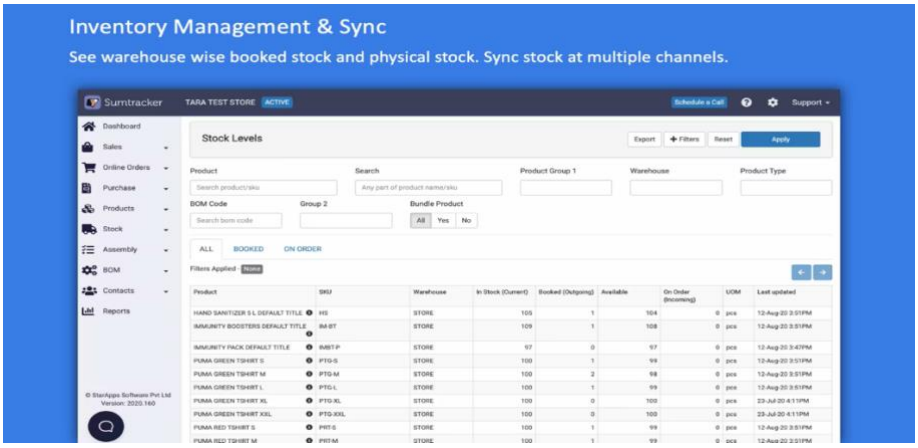


Figure 2. Aperçu du logiciel Sumtracker [25]

Source : Web

Sumtracker est une solution cloud de gestion des stocks conçue pour les PME et les entreprises de e-commerce retail opérant sur de multiples sites ou canaux de vente, tels que des entrepôts, des magasins physiques et des boutiques en ligne. Ce système se distingue par son excellent rapport coût-fonctionnalités, sa flexibilité et sa facilité de déploiement pour la gestion multi-site, offrant un suivi des stocks en temps réel, une gestion fluide des transferts et une

[23] https://en.wikipedia.org/wiki/ERPNext Consulté le 09/12/2025
[24] https://erpnext.fr.softonic.com/applications-web Consulté le 09/12/2025
[25] https://gdm-catalog-fmapi-prod.imgix.net/ProductScreenshot/84194962-dc52-432f-a81a-5e8862a1dac8.png?auto=format&q=50 Consulté le 25/03/2026

synchronisation efficace grâce à ses capacités d'intégration avec diverses plateformes de vente. Il intègre également des outils analytiques permettant de générer des alertes de rupture ou de surstock, de calculer les jours de couverture restants et d'établir des prédictions de la demande basées sur les données historiques. Néanmoins, Sumtracker est moins complet qu'un ERP traditionnel, nécessitant l'utilisation d'outils complémentaires pour couvrir les besoins en comptabilité, en finance ou en production et peut voir ses capacités limitées face à des volumes de transactions très importants ou à des flux logistiques hautement complexes.[26]

2.2.3.10. SAP Business One

Figure 3. Aperçu du logiciel SAP Business [27]

Source : Web

SAP Business One est un progiciel de gestion intégré (ERP) complet conçu pour les PME en croissance et les structures multi-sites nécessitant une centralisation rigoureuse de leurs opérations industrielles, commerciales et de distribution. Ce système unifié se distingue par sa forte intégration interne, synchronisant en temps réel la finance, le CRM, les ventes et une gestion avancée des stocks, suivi multi-entrepôts, numéros de lots, emplacements et rotations. Il automatise également le cycle des achats et le réapprovisionnement tout en offrant des outils d'analyse de performance et des tableaux de bord en temps réel. Bien qu'il propose des fonctionnalités professionnelles haut de gamme à un coût inférieur à celui des ERP pour grandes entreprises et qu'il bénéficie d'un riche écosystème d'extensions, SAP Business One implique un investissement financier plus élevé qu'un simple logiciel de stock. De plus, son déploiement est plus lourd, exigeant un paramétrage professionnel obligatoire et sa complexité fonctionnelle

[26] https://www.shopify.com/fr/blog/logiciels-de-gestion-de-stock-le-top-10-2023 Consulté le 09/12/2025
[27] https://www.projectline.ca/hs-fs/hubfs/SAP-Business-One-HANA-Dashboard-Widget-Guide-1200x600.png?width=715&name=SAP-Business-One-HANA-Dashboard-Widget-Guide-1200x600.png Consulté le 25/03/2026

peut se révèle être disproportionnée pour de très petites structures ou est moins flexible face à des changements de processus ultra-rapides.

Tableau Comparatif

Source : Conception hauteur

Nom de l'outil	**Idéal pour**	**Plateformes prises en charge**	**Caractéristique remarquable**	**Tarification**
NetSuite ERP	Moyennes et grandes entreprises, chaînes logistiques complexes	Cloud	ERP complet intégrant stock, achats, ventes, finance	999$ /mois frais par utilisateur
Zoho Inventory	PME, TPE, startups, commerces	Cloud et Web	Simplicité, intégration écosystème Zoho	0 à 249€ /mois selon plan
Odoo Inventory	TPE à grandes entreprises (modulable)	Cloud et On-premise	ERP open-source hautement personnalisable	Gratuit avec options payantes
Fishbowl Inventory	Industrie, fabrication, distribution	Cloud et Desktop	Gestion d'entrepôt et support de production	4 395$ /an
Cin7	Retail, distribution multicanal, e-commerce	Cloud	Synchronisation multicanal avancée	À partir de 299$ /mois
Lightspeed Retail	Boutiques et commerces physiques	Cloud et Mobile	POS et gestion des stocks intégrée	À partir de 69 $/mois
inFlow Inventory	PME, petites entreprises	Cloud et Desktop	Outil simple pour remplacer Excel	À partir de 79$/mois
ERPNext	PME, entreprises en croissance	Cloud, On-premise	ERP complet gratuit si auto-hébergé	Gratuit coût d'hébergement
Sumtracker	PME e-commerce retail multi-sites	Cloud	Gestion multi-entrepôts et prévisions simples	Abordable, plan variable
SAP Business One	PME et entreprises en croissance	Cloud et Desktop	ERP intégré avec finance et production	94$ /utilisateur /mois

2.3. Positionnement

Après l'analyse détaillée des différentes approches de prédiction des niveaux de stock, il apparaît que le machine learning supervisé constitue une solution robuste et pragmatique pour les entreprises souhaitant optimiser leur gestion des stocks. Les modèles supervisés tels que la régression, le Random Forest, KNN, SVM (ou le XGBoost) offrent des résultats fiables avec des métriques de performance solides (R^2 élevé, MAE et RMSE raisonnables), tout en restant relativement simples à mettre en œuvre et interprétables.

Bien que le Deep Learning offre une précision supérieure dans certains contextes, ils requièrent de grandes quantités de données et une puissance computationnelle importante, ce qui peut représenter un frein pour les entreprises avec des ressources limitées. À l'inverse, le Machine Learning supervisé permet d'atteindre un bon compromis entre performance, rapidité d'implémentation et lisibilité des résultats, tout en pouvant intégrer progressivement des variables externes pour améliorer la précision.

Ainsi, dans le cadre de notre étude sur la prédiction des niveaux de stock, nous nous positionnons sur l'usage du modèle du Machine Learning supervisé qui apparaît comme le choix le plus pertinent. En effet, il permet d'obtenir des prévisions fiables et exploitables, tout en conservant une flexibilité pour évoluer vers des approches plus complexes si nécessaire. Ce choix méthodologique garantit une optimisation efficace des approvisionnements et constitue une base solide pour l'adaptation future des modèles aux besoins spécifiques de l'entreprise.

De ce fait, afin de tirer des avantages sur un modèle sans en laisser ceux d'un autre tout aussi utile, une approche hybride combinant Random Forest et XGBoost constitue une solution performante pour la prédiction des niveaux de stock. En effet :

- **Le modèle Random Forest** : Il a été retenu dans cette étude en raison de sa robustesse et de ses bonnes performances en prédiction, notamment sur des données complexes et non linéaires comme celles liées aux niveaux de stock. Il repose sur une approche d'ensemble (ensemble learning) consistant à combiner plusieurs modèles simples afin d'obtenir un modèle global plus performant. Le principe de fonctionnement du Random Forest est basé sur la construction d'un grand nombre d'arbres de décision à partir de sous-échantillons des données d'apprentissage. Chaque arbre est entraîné sur une portion différente du jeu de données, sélectionnée aléatoirement, ce qui permet d'introduire de la diversité dans les modèles générés. À cela s'ajoute une sélection aléatoire des variables explicatives à chaque division de l'arbre, renforçant ainsi le

caractère aléatoire du modèle et limitant le risque de surapprentissage. La prédiction finale est obtenue par agrégation des résultats de l'ensemble des arbres [35, 37]. Dans le cas d'un problème de régression, comme la prédiction des niveaux de stock, cette agrégation correspond à la moyenne des prédictions individuelles des arbres [35]. Cette combinaison de plusieurs « apprenants faibles » permet d'obtenir un modèle global plus stable, précis et moins sensible au bruit présent dans les données [36]. L'objectif est de générer des arbres à la fois performants et suffisamment diversifiés, afin de réduire la variance globale et d'améliorer la capacité de généralisation du modèle [36]. Bien que le Random Forest soit extrêmement robuste, il peut parfois manquer de finesse sur les changements de tendance brutaux.

- **Le XGBoost** (ou Extreme Gradient Boosting) : c'est un algorithme d'apprentissage automatique supervisé reconnu pour sa rapidité, sa vitesse d'exécution et ses performances élevées, particulièrement avec des données structurées et tabulaires [32, 33]. Développé en 2016, il s'est imposé comme l'un des outils les plus puissants pour résoudre des problèmes de classification et de régression dans divers domaines industriels [24, 33]. L'algorithme repose sur le cadre du gradient boosting appliqué à des arbres de décision [33]. Il fonctionne de manière séquentielle, au lieu de construire des arbres en parallèle, il les construit en série et chaque nouvel arbre est ajouté pour corriger les erreurs commises par les arbres précédents afin de minimiser globalement la fonction de perte [33, 39 ; 40]. Le principe est celui de l'amélioration continue, le premier arbre fait une prédiction simpliste, le deuxième tente de corriger les erreurs du premier, le troisième corrige les erreurs du deuxième, et ainsi de suite. Ce processus s'appelle le Gradient Boosting. Il intègre des mécanismes pour gérer automatiquement les données manquantes, le bruit et les structures de données creuses (sparse aware) [33, 40]. Pour éviter le surapprentissage XGBoost inclut des termes de régularisation dans sa fonction, ainsi qu'un seuil de division qui contrôle la croissance de l'arbre [38, 39].

Dans un premier temps, les deux modèles sont entraînés en parallèle sur le même jeu de données d'apprentissage, chacun construisant ses propres prédictions en fonction de sa logique interne ; agrégation d'arbres indépendants pour le Random Forest et correction séquentielle des erreurs pour XGBoost. Ainsi, le Random Forest traite les données en cherchant à réduire la variance (les erreurs dues aux fluctuations aléatoires). Le XGBoost traite les mêmes données en cherchant à réduire le biais (les erreurs de précision sur les valeurs extrêmes). Dans un second temps, une fois les deux modèles entraînés, nous fusionnons leurs résultats de prédictions pour

obtenir une prédiction finale plus robuste. Nous avons privilégié la méthode du **Stracking** qui consiste à faire apprendre un modèle à partir des prévisions combinées des deux algorithmes. Cette fusion permet de compenser les faiblesses de l'un par les forces de l'autre. Si le Random Forest est trop prudent sur un pic de vente, le XGBoost viendra pousser la prédiction vers le haut pour coller à la réalité. Cela garantit à l'entreprise une prédiction fiable, même lors de variations brusques de la demande (promotions, veilles de fêtes etc.).

2.4. Conclusion

En résumé, l'analyse de l'état de l'art sur la prédiction des niveaux de stock met en évidence la richesse et la diversité des approches offertes par le Machine Learning et le Deep Learning. Les modèles supervisés classiques, tels que Random Forest, KNN, OLS ou la Programmation Linéaire, se révèlent robustes, interprétables et adaptés à une mise en œuvre pratique, offrant un bon compromis entre précision et simplicité. Les techniques de Deep Learning et les méthodes hybrides permettent d'améliorer encore la performance des prévisions, notamment pour des séries temporelles complexes ou des environnements dynamiques, mais nécessitent des volumes de données importants et des ressources computationnelles élevées. Ainsi, cette analyse souligne que le Machine Learning supervisé offre une solution fiable et accessible pour la prédiction des niveaux de stock. Il permet non seulement d'obtenir des prévisions précises à partir des données historiques, mais aussi de poser les bases pour intégrer progressivement des facteurs externes ou des modèles plus complexes. Cette approche représente donc une base solide pour optimiser la gestion des stocks, réduire les coûts logistiques et soutenir la prise de décision stratégique au sein de l'entreprise.

Chapitre 3 : Méthodologie et Conception de la solution

3.1. Introduction

Ce chapitre présente la démarche méthodologique adoptée pour la conception et la mise en œuvre du modèle de prédiction des niveaux de stock. Il décrit les différentes étapes nécessaires à la transformation des données brutes en informations exploitables pour la prise de décision. Après une présentation des données utilisées, une attention particulière est accordée à leur préparation, notamment à travers les opérations de nettoyage et la création de variables pertinentes. Enfin, le processus d'entraînement et de validation des modèles est détaillé afin de garantir la fiabilité et la précision des prédictions obtenues.

3.2. Cahier de charge

Le cahier des charges définit clairement les besoins et les exigences techniques pour mettre en place un système de prédiction des niveaux de stock efficace. Il permet de clarifier les objectifs du projet et d'orienter la mise en œuvre d'une solution répondant aux attentes des utilisateurs et aux standards techniques.

3.2.1. Contexte du projet

Ceci est un projet de prédiction des niveaux de stocks par machine learning pour une gestion optimale des approvisionnements. Ainsi, avec le machine learning, nous aimerions mettre en place des stratégies et des outils pour gérer de manière efficace et efficiente les flux de marchandises et ainsi avoir connaissance des besoins en approvisionnement pour mieux les optimisés.

3.2.2. Objectif

Notre objectif est de nous assurer que grâce à notre solution, les entreprises puissent :

- Améliorer l'exactitude des prévisions : réduire le taux d'erreur des prévisions de demande ;
- Intégrer de nouveaux facteurs dans les modèles de prédiction : inclure les données météo, les événements spéciaux et les tendances du marché dans les modèles de prédiction ;
- Diminuer le taux de rupture de stock : améliorer la disponibilité des produits pour les clients ;
- Segmenter votre assortiment de produits : certains produits peuvent nécessiter une gestion des stocks plus spécifique ;
- Utiliser des outils d'analyse prédictive : exploiter les données pour identifier les tendances et les anomalies.

3.2.3. Le périmètre

Il est crucial de définir un périmètre clair pour orienter nos recherches. Notre solution concernera principalement le secteur de la grande distribution, où les prédictions de stock sont cruciales pour éviter les ruptures ou les excédents de stock et des pertes de ventes. La prédiction reposera sur l'analyse des données historiques et concernera les ventes passées, les commandes auprès des fournisseurs passées ainsi que les délais de livraison et le niveau de stock. Mais aussi on tiendra en compte les données externes comme les périodes de fêtes, les saisons et autres.

3.2.4. Description du cible

Notre solution de prédiction serait principalement utilisée par le département de Gestion des Stocks et Approvisionnements. Ce département est directement responsable de la gestion des stocks, des commandes et de l'approvisionnement. Il serait le principal utilisateur de la solution. Il pourrait se baser sur les prédictions pour ajuster les stratégies d'achat et éviter des excédents ou des ruptures de stock.

3.2.5. Les besoins fonctionnels

Afin de répondre efficacement aux enjeux de gestion des stocks, le système doit proposer plusieurs fonctionnalités essentielles :

- Système de prédiction : le système doit être capable de prédire les niveaux de stock pour chaque produit, à différents horizons temporels (court terme, moyen terme, long terme) ;
- Interface utilisateur : un tableau de bord interactif pour visualiser les recommandations d'approvisionnement et les alertes de rupture de manière claire ;
- Intégration avec d'autres systèmes : le système de prédiction doit pouvoir s'intégrer avec les systèmes existants de gestion des stocks ou de planification des approvisionnements.

3.2.6. Les besoins non fonctionnels

Outre les fonctionnalités principales, certains besoins non fonctionnels sont indispensables pour assurer la bonté du système. Ainsi les points essentiels sont :

- La performance : le système doit être capable de traiter de grandes quantités de données en temps réel, avec des prédictions précises à court, moyen et long terme ;
- La sécurité : le système doit garantir la sécurité des données, notamment la confidentialité des informations commerciales sensibles ;

- La fiabilité : le système doit être robuste et disponible sans interruption pendant les heures de fonctionnement.

3.2.7. Outils et technologies exigés

Pour garantir l'efficacité du projet, les outils et technologies suivants ont été utilisées :

- Environnement de développement : VS Code ;
- Éditeur de code : Jupyter NoteBook ;
- Langages de programmation : Python ;
- Outils de Machine Learning : scikit-learn ;
- Outils de traitement des données : Pandas, NumPy ;
- Outils de visualisation : Matplotlib ;
- Système d'exploitation : Windows.

3.3. Cadre expérimentale

3.3.1. Environnement d'expérimentation et bibliothèques

- **Python** : c'est un langage de programmation **élégant, robuste et de haut niveau** qui combine la puissance des langages compilés traditionnels avec la simplicité des langages de script et interprétés. Créé par **Guido van Rossum** à la fin de l'année 1989 et diffusé publiquement en 1991, il a été conçu à l'origine pour répondre aux limites des outils existants dans un contexte de recherche [41] ;
- **VS Code**[28] : c'est l'environnement de développement (IDE) principal utilisé pour structurer et éditer le code source du projet. Développé par Microsoft, il est prisé pour sa légèreté, sa polyvalence et son vaste catalogue d'extensions qui facilitent le débogage et la gestion des scripts Python ;
- **Jupyter NoteBook**[29] : il s'agit d'une application web open-source permettant de créer des documents interactifs combinant code exécutable, texte narratif et visualisations graphiques. C'est l'outil idéal pour la phase d'exploration des données (EDA) et la présentation pédagogique des résultats de prédiction ;
- **Pandas**[30] : bibliothèque incontournable pour l'analyse de données en Python, elle introduit les structures de données appelées « DataFrames ». Elle permet de manipuler,

[28] **Source :** code.visualstudio.com Consulté le 14/04/2026
[29] **Source :** jupyter.org Consulté le 14/04/2026
[30] **Source :** pandas.pydata.org Consulté le 14/04/2026

nettoyer et transformer des tableaux de données complexes (comme des historiques de ventes) de manière performante et intuitive ;

- **NumPy**[31] : c'est la bibliothèque fondamentale pour le calcul scientifique. Elle fournit des structures de tableaux multidimensionnels de haute performance et des fonctions mathématiques nécessaires au traitement des données numériques avant leur passage dans les algorithmes de machine learning ;
- **Scikit-Learn**[32] : c'est la bibliothèque de référence pour le Machine Learning classique. Elle offre des outils robustes pour le partitionnement des données, ainsi qu'une implémentation efficace des algorithmes de régression et de classification, dont le **Random Forest** utilisé dans ce projet.
- **Matplotlib**[33] **:** c'est une bibliothèque Python utilisée pour la visualisation de données. Elle permet de créer différents types de graphiques, tels que des courbes, des histogrammes ou des diagrammes, afin de représenter les données de manière claire. Elle offre également de nombreuses options de personnalisation, comme le choix des couleurs, des styles et des formats, ce qui en fait un outil performent pour l'analyse et la présentation des résultats.

Des guides d'installation des outils et bibliothèques sont recensés dans le tableau suivant :

Guides d'installation des outils et API

Source : Conception Hauteur

Outils	Lien d'installation des outils
Python	Download Python | Python.org
VS Code	code.visualstudio.com
Jupyter NoteBook	jupyter.org
Pandas	pandas.pydata.org
NumPy	numpy.org
Scikit-Learn	scikit-learn.org
Matplotlib	https://matplotlib.org/stable/index.html

[31] **Source :** numpy.org Consulté le 14/04/2026
[32] **Source :** scikit-learn.org Consulté le 14/04/2026
[33] **Source :** https://matplotlib.org/stable/index.html consulté le 14/05/2026

3.3.2. Métriques de performance de prédiction des stocks

Tableau des Métriques d'évaluation

Source : Conception Hauteur

Métrique	Définition	Utilité	Exemple
R^2 (Coefficient de détermination)	Mesure la proportion de la variance des données expliquée par le modèle.	Évaluer la précision globale du modèle. Une valeur proche de 1 signifie que le modèle explique bien les variations des stocks.	R^2 = 0,92 : le modèle explique 92 % de la variance des niveaux de stock.
MAE (Mean Absolute Error)	Moyenne des écarts absolus entre les valeurs réelles et prédites.	Indique l'erreur moyenne sans tenir compte du sens. Plus le MAE est faible, plus le modèle est précis.	MAE = 15 unités : en moyenne, la prédiction est à 15 unités près du stock réel.
RMSE (Root Mean Squared Error)	Racine carrée de la moyenne des carrés des erreurs.	Sensible aux grandes erreurs ; utile quand on veut pénaliser fortement les erreurs importantes.	RMSE = 20 unités : certaines prévisions peuvent s'écarter fortement du stock réel.
MSE (Mean Squared Error)	Moyenne des carrés des erreurs entre valeurs réelles et prédites.	Évalue l'erreur globale, mais reste à l'échelle quadratique ; utile pour comparer différents modèles.	MSE = 400 : erreurs quadratiques moyennes de 400 unités².
MAPE (Mean Absolute Percentage Error)	Moyenne des erreurs absolues en pourcentage par rapport aux valeurs réelles.	Permet de comparer les performances entre produits ou séries de stocks de tailles différentes.	MAPE = 8 % : en moyenne, les prévisions s'écartent de 8 % du stock réel.
SMAPE (Symmetric Mean Absolute Percentage Error)	Variante du MAPE normalisée par la moyenne des valeurs réelles et prédites.	Évite les problèmes du MAPE lorsque les valeurs réelles sont proches de zéro.	SMAPE = 10 % : l'erreur relative moyenne corrigée est de 10 %.
MedAE (Median Absolute Error)	Médiane des écarts absolus entre valeurs réelles et prédites.	Moins sensible aux valeurs aberrantes que le MAE.	MedAE = 12 unités : la moitié des prévisions ont une erreur ≤ 12 unités.
RMSLE (Root Mean Squared Logarithmic Error)	Racine de la moyenne des carrés des erreurs logarithmiques.	Utile quand les stocks varient sur de grandes échelles et qu'on veut pénaliser moins les grandes valeurs.	RMSLE = 0,05 : erreurs logarithmiques faibles malgré des stocks très différents.
Bias / MBE (Mean Bias Error)	Moyenne des erreurs (réelles – prédites).	Indique si le modèle a tendance à **surestimer** ou **sous-estimer** les stocks.	MBE = -3 : le modèle sous-estime en moyenne de 3 unités.
Tracking Signal	Ratio de la somme des erreurs cumulées sur la MAE.	Permet de détecter une dérive ou un biais systématique dans les prévisions.	Tracking Signal = 2 : la prévision commence à diverger des valeurs réelles.
RRMSE (Relative RMSE)	RMSE normalisé par la moyenne ou la plage des valeurs.	Permet de comparer la performance de modèles sur différents produits ou périodes.	RRMSE = 0,12 : erreur relative de 12 % par rapport à la moyenne des stocks.

Pour évaluer la qualité des modèles de prédiction, plusieurs métriques peuvent être utilisées.

Dans le tableau ci-dessus, nous vous avons présenté différentes métriques d'évaluation de

performance. En somme, l'évaluation de la performance d'un modèle de prédiction des stocks ne repose que sur une combinaison d'indicateurs complétement adaptés aux objectif opérationnels. Alors que le coefficient de détermination R^2 valide la pertinence globale du modèle, des mesures absolues comme la MAE, la MSE ou la RMSE permettent de qualifier l'écart entre les valeurs prédites et réelles tout en appliquant (pour le dernier), une pénalité accrue aux erreurs les plus grandes. Ainsi, les métriques comme le MAPE, le SMAPE ou la RRMSE offrent une vision en pourcentage. La MedAE garantit une analyse robuste face aux anomalies, le RMSLE s'adapte aux variations d'échelle extrêmes, tandis que le biais MBE et le Tracking Signal jouent un rôle crucial pour identifier les tendances à la sur ou sous-estimation.

3.3.3. Présentation et source des données

Dans le cadre de cette étude, les données utilisées proviennent principalement des historiques de ventes et des niveaux de stock d'une entreprise (X). Ces données constituent la base essentielle pour la construction du modèle de prédiction. Avant toute manipulation, il est essentiel d'exploiter notre base de travail. La taille du jeu de données détermine la profondeur de l'apprentissage de nos algorithmes. Pour cette étude, nous allons exploiter un historique comprenant **116880 lignes** et **8 colonnes**, ce qui offre une base statistique solide pour capturer les tendances de consommation. La diversité des types de données permet au modèle de croiser différentes natures d'informations. Notre fichier se compose de variables de types temporelles, numériques et catégorielles. Chaque colonne de notre jeu de données joue un rôle précis dans la prédiction. Le contenu des fichiers exploités regroupe principalement :

- Le code produit
- La catégorie
- Le fournisseur
- Quantité-ouverture
- Quantité-fermeture
- Quantité-vendu
- Est-Approvisionne
- Date

	Produit	Categorie	Fournisseur	Quantite_Ouverture	Quantite_Fermeture	Quantite_Vendu	Est_Approvisionne	Date
0	Huile Dinor 5L	Alimentaire	Safa Agro	68	52	16	Non	2010-01-01
1	Coca-Cola 1.5L	Boisson	Coca-Cola Sénégal	87	80	7	Non	2010-01-01
2	Riz Royal Umbrella 25kg	Alimentaire	Patisen	48	43	5	Non	2010-01-01
3	Papier toilette Lotus	Hygiène	Unilever Afrique	72	66	6	Non	2010-01-01
4	Dentifrice Colgate	Hygiène	Colgate-Palmolive	131	122	9	Non	2010-01-01

Figure 4. Aperçu sur les données brute

Source : Conception Hauteur

3.3.4. Préparation des données

La préparation des données constitue une étape importante dans le processus de modélisation, car la qualité des données agit directement sur la performance des modèles. Une donnée manquante ou erronée pourrait fausser totalement les prévisions de stock.

3.3.4.1. Nettoyage : gestion des valeurs manquantes, suppression des anomalies

Dans un premier temps, un travail de nettoyage des données est réalisé. Celui-ci consiste à traiter les valeurs manquantes, soit en les supprimant, soit en les remplaçant par des valeurs estimées. Ensuite, une phase de transformation des données est effectuée. Les variables qualitatives sont converties en variables numériques à l'aide de techniques d'encodage afin de pouvoir être exploitées par les algorithmes. De ce fait nous n'avons constaté aucunes valeurs manquantes dans les différentes colonnes. Cependant, on a transformé la colonne « Date » en type **datetime** pour ne pas qu'elle soit considérée comme chaine da caractère**.** Les colonnes comme Produit, Catégorie, Fournisseur et Est_Approvisionne seront à leur tour après avoir été exploités, encoder avec **`LabelEncoder`** de manière à ce qu'ils n'empêchent pas aux algorithmes de fonctionner n'étant de types numériques.

3.3.4.2. Séparation des données : (jeu d'entraînement, jeu de test)

Une fois les données préparées, la phase d'entraînement des modèles est engagée. Cette étape commence par la séparation du jeu de données en deux parties : un jeu d'entraînement, utilisé pour apprendre le modèle et un jeu de test, destiné à évaluer ses performances. Les données vont de **2010 à 2025** et nous en avons fait une répartition de **2010-2023 pour le train** et de **2024-2025 pour le test**. Les modèles Random Forest et XGBoost, sont ensuite entraînés sur les données.

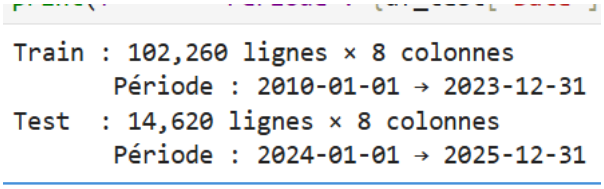


Figure 5. La taille des données d'entrainement et de test

Source : Conception Hauteur

3.3.4.3. Création de nouvelles colonnes

Une étape importante concerne également la création de nouvelles variables. Il s'agit d'enrichir le jeu de données en n y intégrant des indicateurs pertinents, tels que les jours de semaine et les

week-ends, des variables saisonnières ou des indicateurs liés aux événements (par exemple, les périodes de fêtes comme Noël ou Korité). En effet, la demande des produits peut varier en fonction de plusieurs éléments liés au temps, aux habitudes des consommateurs ainsi qu'au contexte socio-économique et religieux. Pour cette raison, les données d'entraînement et de test ont d'abord été regroupées afin de faciliter le traitement et l'enrichissement des informations. Cette étape a permis de créer de nouvelles variables capables de mieux représenter les comportements réels des clients et les différentes périodes influençant les ventes. Dans un premier temps, plusieurs informations ont été extraites à partir de la date, notamment l'année, le mois, le jour du mois, le trimestre et le nombre de semaine dans l'année. Ces variables permettent au modèle de repérer certaines tendances qui se répètent au fil du temps. Des variables liées au cycle hebdomadaire ont été également ajoutées, comme le jour de la semaine et le week-end. Cette distinction est importante, car les habitudes d'achat changent souvent entre les jours ouvrables et les week-ends (les consommateurs disposent généralement de plus de temps durant le week-end). Deux autres variables ont été créées pour représenter les comportements d'achat en début et en fin de mois. Les périodes de paiement des salaires influencent fortement la consommation. Afin de mieux prendre en compte le contexte sénégalais, les jours fériés nationaux et religieux ont aussi été intégrés dans les données. L'un permet d'identifier les jours fériés, tandis que l'autre indique les veilles de fête. Cette dernière information est particulièrement utile, car les consommateurs effectuent souvent leurs achats la veille des grandes célébrations. Par la suite, des variables appelées variables de retard ont été créées afin de donner des indices aux modèles. Ces variables représentent les ventes réalisées le jour précédent, une semaine auparavant, deux semaines auparavant ou encore environ un mois plus tôt (vendu_lag(x)). L'objectif est de permettre au modèle de comprendre que les ventes d'aujourd'hui sont souvent liées à celles des périodes précédentes. Par exemple, un produit qui s'est beaucoup vendu récemment a de fortes chances de continuer à bien se vendre dans les jours suivants. Enfin, des moyennes et des écarts-types glissants ont été calculés sur des périodes de 7, 14 et 28 jours à travers les variables vendu_rolling_mean et vendu_rolling_std. Les moyennes glissantes permettent de suivre la tendance récente des ventes, tandis que les écarts-types mesurent les variations ou l'irrégularité des ventes sur une période donnée. En ce sens, un calendrier de fête a aussi été intégré. En effet avec **Workalendar** nous nous somme basé sur le calendrier islamique pour fixer les jours de fête religieuse musulmanes. On a aussi fixé les fêtes chrétiennes et civiles étant aussi des fêtes fixes. Ces informations aident le modèle à mieux comprendre l'évolution de la demande dans le temps.

La figure ci-dessous montre une illustration de calendrier de fêtes sénégalaises pour l'année 2024.

```
Fériés 2024
 2024-01-01  →  New year
 2024-04-04  →  Fête nationale
 2024-04-10  →  Korité (Aïd al-Fitr)
 2024-05-01  →  Fête du Travail
 2024-06-16  →  Tabaski (Aïd al-Adha)
 2024-07-07  →  Nouvel An Hégirien
 2024-07-16  →  Tamkharit (Achoura)
 2024-08-15  →  Assomption
 2024-08-22  →  Magal de Touba
 2024-09-15  →  Gamou / Mawlid
 2024-11-01  →  Toussaint
 2024-12-25  →  Christmas Day

2024-04-10 est ouvrable ? False
8 avr + 5 jours ouvrables = 2024-04-16
Jours ouvrables en 2024 : 253
```

Figure 6. Calendrier de fêtes sénégalaises 2024

Source : Conception Hauteur

3.3.4.4. Analyse et prétraitement des données

Nous avons fait une représentation graphique de la distribution des ventes journalières par catégorie. En effet elle est représentée par un boxplot par catégorie et un histogramme global des ventes de tous catégories confondues.

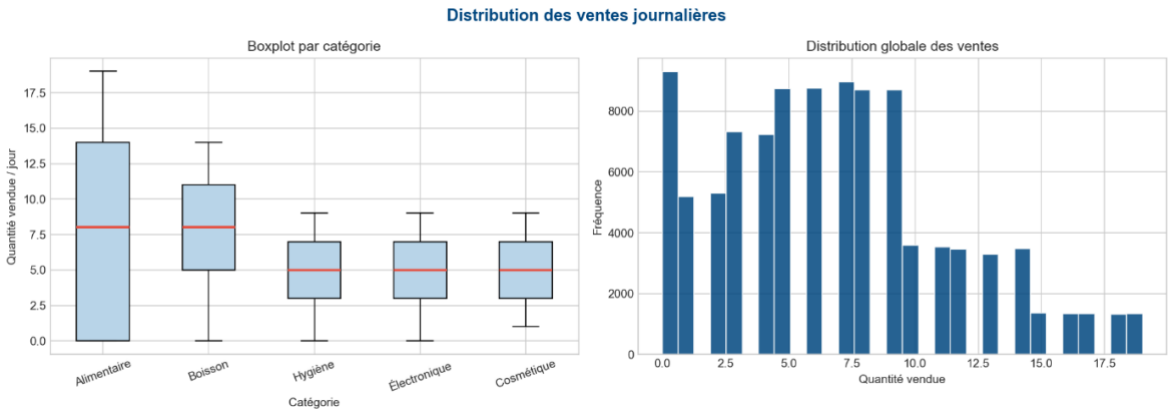


Figure 7. Distribution des ventes journalières

Source : Conception Hauteur

Le boxplot met en évidence la disposition des quantités vendu par jour pour chaque catégorie. Pour la catégorie alimentaire, les ventes par jour peuvent allés de 0 à 19 unités selon la moustache avec une médiane qui tourne autour de 8 unités. Cela peut signifier que les produits alimentaires connaissent une forte fluctuation des ventes ce qui s'explique par leur consommation régulière. Pour les boissons, avec une médiane très proche ou même égale à la catégorie alimentaire, une variabilité importante est constatée. Toutefois les vente sont moins disperser et tournent souvent aux alentours de 5 à 11 unités par jours mais peuvent aller jusqu'à 14 unités. Les catégories hygiène, électronique et cosmétique affichent une disposition beaucoup plus constante des ventes par jour. Leurs médianes tournent autour de 5 unités. Les ventes excessives y sont rares.

Après la distribution des ventes journalières nous en avons représenté une mensuelle des ventes moyennes par catégorie. Elle nous permet de voir l'évolution des ventes mensuelle pour peut-être remarquer une éventuelle saisonnalité.

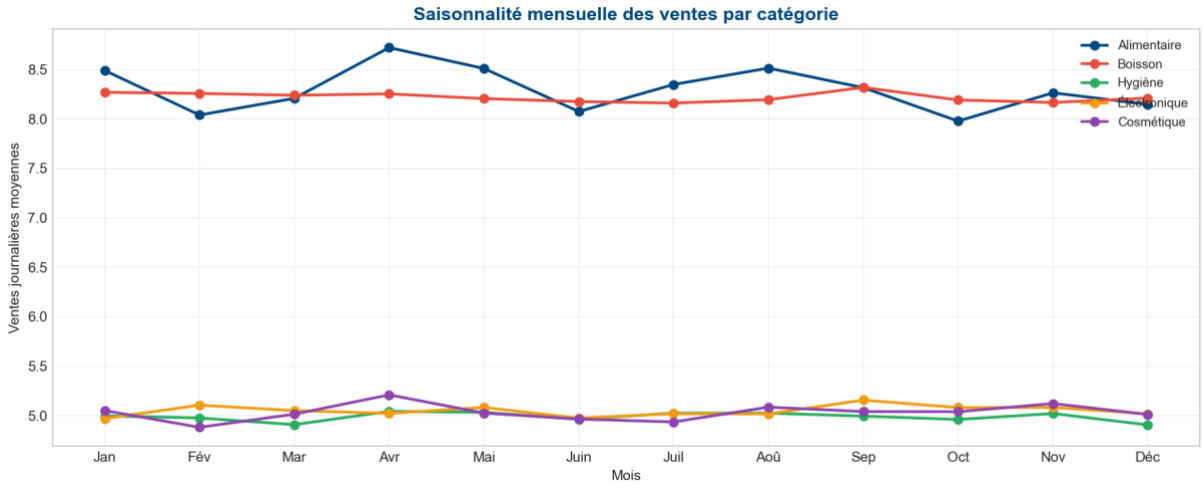


Figure 8. Saisonnalité mensuelle des ventes

Source : Conception Hauteur

Dans cette graphique, chaque courbe représente une catégorie. L'Alimentaire représenté par la couleur bleue est visuellement celle qui représente les ventes les plus élevées pour cette année. On remarque un pic aux mois d'Avril et Août ce qui peut être dû aux fêtes religieuses ou aux vacances et une légère baisse en juin et octobre. Les boissons montrent un comportement relativement stable tout au long de l'année. Cela témoigne d'une consommation régulière indépendante des saisons. Les trois catégories restantes présentes des niveaux plus faibles de ventes mais bien constants avec quelque fois de légère hausse pour les produits cosmétiques.

3.3.5. Procédure de l'étude de la viabilité financière du modèle à concevoir

Pour la viabilité du projet, on va faire l'étude technique qui consiste à analyser les besoins en investissement, le BFR le coût total du projet et le montage financière. Et ensuite dans l'étude financière nous allons analyser la rentabilité à travers la VAN, l'IP, le TRI ainsi que le risque avec le DRCI, l'indice de sécurité économique et financier et l'effet de levier d'exploitation.

3.4. Conclusion

En somme, ce troisième chapitre a permis de mettre en place l'ensemble de la démarche méthodologique nécessaire à la mise en œuvre de notre système prédictif. Le passage d'une base de données brute à un environnement de modélisation exploitable a exigé une phase rigoureuse de nettoyage, de traitement des variables qualitatives et qualitatives et de structuration temporelle. L'étape clé d'enrichissement des données, à travers la création de variables de retard, de moyennes glissantes et l'intégration du calendrier des fêtes sénégalaises, offre aux algorithmes la profondeur contextuelle indispensable pour capturer les fluctuations de la demande. Ces ensembles de traitements structurés et validés posent les fondations techniques indispensables pour aborder sereinement la phase d'apprentissage et analyser les performances des algorithmes Random Forest, XGBoost et du modèle hybride.

Chapitre 4 : Présentation et Analyse des Résultats

4.1. Introduction

Après avoir exposé la méthodologie et la conception de notre solution dans le chapitre précédent, nous présentons ici les résultats obtenus. Ce chapitre est dédié à l'évaluation de la performance de notre modèle prédictif et à l'analyse de sa pertinence pour la gestion des stocks. À travers cette étape, il s'agit de vérifier la capacité du modèle à produire des prévisions fiables et exploitables dans un contexte réel. Nous y détaillerons les mesures de précision statistique, la visualisation des prévisions face à la réalité du terrain, ainsi que l'importance des différents facteurs d'influence identifiés par l'intelligence artificielle. Une attention particulière sera également portée à l'interprétation des résultats, afin de mieux comprendre les comportements observés et d'identifier les éventuelles limites du modèle. Enfin, ce chapitre permettra de dégager des enseignements utiles pour l'amélioration des stratégies d'approvisionnement.

4.2. Résultat des modèles

4.2.1. Prédiction avec Random Forest

4.2.1.1. Architecture du modèle

Les paramètres retenus pour le modèle de Random Forest ont été fait de façon à obtenir un bon équilibre entre la capacité d'apprentissage du modèle et sa capacité à bien généraliser sur de nouvelles données. En effet, un modèle trop simple risque de ne pas capturer correctement les relations présentes dans les données, tandis qu'un modèle trop complexe peut apprendre le bruit et produire un surapprentissage. Les valeurs sélectionnées permettent ainsi d'obtenir des prédictions plus stables et plus fiables. Le paramètre n_estimators a été fixé à 300, ce qui signifie que la forêt est composée de 300 arbres de décision. L'augmentation du nombre d'arbres améliore généralement la précision et la stabilité des résultats grâce à la combinaison des prédictions de plusieurs arbres. Les différents tests réalisés ont montré qu'au-delà de 200 arbres, l'amélioration des performances devient très faible. Le choix de 300 arbres permet donc d'assurer une meilleure robustesse du modèle sans augmenter excessivement le temps de calcul. La profondeur maximale des arbres, définie par le paramètre max_depth, a été limitée car des arbres trop profonds ont tendance à mémoriser les données d'entraînement au lieu d'apprendre des tendances générales. Une profondeur de 15 niveaux offre un compromis intéressant, les arbres restent suffisamment détaillés pour détecter les relations complexes entre les variables tout en limitant le risque de mémorisation. Le paramètre min_samples_split impose qu'un nœud doit contenir au moins 10 observations avant d'être séparé en deux sous-groupes. Cette contrainte évite la création de divisions basées sur un trop petit nombre d'exemples, ce qui permet de limiter la spécialisation excessive des arbres et d'améliorer la stabilité globale du

modèle. De manière similaire, le paramètre min_samples_leaf impose qu'une feuille terminale contienne au minimum 5 observations. Cette condition permet d'éviter la formation de feuilles trop spécifiques à quelques cas particuliers. Ainsi, chaque prédiction repose sur un nombre suffisant d'observations, ce qui contribue à réduire la variance du modèle et à améliorer sa capacité de généralisation. Le paramètre max_features permet de sélectionner aléatoirement un nombre limité de variables lors de chaque division d'un arbre. Dans ce cas, environ cinq variables sont examinées à chaque nœud parmi l'ensemble des variables disponibles. Cette approche réduit la corrélation entre les arbres de la forêt, car chaque arbre se construit différemment. La diversité obtenue améliore alors les performances globales du modèle en rendant les prédictions plus robustes. Enfin, le paramètre bootstrap active le principe du bagging, qui constitue le fonctionnement fondamental des forêts aléatoires. Chaque arbre est entraîné sur un échantillon aléatoire du jeu de données d'apprentissage, avec remise. Cela signifie que certaines observations peuvent être répétées tandis que d'autres peuvent ne pas être utilisées pour un arbre donné. Cette stratégie permet de diversifier les arbres de la forêt et de réduire la variance globale du modèle, ce qui améliore sa capacité à produire des prédictions fiables sur de nouvelles données.

4.2.1.2. Présentation et Interprétation des résultats sur le jeu d'entraînement

Les résultats obtenus à travers les différents folds montrent que le modèle présente un comportement globalement stable lors de la phase d'apprentissage. En utilisant la méthode de validation croisée temporelle avec **TimeSeriesSplit**, chaque fold permet d'évaluer le modèle sur une période différente tout en respectant l'ordre chronologique des données. On remarque d'abord que le temps d'entraînement augmente progressivement d'un fold à l'autre. Cette évolution est logique puisque, à chaque nouvelle itération, le modèle dispose d'un volume de données plus important pour apprendre (exemple : si le fold1 s'entraine sur 2010-2013 et prédis 2014, le fold2 lui s'entraine sue 2010-2014 et prédis 20215 et ainsi de suite). Ainsi, le temps passe d'environ **5 secondes** pour le premier fold à plus de **26 secondes** pour le dernier. Cette augmentation traduit simplement l'enrichissement progressif des données d'apprentissage. Les métriques d'erreur obtenues restent relativement constantes sur l'ensemble des folds. Les valeurs du MAE varient très peu, entre **2,46** et **2,51**, ce qui signifie que les prédictions du modèle s'écartent en moyenne d'environ deux à trois unités par rapport aux valeurs réelles des niveaux de stock. Cette faible variation montre que le modèle est d'une précision stable quelle que soit la période de validation considérée. De la même manière, les valeurs du RMSE restent proches

de **3** sur tous les folds. Comme cette métrique pénalise davantage que le MAE les erreurs importantes, ces résultats indiquent que le modèle génère peu d'écarts extrêmes entre les valeurs prédites et les valeurs réelles. Les prédictions restent donc cohérentes même lorsque certaines fluctuations apparaissent dans les données. Concernant le coefficient de détermination R^2, les scores obtenus se situent entre **0,54** et **0,57**. Cela signifie que le modèle parvient à expliquer à peu près la moitié de la variabilité des niveaux de stock observés. Même si ce résultat peut encore être amélioré, il montre déjà que le modèle réussit à capter une partie significative des tendances présentes dans les données historiques. « Grosso modo », ces résultats montrent que le modèle offre des performances satisfaisantes pour une première phase de prédiction. Toutefois, certaines améliorations restent envisageables.

Résultats par Fold des métriques pour le RF en entraînement

Source : Conception Hauteur

FOLDS	MAE	RMSE	MAPE	R2	Temps d'entrainement
Fold 1	2,51	3,10	62,85%	0,5545	5,2 secondes
Fold 2	2,46	3,05	62,21%	0,5596	10,5 secondes
Fold 3	2,47	3,05	64,98%	0,5657	15,2 secondes
Fold 4	2,51	3,09	64,17%	0,5555	21,5 secondes
Fold 5	2,47	3,05	63,60%	0,5644	26,5 secondes

4.2.1.3. Teste du modèle sur les données 2024-2025

Une fois entraîné, le modèle a été testé sur les données de la période 2024-2025. Les résultats obtenus sont accès encourageants.

APERÇU DES PRÉDICTIONS RANDOM FOREST VS RÉALITÉ

	Valeur Réelle (Test)	Prédiction Random Forest	Erreur Absolue
12012	10	12.14	2.14
12013	15	11.87	3.13
12014	18	12.20	5.80
12015	13	11.80	1.20
12016	11	11.79	0.79
12017	12	11.82	0.18
12018	8	10.67	2.67
12019	10	9.60	0.40
12020	0	0.43	0.43
12021	0	0.01	0.01

Figure 9. Prédiction RF / Données réelles

Source : Conception Hauteur

La figure ci-dessous représente les résultats du RF sur le jeu de test. Avec un coefficient de détermination (R^2) de 0,62, l'algorithme parvient à expliquer plus de 60 % des variations de stock soit plus de la moitié, ce qui témoigne d'une compréhension non négligeable des mécanismes de vente. Par jour, l'erreur moyenne MAE est de 2,3 unités. Cela signifie que si l'algorithme prévoit qu'il restera 10 unités, la réalité se situera très probablement entre 8 et 12. C'est une marge d'erreur pas trop élevé ce qui pourrai permettre d'éviter les ruptures de stock. Bien que l'erreur relative MAPE affiche 59 %, le fait que le MAE soit proche du RMSE montre que le modèle ne fait pas tout le temps des erreurs monstrueuses. Ce chiffre s'explique par les petits volumes vendus. C'est-à-dire que se tromper de deux unités sur une petite quantité de donnée représente mathématiquement un pourcentage accès élevé, sans que cela ne soit un échec total pour autant.

```
Performances sur le JEU DE TEST
 [Random Forest - Test]
   MAE  = 2.2949 unités/jour
   RMSE = 2.8364 unités/jour
   MAPE = 59.58%
   R²   = 0.6238
```

Figure 10. Test de performance du RF sur les données 2024-2025

Source : Conception Hauteur

4.2.2. Prédiction avec XGBoost

4.2.2.1. Architecture du modèle

Pour le modèle XGBoost, ces paramètres ont été définis de manière assure une bonne performance prédictive et une capacité de généralisation. L'objectif était de construire un modèle capable d'apprendre efficacement les tendances présentes dans les données sans pour autant mémoriser les particularités propres au jeu d'entraînement. Le nombre d'arbres a été fixé à 700 avec n_estimators. Ce choix permet au modèle de corriger progressivement ses erreurs au fil de l'avancement. Plus le nombre d'arbres est élevé, plus le modèle peut affiner ses prédictions. Toutefois, afin d'éviter un apprentissage trop agressif, un faible taux d'apprentissage de 0,05 fixer à l'aide du paramètre learning_rate a été utilisé. Cela signifie que chaque nouvel arbre apporte uniquement une petite correction aux erreurs précédentes.

L'apprentissage devient alors plus progressif et généralement plus stable, ce qui améliore la qualité des prédictions sur de nouvelles données. max_depth nous a permis de limiter la profondeur maximale des arbres à 7 niveaux. Cette limitation permet d'éviter la création d'arbres trop complexes qui risqueraient de reproduire fidèlement les anomalies présentes dans les données d'entraînement. Les paramètres subsample et colsample_bytree introduisent volontairement une stratégie aléatoire dans l'apprentissage. À chaque itération, le modèle utilise seulement 80 % des observations et 80 % des variables disponibles. Cette stratégie favorise la diversité des arbres construits et réduit le risque de surapprentissage. Elle permet également d'obtenir un modèle plus robuste face aux variations des données. Des mécanismes de régularisation ont également été intégrés afin de contrôler la complexité du modèle. La régularisation L1, définie par reg_alpha, réduit l'influence des variables les moins pertinentes en diminuant certains poids jusqu'à zéro. Cela aide le modèle à se concentrer principalement sur les variables réellement utiles. La régularisation L2, représentée par reg_lambda, agit comme un mécanisme de stabilisation en empêchant certains coefficients de devenir excessivement élevés. Enfin, le paramètre min_child_weight impose un nombre minimal d'observations dans les feuilles terminales des arbres. Cette contrainte évite la création de feuilles trop spécifiques basées sur très peu de données, ce qui améliore la stabilité du modèle. Dans l'ensemble, ces réglages ont été choisis afin d'obtenir un modèle XGBoost capable de produire des prédictions plus fiables et plus stables dans un contexte réel de prédiction des ventes, où les données peuvent être fortement influencées par des variations saisonnières, économiques ou comportementales.

4.2.2.2. Présentation et interprétation des résultats sur le jeu d'entraînement

Après l'évaluation du Random forest, cette seconde expérimentation permet d'observer une autre prédiction avec le XGBoost. Les résultats obtenus montrent que le modèle adopte un comportement assez régulier tout au long des différentes phases de validation temporelle avec des performances en dessous de celles du RF. L'utilisation de la méthode TimeSeriesSplit nous permet toujours de respecter l'ordre chronologique des données. À chaque fold, le modèle apprend sur beaucoup plus de données avant de réaliser ses prédictions. Cette logique explique naturellement l'augmentation progressive du temps d'entraînement observée au fil des folds. Celui-ci passe d'environ 5,8 secondes pour le premier fold à près de 12,9 secondes pour le dernier. Cette évolution était attendue. Sur le plan des performances, les résultats sont globalement cohérents et relativement stables d'un fold à l'autre. Le MAE varie entre 1,80 et 1,94, ce qui signifie que les prédictions s'écartent en moyenne d'environ deux unités des valeurs

réelles. Cette faible amplitude confirme que le modèle reste précis et ne présente pas de dérive importante selon les périodes de validation. De la même manière, le RMSE se situe entre 2,38 et 2,54. Cette proximité des valeurs montre que les erreurs importantes restent limitées et que le modèle ne produit pas de prédictions fortement éloignées. Les performances sont donc régulières, ce qui renforce la fiabilité globale du modèle sur l'ensemble des folds. Le coefficient de détermination R^2 se situe entre 0,6965 et 0,7316. Ces valeurs indiquent que le modèle parvient à expliquer environ 70 % de la variabilité des données, ce qui représente une performance satisfaisante dans un contexte de prévision de séries temporelles. On note même une légère amélioration sur certains folds, ce qui suggère une bonne capacité d'adaptation du modèle aux différentes périodes d'apprentissage. Enfin, le MAPE varie entre 53,25 % et 61,51 %. Ce taux d'erreur reste relativement élevé. Cela peut s'expliquer par la variabilité naturelle des données. Dans l'ensemble, ces résultats montrent que le modèle offre des performances globalement solides et cohérentes, avec une bonne stabilité entre les folds et une capacité explicative correcte. Comparé au RF, il présente des performances beaucoup plus satisfaisantes et reste fiable pour la prédiction des niveaux de stock. Toutefois, la présence d'un MAPE encore élevé suggère qu'il existe une marge d'amélioration.

Résultats par Fold des métriques pour le XGBoost en entraînement

Source : Conception Hauteur

FOLDS	MAE	RMSE	MAPE	R2	Temps d'entrainement
Fold 1	1,93	2,53	54,63 %	0,6965	5,8 secondes
Fold 2	1,80	2,38	53,25 %	0,7316	7,7 secondes
Fold 3	1,84	2,44	56,51 %	0,7230	9,1 secondes
Fold 4	1,94	2,54	61,51 %	0,7004	8,7 secondes
Fold 5	1,89	2,48	57,46 %	0,7125	12,9 secondes

4.2.2.3. Test du modèle sur les données 2024-2025

Une fois entraîné, le XGBoost aussi a été testé sur les données de la période 2024-2025. Voici les résultats obtenus :

APERÇU DES PRÉDICTIONS VS RÉALITÉ

	Valeur Réelle (y_test)	Prédiction XGBoost	Erreur Absolue
12012	10	12.08	2.08
12013	15	14.73	0.27
12014	18	17.50	0.50
12015	13	12.52	0.48
12016	11	11.16	0.16
12017	12	12.13	0.13
12018	8	7.66	0.34
12019	10	10.13	0.13
12020	0	0.28	0.28
12021	0	0.00	0.00

Figure 11. Prédiction XGB / Réalité

Source : Conception Hauteur

L'évaluation du modèle sur le jeu de test montre de très bonnes performances en matière de prévision. Avec un coefficient de détermination de 0,8236, le modèle explique plus de 82 % de la variabilité des données. Cela indique que les variables explicatives choisies sont pertinentes et que le comportement global de la série temporelle est bien capturée. Le MAE, estimée à 1,48, montre que les écarts entre les valeurs prédites et les valeurs réelles restent faibles en moyenne. La RMSE, avec une valeur de 1,94, confirme cette tendance et suggère que les erreurs sont globalement stables. En ce qui concerne le MAPE, qui atteint 44,45 %, comme on l'avait expliqué pour le RF, il convient de nuancer son interprétation. En effet, dans des contextes où les valeurs réelles sont faibles, de petites différences peuvent entraîner des pourcentages d'erreur élevés. Ce résultat reflète donc davantage la sensibilité de cet indicateur aux faibles volumes que des limites réelles du modèle. Dans l'ensemble, ces résultats confirment que XGBoost constitue un outil de prévision fiable et robuste. Il permet d'obtenir des estimations stables et cohérentes, tout en réduisant l'incertitude liée aux prévisions, ce qui en fait un outil pertinent pour l'aide à la décision.

```
Performances XGBoost sur le JEU DE TEST
  [XGBoost - Test]  MAE=1.4813  RMSE=1.9423  MAPE=44.45%  R²=0.8236
```

Figure 12. Performance de XGBoost sur le jeu de test

Source : Conception Hauteur

4.2.3. Le modèle hybride

4.2.3.1. Architecture du modèle

Dans le but d'améliorer les performances de prédiction des deux premiers modèles, une approche hybride basée sur la technique du **stacking** a été mise en place. Cette méthode consiste à combiner plusieurs algorithmes afin de profiter de leurs avantages respectifs et de réduire leurs limites individuelles. L'objectif principal de cette architecture hybride est donc d'obtenir des prédictions plus robustes, plus stables et plus précises sur les données temporelles. L'architecture retenue repose sur trois niveaux complémentaires. Dans un premier temps, les deux modèles de base sont entraînés séparément sur les données historiques : le Random Forest et le XGBoost. Une fois ces deux modèles entraînés, leurs prédictions ne sont pas directement utilisées comme résultats finaux. Elles servent plutôt de nouvelles variables d'entrée pour un troisième modèle appelé méta-modèle. Dans notre mémoire, le méta-modèle que l'on a choisi est une régression **Ridge**, c'est une régression linéaire régularisée utilisant une pénalisation L2. Le rôle de ce modèle est d'apprendre comment combiner intelligemment les prédictions issues du Random Forest et du XGBoost afin de produire une estimation finale plus performante. Afin de respecter la nature chronologique des données, la validation croisée temporelle (TimeSeriesSplit) a été utilisée aussi pour construction du modèle hybride. Les prédictions produites sur les différentes périodes de validation sont appelées prédictions Out-Of-Fold (OOF) et servent à entraîner le méta modèle Ridge.

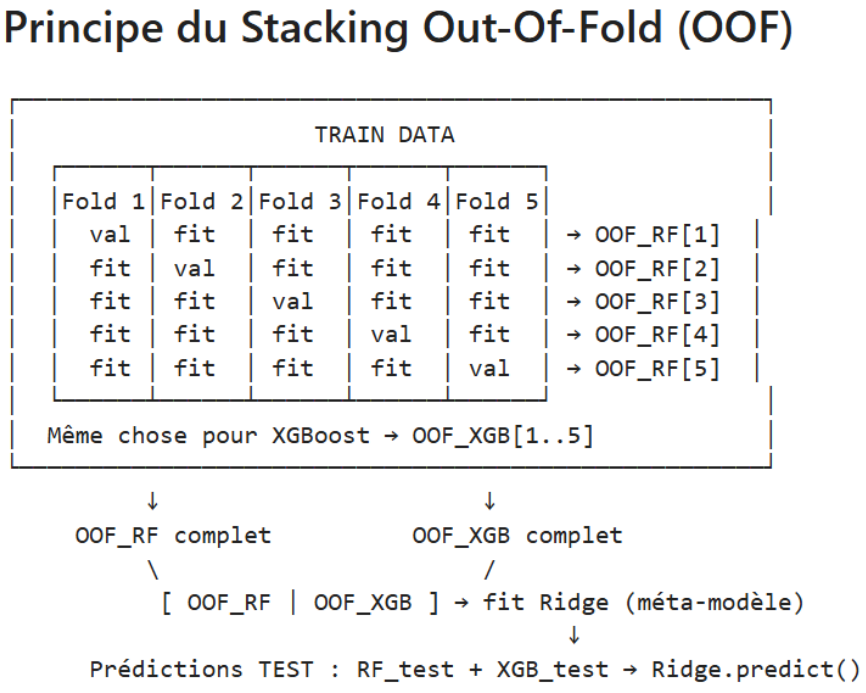


Figure 13. Schématisation du Out-Of-Fold

Source : Conception Hauteur

Les premières analyses réalisées sur les modèles de base montrent déjà une différence notable entre Random Forest et XGBoost. Les résultats montrent clairement que XGBoost capte mieux les dynamiques complexes et les variations présentes dans les séries temporelles de ventes.

```
=== Comparaison Random Forest vs XGBoost ===
Métrique   | Random Forest   | XGBoost
-------------------------------------------
MAE        |          2.2949 |          1.4813
RMSE       |          2.8364 |          1.9423
MAPE       |         59.5800 |         44.4500
R2         |          0.6238 |          0.8236
```

Figure 14. Comparaison test du RF et XGB

Source : Conception Hauteur

L'analyse des prédictions OOF confirme cette tendance. Le score MAE obtenu par le Random Forest sur les données OOF est de 2,6327 contre 2,1394 pour XGBoost. Cela signifie que, même dans les phases de validation croisée temporelle, XGBoost reste le modèle le plus performant parmi les apprenants de base. Cette différence explique naturellement pourquoi le méta modèle Ridge accorde davantage d'importance aux prédictions de XGBoost lors de la combinaison finale. L'étude des coefficients du méta-modèle constitue également un élément très intéressant pour comprendre le fonctionnement interne de l'architecture hybride. Les résultats obtenus montrent un coefficient négatif pour le Random Forest (-0,2923) et un coefficient positif beaucoup plus élevé pour XGBoost (1,2647). Cela signifie que le modèle hybride se base principalement sur les prédictions de XGBoost pour construire ses estimations finales. En pratique, le Random Forest apporte peu d'informations supplémentaires utiles par rapport à XGBoost. Le modèle Ridge apprend donc automatiquement à réduire son influence afin de privilégier le modèle le plus fiable. Le graphique représentant les poids du méta-modèle illustre clairement cette situation. La barre rouge qui représente XGBoost est largement dominante, traduisant sa contribution majeure dans la prédiction finale. À l'inverse, le poids négatif attribué au Random Forest représenté par la barre bleue montre que certaines de ses prédictions introduisent davantage d'erreurs ou d'informations redondantes. Le méta-modèle ajuste donc automatiquement les contributions de chaque algorithme afin d'obtenir la combinaison la plus performante possible.

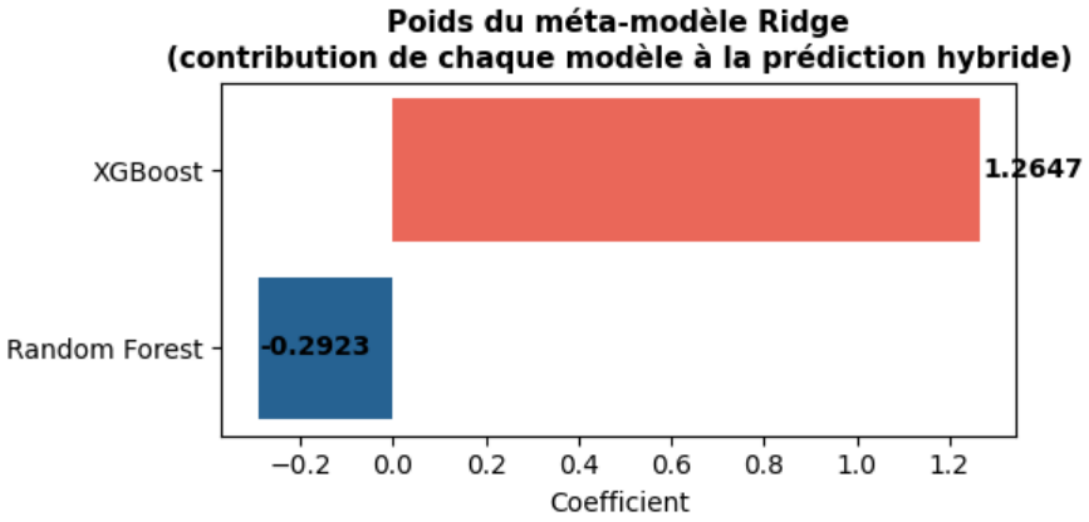


Figure 15. Contribution de chaque modèle à la prédiction hybride

Source : Conception Hauteur

4.2.3.2. Test du modèle hybride sur les données 2024-2025

Après l'entraînement du modèle hybride, les performances finales ont été évaluées sur le jeu de test couvrant la période 2024-2025. Deux variantes ont été comparées : une version NNLS imposant des coefficients positifs et une version Ridge classique. Les résultats montrent que le modèle Ridge reste légèrement plus performant. Sur le jeu de test final, le modèle hybride Ridge obtient un MAE de 1,4546, un RMSE de 1,8814, un MAPE de 42,81% ainsi qu'un coefficient R^2 de 0,8345. Ces résultats sont meilleurs que ceux obtenus individuellement par Random Forest et légèrement supérieurs à ceux du modèle XGBoost seul. Cette amélioration, bien que modérée, confirme l'intérêt du stacking dans ce cadre. Le MAE inférieur à 2 signifie que, globalement, les prédictions du modèle hybride s'écartent en moyenne de moins de deux unités par rapport aux valeurs réelles observées. Cette précision est particulièrement satisfaisante compte tenu des fluctuations importantes pouvant exister dans les données de vente quotidiennes. Le RMSE de 2 unités montre également que les erreurs importantes restent relativement limitées. Comme cette métrique pénalise davantage les grandes erreurs de prédiction, cette faible valeur indique que le modèle hybride reste globalement stable même dans des situations complexes. Le MAPE de 42,81% peut sembler relativement élevé, mais cette situation est fréquente dans les séries temporelles comportant de nombreuses petites valeurs ou des jours avec de faibles quantités vendues. Dans ce contexte, une légère différence entre la valeur réelle et la valeur prédite peut provoquer une forte variation en pourcentage. Malgré cela, les autres métriques montrent que les prédictions restent cohérentes et fiables dans l'ensemble. Le coefficient de détermination R^2 de 0,8345 confirme enfin la bonne capacité

explicative du modèle hybride. Cela signifie que près de 83 % de la variabilité des quantités vendues sont correctement expliquée par le système de prédiction mis en place. Cette performance montre que l'architecture hybride parvient à capter une grande partie des comportements présents dans les données historiques.

```
=== COMPARAISON FINALE DES MODÈLES ===
Métrique    | Random Forest    | XGBoost          | Hybride
------------------------------------------------------------
```

Modèle	MAE	RMSE	MAPE	R^2
Random Forest	2.2949	2.8364	59.5781	0.6238
XGBoost	1.4813	1.9423	44.4542	0.8236
Hybride	1.4546	1.8814	42.8075	0.8345

Figure 16. Résultat par métrique des trois modèles

Source : Conception Hauteur

4.2.4. Interprétation des graphiques

Au cours de ces apprentissages, nous avons généré des graphiques pour la visualisation de certain cas :

- Degré d'importance des variables pour la prédiction RF

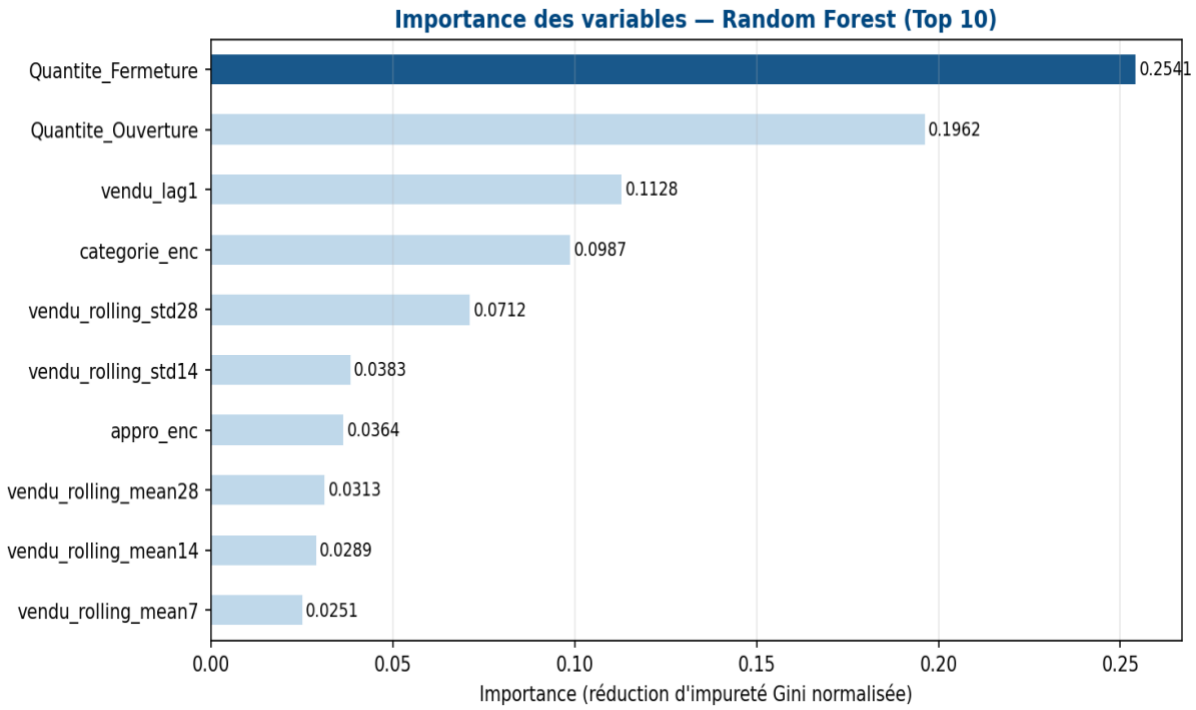


Figure 17. Degrés d'importance des variables pour la prédiction RF

Source : Conception Hauteur

L'analyse du graphique d'importance des variables ci-dessus permet de mieux comprendre comment le modèle Random Forest construit ses prédictions, en mettant en exposant les dix caractéristiques les plus influentes. On observe tout d'abord une domination très nette des variables liées directement au stock. La quantité de fermeture apparaît comme le facteur le plus déterminant, avec un score de 0,2541, suivie de près par la quantité d'ouverture (0,1962) sur une échelle de [0,0 à 0,2]. À elles seules, ces deux variables concentrent près de la moitié de l'importance totale. Cela reste cohérent dans le sens ou le niveau de stock disponible constitue l'information la plus immédiate pour expliquer les ventes, surtout dans une logique de prévision. Le décalage d'un jour vendu_lag1, avec une valeur de 0,1128, confirme que les ventes d'un jour dépendent fortement de celles de la veille. Au-delà de ces trois variables, le modèle intègre également des éléments plus contextuels. La variable liée à la catégorie du produit, après encodage, arrive en quatrième position avec un poids de 0,0987. Cela montre clairement que le comportement des ventes n'est pas fixe et varient selon le type de produit. De son côté, le statut d'approvisionnement appro_enc, bien que moins influent (0,0417), apporte une information complémentaire utile pour affiner certaines décisions. La suite du classement met en évidence l'importance des variables temporelles.

- Degré d'importance des variables pour la prédiction XGB

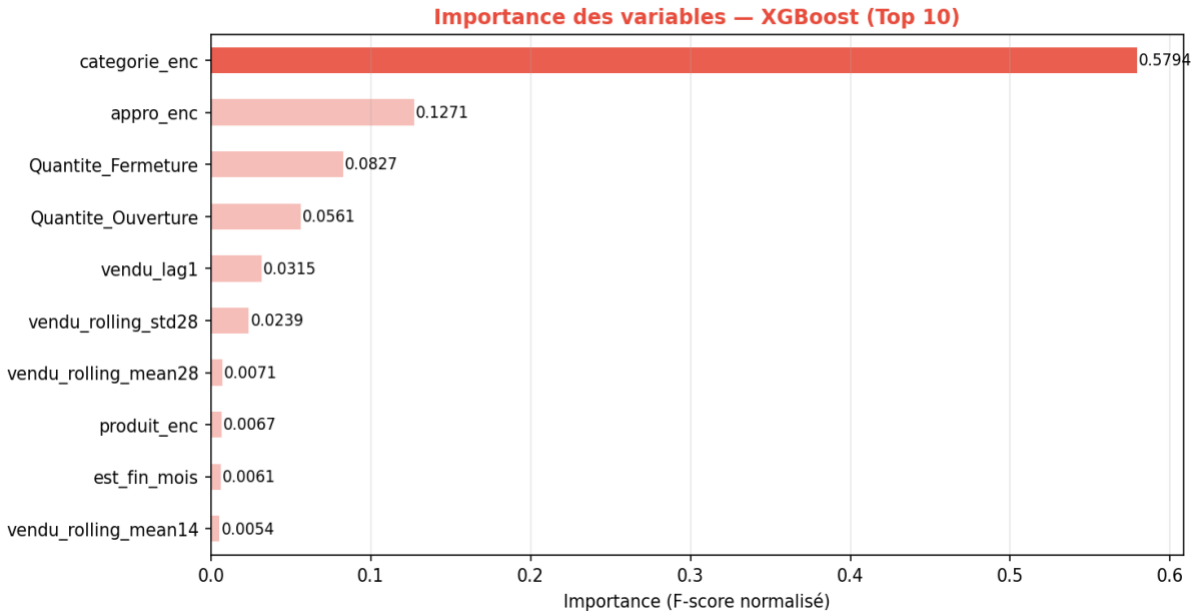


Figure 18. Degrés d'importance des variables pour la prédiction XGB

Source : Conception Hauteur

L'analyse du graphique d'importance des variables du modèle XGBoost ci-dessus met en lumière la structure des facteurs influençant les prédictions. On remarque tout de suite que toutes les variables ne jouent pas le même rôle, certaines se démarquant très nettement des autres. En particulier, la variable categorie_enc qui ressort comme étant de loin la plus déterminante, avec un poids de 0,5794 sur une échelle de [0,0 à 0,600], suivi de la variable appro_enc avec 0,1271. Cela signifie concrètement que le modèle accorde une importance majeure à la nature du produit. Autrement dit, ce ne sont pas uniquement les quantités ou les historiques qui guident la prédiction, mais aussi le type même du produit (ce qui traduit des comportements de vente différents selon les catégories). Cette dominance montre que XGBoost capte bien la structure globale des données. Dans un second temps, on retrouve les variables liées aux niveaux de stock, notamment Quantite_Fermeture (0,0827) et Quantite_Ouverture (0,0561). Leur présence parmi les variables importantes reste logique, puisque la disponibilité des produits peut aussi influencer les ventes. Toutefois, leur poids reste nettement inférieur à celui de la catégorie, ce qui suggère que, contrairement au modèle Random Forest, XGBoost ne se base pas principalement sur les quantités brutes, mais davantage sur le contexte. Par ailleurs, la variable vendu_lag1 (0,0315) confirme l'importance de la dimension temporelle. Le modèle tient compte des ventes récentes pour anticiper celles du lendemain, ce qui traduit une certaine continuité dans le comportement des données. Cette dépendance à court terme est typique des séries temporelles et montre que les variables de décalage sont pertinentes. Les variables comme vendu_rolling_std28 et vendu_rolling_mean28 apportent, quant à elles, une contribution intermédiaire mais non négligeable. Elles permettent au modèle de mieux intégrer des éléments liés à la variabilité de la demande. Même si leur impact est plus modéré, elles enrichissent la compréhension globale du phénomène. Enfin, les variables restantes, interviennent de manière plus discrète. Leur rôle est davantage d'ajuster les prédictions que de les orienter directement.

- Courbe de convergence RMSE de XGB

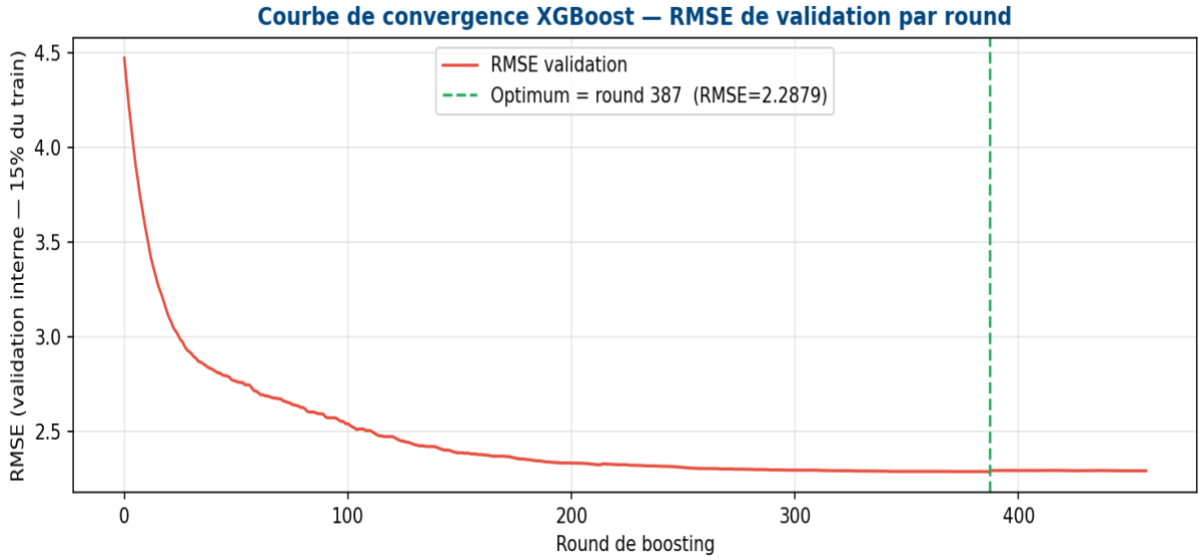


Figure 19. Courbe de convergence du RMSE avec XGBoost

Source : Conception Hauteur

La figure ci-dessus représente l'évolution de l'erreur quadratique moyenne (RMSE) du modèle XGBoost en fonction des étapes de boosting (rounds). Elle permet d'analyser le comportement du modèle au cours de son apprentissage et d'évaluer sa capacité de stabilisation. Au début de l'entraînement, on observe une diminution très rapide du RMSE. En effet il passe d'environ 4,5 à près de 3 en seulement quelques d'itérations. Cette phase traduit un apprentissage initial efficace, où le modèle capture rapidement les structures principales des données et corrige les erreurs les plus importantes. Par la suite, la courbe continue de décroître mais de manière plus progressive. Entre environ 150 et 350-400 rounds, la diminution du RMSE devient plus lente. Cela indique que le modèle entre dans une phase de perfectionnement, où il ajuste progressivement ses prédictions pour améliorer sa précision.

Les gains deviennent plus faibles, mais restent réguliers. À partir d'environ 400 rounds, la courbe se stabilise. L'amélioration du RMSE devient faible, ce qui signifie que le modèle a pratiquement atteint son niveau optimal de performance. Autrement dit, le RMSE minimum est atteint autour du round 387 avec une valeur d'environ 2,2879, comme indiqué par la ligne verticale verte en pointillés. L'absence de remontée du RMSE sur le jeu de test est un point particulièrement important.

Elle montre que le modèle ne souffre pas de surapprentissage (overfitting) dans cet apprentissage. En effet, si c'était le cas, on observerait une augmentation de l'erreur après un

certain nombre d'étapes. Ici, la courbe reste décroissante jusqu'à la fin, ce qui indique une bonne généralisation.

- Courbe de convergence du RMSE du modèle hybride

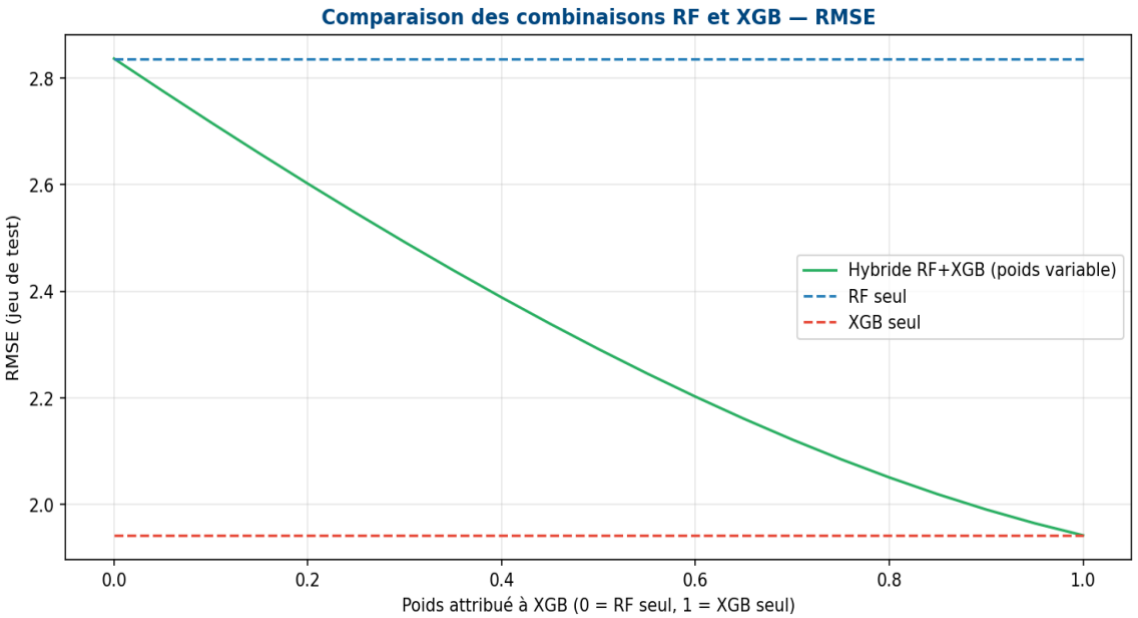


Figure 20. Courbe du RMSE avec le modèle hybride

Source : Conception Hauteur

Dans ce graphique nous avons une représentation de la courbe du RMSE selon le poids donné à XGB ou RF. La ligne bleu en pointillé représente la métrique quand le modèle hybride s'appuie seulement du RF, ainsi on remarque que l'erreur est assez élevée, jusqu'à presque 3 points. C'est le contraire pour l'utilisation de XGB seul représenté par la ligne en couleur rouge qui reste bas (-2 points) et constant montrant que XGB est plus performant.

Ce que confirme la ligne verte qui représente l'hybridation selon le poids donné à XGB de 0 à 1. Ainsi donc la courbe descend progressivement, ce qui signifie que plus on donne de poids à XGB plus l'erreur diminue.

- Performance du modèle hybride par catégorie

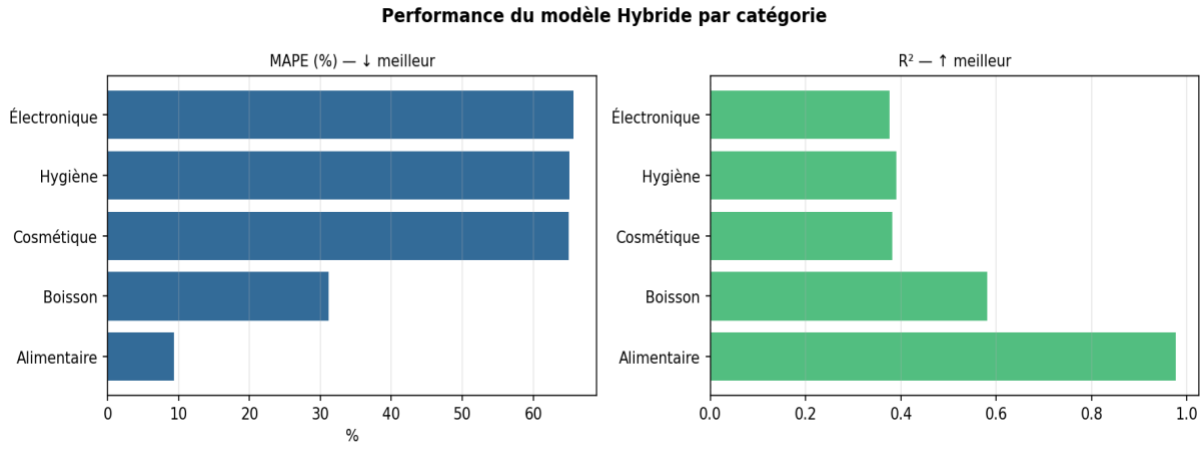


Figure 21. Performance du modèle hybride par catégorie

Source : Conception Hauteur

Dans le graphique ci-dessus nous pouvons constater le degré de performance du modèle hybride pour chaque catégorie. Ainsi on voit clairement que la catégorie alimentaire est de loin la mieux expliquée par l'algorithme avec un MAPE de -10% et un R^2 qui s'approche des 100%. La catégorie boisson est la suivante avec une explication de l'algorithme de presque de 60% et une erreur MAPE d'environ 31%. Ensuite vient la catégorie hygiène avec un R^2 de près de 40% qui avec les catégories électronique et cosmétique restent très difficile à prédire de par leur consommation souvent très irrégulière.

- Prédiction de la farine par les algorithmes VS réel

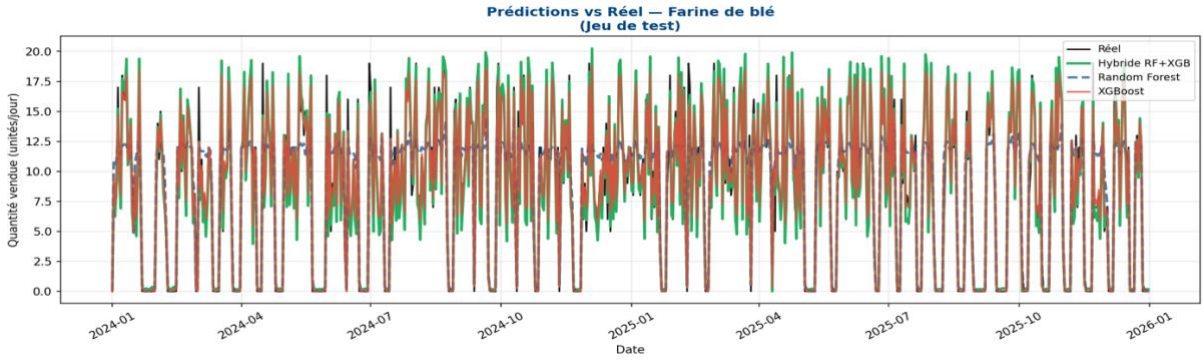


Figure 22. Prédiction VS Réalité de Farine de blé sur les données de test

Source : Conception Hauteur

La figure ci-dessus est la représentation graphique entre les ventes réelles et les prévisions sur la farine de blé. Face à une demande réelle en noir très instable allant de 0 à 14 unités, le modèle RF en bleu pointillé paraît bien loin de la réalité. Sa trajectoire qui tourne autour de 8,5 unités ne suit pas bien les fluctuations quotidiennes, ce qui explique sa mise à l'écart par le méta

modèle. À l'inverse, XGBoost en orange fait preuve d'une excellente réactivité en épousant fidèlement le rythme en dents de scie de la série. Cette dynamique se transmet directement au modèle Hybride en vert, dont le tracé se superpose presque parfaitement à celui de XGBoost. Ces résultats s'expliquent du fait que les produits alimentaires sont expliqués à 97% par le modèle hybride.

- Prédiction de coca cola par les algorithmes VS le réel

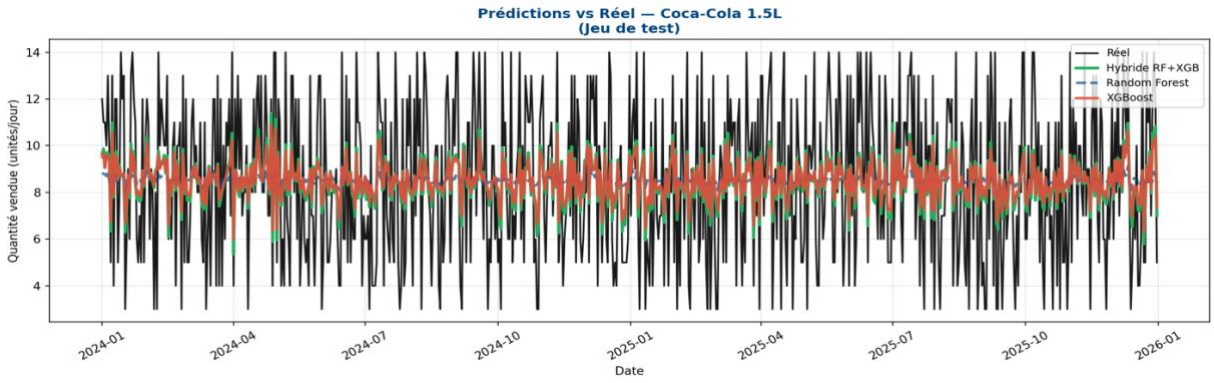


Figure 23. Prédiction VS Réalité de coca-cola 1,5L sur les données de test

Source : Conception Hauteur

L'algorithme RF en bleu pointillé adopte une approche très prudente. Sa courbe est presque droite. Il capte la tendance moyenne, mais il ne suit pas du toutes les variations quotidiennes. Il est donc stable mais peu réactif. XGB en rouge est beaucoup plus dynamique, sa courbe suit bien mieux les fluctuations des ventes, même si elle reste bien loin de la réalité. Il arrive à capter une partie des hausses et des baisses, ce qui montre une meilleure capacité d'adaptation. Le modèle hybride en vert se comporte à peu près comme XGB. Sa courbe se superpose presque à celle de celui-ci avec de légères différences. Cela confirme toujours que le modèle hybride donne plus de poids à XGB. Au final, aucun modèle ne peut reproduire parfaitement les pics extrêmes. C'est typique d'un produit comme le Coca-Cola 1,5L, où la demande peut changer rapidement selon de nombreux facteurs. Mais l'approche hybride reste la plus équilibrée.

- Création d'une Baseline de 7 jours VS les autres algorithmes

Dans notre projet de prédiction, pour ne pas faire de manière isolée l'évaluation de la performance de nos algorithmes et pour démontrer leurs valeurs ajoutées, nous avons établis un point de comparaison initial appelé **Baseline** comme modèle de référence. La Baseline représente la solution la plus simple, intuitive ou traditionnelle pour résoudre le problème posé (la prédiction). En prédiction de séries temporelles, elle repose généralement sur une approche

naïve, comme l'application d'une moyenne glissante sur les derniers jours observés ou la reproduction à l'identique de la valeur de la veille. Ce modèle de référence ne nécessite aucun apprentissage ni de calcul avancé. L'intérêt de la Baseline est d'une part, pour savoir si un modèle de machine-learning parviendrait à surpasser une règle empirique simple. Mais aussi elle permet de quantifier précisément le gain de performance en mesurant l'écart entre les erreurs de cette approche naïve et celles des algorithmes.

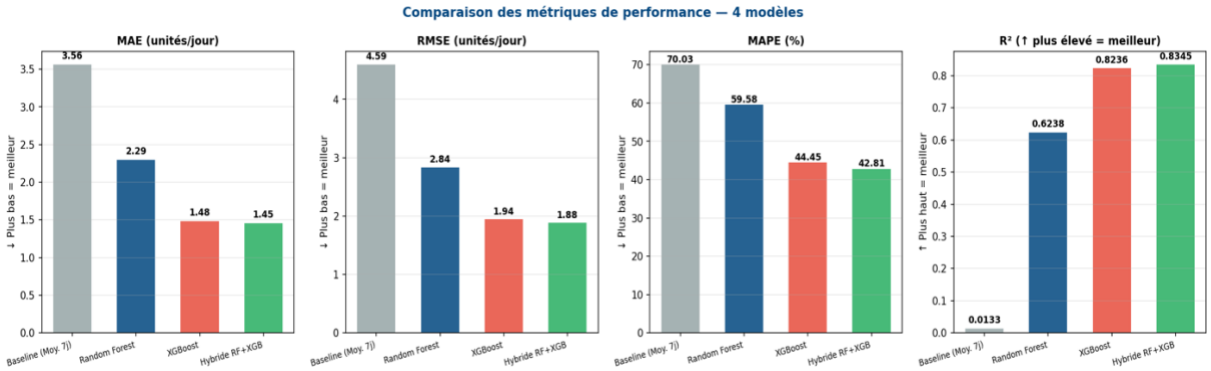


Figure 24. Comparaison de la Baseline et des algorithmes par métriques

Source : Conception Hauteur

L'analyse comparative des performances sur le jeu de test dont expose la figure ci-dessus met en évidence une hiérarchie claire et constante entre les quatre approches évaluées, validant de manière chiffrée l'apport du RF, de XGB et de l'approche hybride. En guise de point de départ, la Baseline naïve, calculée à partir de la moyenne glissante des sept derniers jours, présente les limites logiques d'une approche purement statistique. Elle enregistre une erreur moyenne absolue MAE de 3,56 unités par jour, une RMSE de 4,59 unités et un pourcentage d'erreur moyen MAPE de 70,03 %. Ces résultats initiaux confirment que les règles empiriques ou les méthodes de gestion traditionnelles peinent à absorber le bruit et les fluctuations quotidiennes des ventes d'où l'intérêt de l'utilisation du machine learning.

4.3. Recommandations

4.3.1. Le Stock de Sécurité et le Pont de Réapprovisionnement

Pour faire face aux aléas du marché, qu'il s'agisse de variations imprévues de la demande ou de retards de livraison des fournisseurs, un calcul automatisé du stock de sécurité a été modélisé pour chaque catégorie de la boutique. Ce calcul repose sur la formule mathématique suivante :

$$SS = Z_{cat} \times \sigma'_{pred} \times \sqrt{L}$$

Où Z_{cat} représente le coefficient de sécurité par catégorie, le σ'_{pred} l'écart-type des prédictions fournis par l'algorithme pour mesurer l'instabilité de la demande et **L** correspond au délai de

livraison du fournisseur. Cette approche permet d'ajuster stratégiquement le niveau de réserve (NB : plus un produit présente des ventes volatiles, plus sa réserve de sécurité est automatiquement rehaussée). Nous avons aussi introduit le concept de Point de Réapprovisionnement (**ROP**). C'est un seuil critique qui déclenche automatiquement une alerte de commande dès que le stock disponible franchit une limite plafonnée. Sa formule est la suivante :

$$\mathbf{ROP = \mu_{pred} \times L + SS}$$

Avec μ_{pred} la demande moyenne quotidienne prédite par le modèle hybride et le délai de livraison L, le système évalue précisément la quantité de produits qui sera écoulée durant la phase de transit, avant d'y ajouter le stock de sécurité **SS**. Le gestionnaire dispose ainsi d'un tableau de bord dynamique indiquant, jour après jour, quels produits doivent être impérativement commandées et en quelles quantités. Ainsi nous avons fait un test sur les données de 2024-2025. Le nombre de rupture dans les dans ces données a été calculé et donne **13640** sur **14620** opérations soit un taux de **93,30%** de rupture. Après la prédiction du modèle hybride et les calculs du SS et ROP, le taux de rupture simulé est égal à **74,79%** soit une réduction de **19,8%.**

```
Ruptures dans le test : 13640
Ruptures évitées (alertes déclenchées) : 2706
Taux de rupture simulé après ML : 74.79%

Réduction du taux de rupture : 93.30% → 74.79% (19.8% de réduction)
```

Figure 25. Réduction du taux de rupture après ML

Source : Conception Hauteur

Bien que l'architecture actuelle offre une base décisionnelle hautement fiable, plusieurs axes d'amélioration technologique restent envisagés pour maximiser les performances du système :

- **Intégration de plus d'événements externes** : l'incorporation de données externes (météo, calendriers promotionnels, événements locaux) permettrait d'affiner la sensibilité du modèle sur les catégories complexes comme l'Électronique, l'Hygiène ou la Cosmétique ;
- **Prédiction spécifique** : remplacer le modèle global actuel par des algorithmes indépendants et entraînés spécifiquement par catégorie de produits afin de mieux capter les dynamiques de consommation propres à chaque type d'articles ;

- **Déploiement** : l'implémentation future d'une interface applicative connectée directement au logiciel de caisse permettra de recalculer les indicateurs **SS** et **ROP** en temps réel, automatisant ainsi le passage des commandes auprès des fournisseurs.

4.3.2. Présentation de la plateforme

Afin de rendre les résultats du modèle de prévision directement exploitables pour notre présentation, une interface web a été développée à l'aide du framework Streamlit. Cette plateforme permet de charger des données de stock, de générer des prévisions de ventes et d'obtenir des recommandations d'approvisionnement. Ainsi les différentes fonctionnalités de l'application sont présentées dans les sections suivantes afin de décrire leurs utilités et leur contribution pour la prise de décision logistique.

4.3.2.1. La page d'accueil

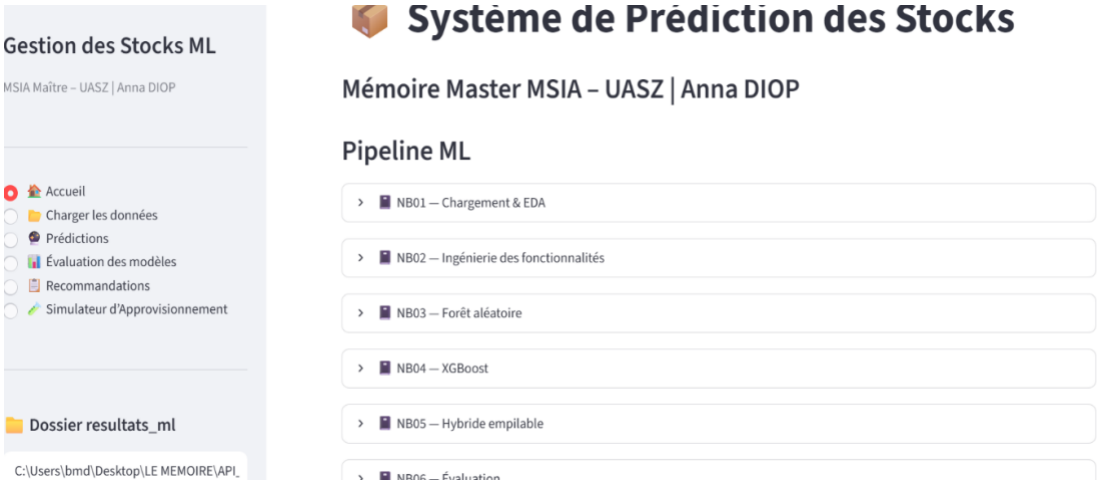


Figure 26. Page d'accueil de la plateforme de présentation

Source : Conception Hauteur

La page d'accueil constitue le point d'entrée de la plateforme. On y retrouve les informations relatives aux notebook (traitement de données, entrainement des algorithmes etc.) qui sont les sources de base qui permettent l'exécution des opérations. Cette interface fournit également un accès aux différentes fonctionnalités du système via le menu latéral de navigation. L'utilisateur peut ainsi charger des données, générer des prédictions, consulter les performances des modèles, obtenir des recommandations d'approvisionnement et réaliser des simulations de réapprovisionnement.

4.3.2.2. Chargement des données

Figure 27. Interface chargement de donnée

Source : Conception Hauteur

Cette interface permet d'importer un fichier de données (le fichier d'entraînement ou de test) en format CSV contenant les informations relatives aux produits et aux ventes. Après le chargement du fichier, la plateforme affiche un aperçu des données afin de vérifier leur conformité. Les principales informations telles que le nombre de ligne et de colonne, la période couverte et le nombre de produits distincts sont également présentées ainsi qu'un aperçu sur les données pour vérifier la conformité. La plateforme exécute ensuite automatiquement une étape de préparation des données comprenant la création de variables temporelles, la génération des retards temporels (lags), le calcul des statistiques glissantes (rolling), l'encodage des variables catégorielles. Cette étape permet de transformer les données brutes en variables exploitables par les modèles de prédiction.

4.3.2.3. Prédiction

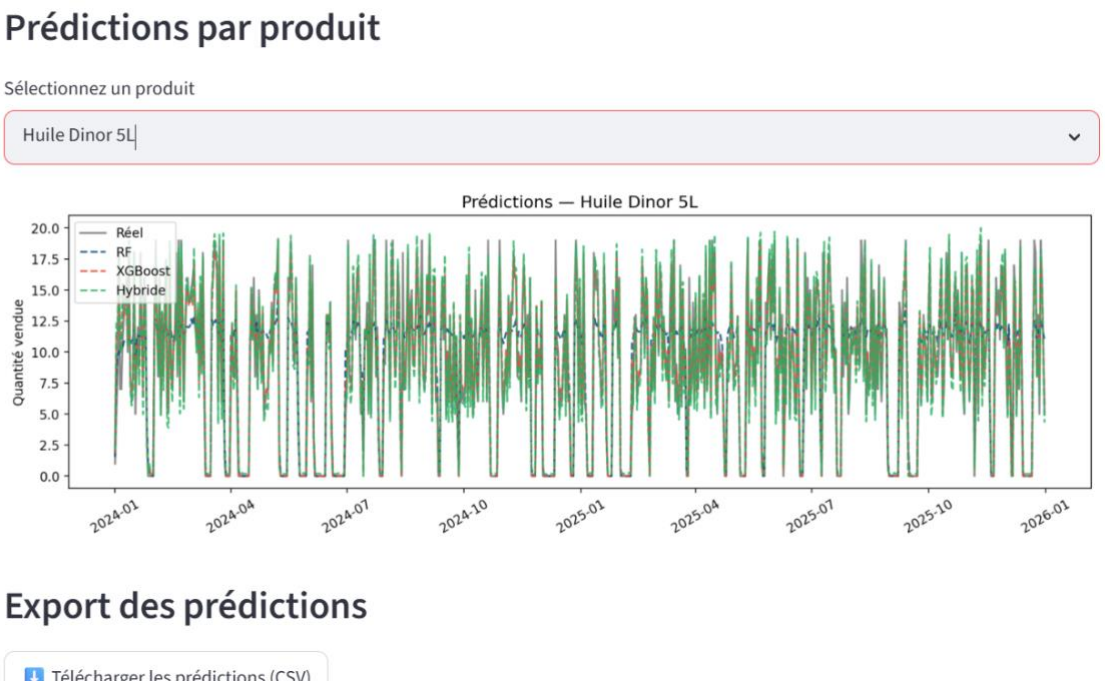
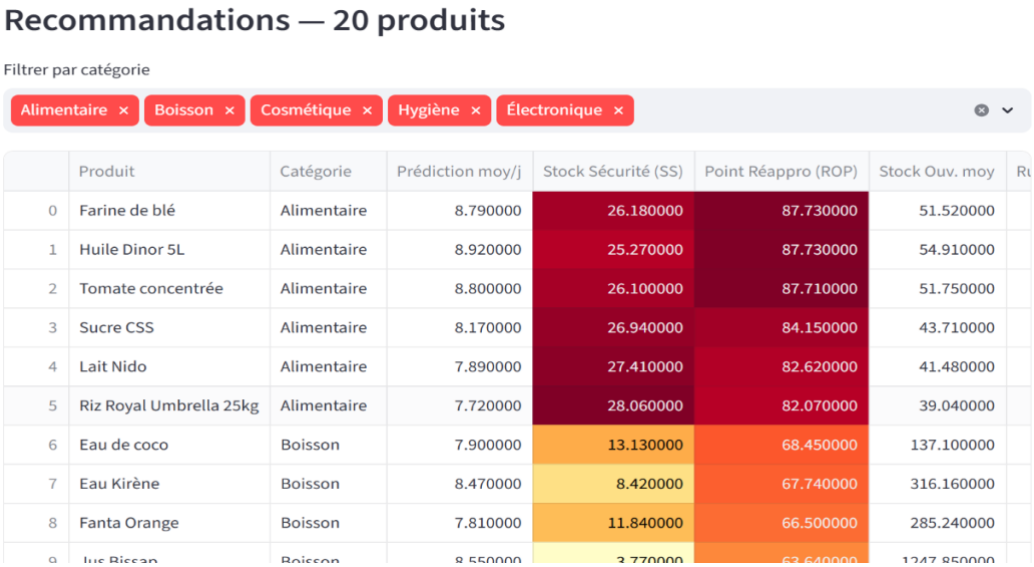


Figure 28. Aperçu onglet prédiction

Source : Conception Hauteur

Cette section permet de générer les prédictions de ventes à partir des données chargées précédemment. Une fois la sélection du produit effectué, on peut sélectionner un produit et visualiser l'évolution des prédictions générées par le modèle hybride. Les résultats sont présentés sous forme graphique afin de faciliter leur interprétation. Il compare les valeurs observées aux valeurs prédites afin d'évaluer visuellement la qualité des prévisions ce constitue la base des décisions de réapprovisionnement. La plateforme offre également la possibilité d'exporter les résultats sous forme de fichier CSV pour une exploitation ultérieure.

4.3.2.4. Recommandation



	Produit	Catégorie	Prédiction moy/j	Stock Sécurité (SS)	Point Réappro (ROP)	Stock Ouv. moy
0	Farine de blé	Alimentaire	8.790000	26.180000	87.730000	51.520000
1	Huile Dinor 5L	Alimentaire	8.920000	25.270000	87.730000	54.910000
2	Tomate concentrée	Alimentaire	8.800000	26.100000	87.710000	51.750000
3	Sucre CSS	Alimentaire	8.170000	26.940000	84.150000	43.710000
4	Lait Nido	Alimentaire	7.890000	27.410000	82.620000	41.480000
5	Riz Royal Umbrella 25kg	Alimentaire	7.720000	28.060000	82.070000	39.040000
6	Eau de coco	Boisson	7.900000	13.130000	68.450000	137.100000
7	Eau Kirène	Boisson	8.470000	8.420000	67.740000	316.160000
8	Fanta Orange	Boisson	7.810000	11.840000	66.500000	285.240000
9	Jus Bissap	Boisson	8.550000	3.770000	63.640000	1247.850000

Figure 29. Aperçu sur l'onglet recommandation

Source : Conception Hauteur

Cette interface transforme les prédictions de ventes en recommandations opérationnelles directement exploitables. Pour chaque produit, le système estime le niveau de stock à maintenir afin de limiter les risques de rupture tout en évitant un sur-stockage excessif. Les recommandations fournies permettent ainsi d'améliorer la disponibilité des produits et de faciliter la planification des commandes. Les résultats sont présentés sous forme de tableau synthétique facilitant la prise de décision. Cette fonctionnalité constitue l'un des principaux apports de la plateforme puisqu'elle traduit les résultats du modèle en actions concrètes d'aide à la décision.

4.3.2.5. Simulateur d'approvisionnement

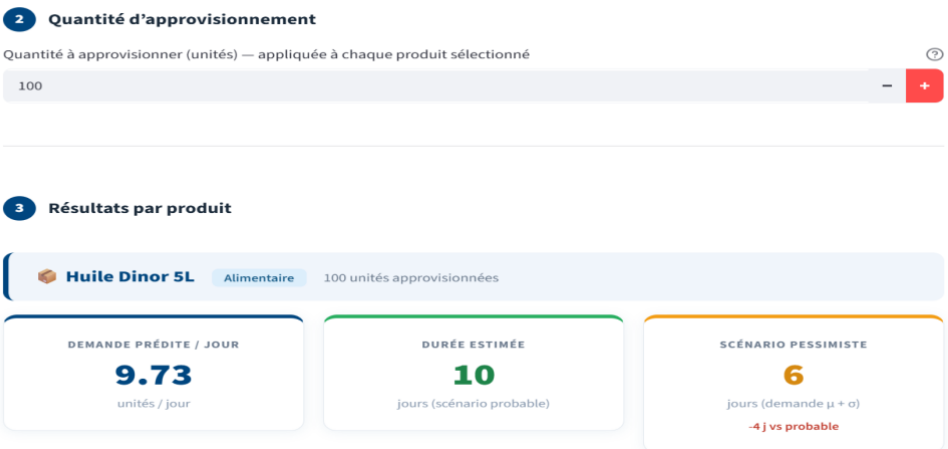


Figure 30. Aperçu onglet simulateur d'approvisionnement

Source : Conception Hauteur

Le simulateur d'approvisionnement constitue un outil d'aide à la prise de décision permettant d'évaluer différents scénarios de réapprovisionnement. On sélectionne un produit ainsi que la quantité dont nous disposons par exemple, ainsi, à partir des prédictions générées par le modèle hybride, la plateforme estime alors la durée pendant laquelle ce stock pourra couvrir la demande future et fournit des informations utiles sur la date optimale pour la planification des commandes et les éventuels risques de rupture. Cette fonctionnalité met en évidence les besoins futurs et facilite la planification des opérations d'approvisionnement.

Figure 31. Simulation d'approvisionnement Eau de Javel la croix

Source : Conception Hauteur

Ainsi sur la figure ci-dessus on voit une simulation du produit « Eau de javel la croix » pour 50 unités. Le modèle prédit une vente de 5,08 unités par jour et donc estime que cette quantité en stock couvrirai au mieux 10 jours et au pire 7 jours. De ce fait, contre tenu que le délai d'approvisionnement est estimé à 7 jours, il est recommandé de passer la commande du produit dans 3 jours pour éviter une rupture.

Figure 32. Simulation d'approvisionnement Fanta Orange

Source : Conception Hauteur

Il en va de même que la précédente pour la figure ci-dessus avec le produit « Fanta Orange » pour 105 unités. Par jour, le modèle prédit une vente de 7,09 unités ce qui fait que le stock peut tenir au mieux jusqu'à 15 jours et au pire 10 jours. Donc, toujours pour un délai d'approvisionnement de 7 jours, la date limite de commande est dans 7 jours.

4.3.3. Étude pratique de la viabilité financière du projet

Pour évaluer concrètement la viabilité du projet, nous avons jugé nécessaire de faire une estimation basée sur des projections réaliste pour une durée de vie de 5 ans. Ainsi le projet nécessitera : un NINEA et registre de commerce pour 75000f, un Site Web estimé au prix de 75000f, des cautions pour le loyer et l'électricité à 100000 et 24000, des matériels de bureau à 175000 un ordinateur portable à 200000, un disque dur à 30000, le loyer de 50000 par mois et la facture d'électricité à 12000 par mois. On prévoit 10 mois de BFR. Le projet sera financé à hauteur de 60% par fond propre et le reste par emprunt au taux d'intérêt de 9 % amortissable sur 5 ans annuités constantes. Le taux sans risque est estimé à 6% et la prime de risque moyenne exigée par les associés est de 7%. Une masse salariale de 150000f est aussi fixer. La solution est proposée sur le marché au prix de 600 000. Sur les 5 années d'exploitation, les perspectives de ventes sont :

Données des Chiffres d'affaires

Source : Conception Hauteur

année	1	2	3	4	5
nombres	3	5	7	8	10

Contre tenu de toutes ces informations, nous avons les résultats suivant :

Résultat de l'étude de la viabilité du projet

Source : Conception Hauteur

Éléments	Montants
Cout Total du projet	**2 375 000**
VAN	**1 981 117,2**
IP	**2,39**
TRI	**0,30**
CMPC	**0,1032**
DRCI	**4 ans 3 mois**

Les résultats de l'étude financière montrent que le projet est économiquement viable. En effet, la VAN est positive (1 981 117,2 F), ce qui signifie que le projet génère une valeur supplémentaire de près de 2 million. L'Indice de Profitabilité (IP) est de 2,39 indiquant que chaque franc investi génère 2,39 F de valeur. De plus, le TRI est estimé à 30 %, un taux largement supérieur au Coût Moyen Pondéré du Capital (CMPC) qui est de 10,32 %, ce qui confirme la rentabilité du projet. Enfin, le DRCI de 4 ans et 3 mois montre que l'investissement initial sera récupéré dans un délai raisonnable. Ces indicateurs permettent de conclure que le projet est financièrement rentable et mérite d'être réalisé.

4.4. Conclusion

Ce dernier chapitre a permis d'évaluer concrètement les performances des différents algorithmes en mettant en évidence leurs forces et leurs limites. Les résultats montrent clairement que les approches de machine-learning, notamment XGBoost, offre une amélioration significative avec un R^2 de 82% par rapport à une méthode de référence simple Baseline. Le méta modèle hybride Ridge basé sur le stacking confirme également son intérêt en combinant intelligemment les atouts des modèles de base pour obtenir des prédictions légèrement plus précises avec un R^2 de 83% et une erreur MAE limité seulement à 1,45 unité par jour. L'analyse des métriques et des visualisations a permis de mieux comprendre le comportement des modèles face aux données réelles des ventes, ainsi que le rôle des variables les plus influentes. L'étude par catégorie montre également que le modèle maîtrise particulièrement bien le segment alimentaire. Elle nous aide aussi à comprendre les limites mathématiques du MAPE face à des produits à faible volume comme ceux des catégories

cosmétique ou électronique. Ces résultats prouvent que notre système hybride est une solution fiable pour optimiser la gestion des stocks.

La plateforme développée constitue une interface opérationnelle permettant d'exploiter les résultats de prédiction des modèles dans un environnement simple et intuitif. Elle facilite l'analyse des données, l'estimation de la demande future et la prise de décision liée à l'approvisionnement. Grâce à l'automatisation des traitements et à la visualisation des résultats, elle nous permet de bien montré que notre algorithme parvient à prédire et donner des recommandations.

Conclusion générale

Dans ce mémoire, nous avons traité l'utilisation du machine learning dans la prédiction des niveaux de stock pour une gestion optimale des approvisionnements. En effet l'utilisation du machine learning dans la gestion des stocks constitue une avancée stratégique par rapport aux approches traditionnelles. L'objectif de ce mémoire consistait à concevoir un système de prédiction des niveaux de stock afin d'offrir aux entreprises de la grande distribution une gestion plus saine et efficace de leur approvisionnement. Pour ce faire nous avons eu à analyser et traiter **116320 lignes** d'historique de vente.

Nous avons enrichi ces données en y intégrant un calendrier qui prend en compte et renseigne toutes les fêtes sénégalaises (religieuses et civiles), des variables de retard ou encore des moyennes glissantes. Ces ajouts ont joué un rôle majeur dans la prédiction. Les algorithmes choisis pour mener à bien ce projet sont le Random Forest et le XGBoost et un méta modèle hybride Ridge.

Les résultats obtenus démontrent que les modèles de machine learning, notamment XGBoost avec un $\mathbf{R^2}$ de **82%** qui surpasse de loin le RF, offrent des performances nettement supérieures à celles de l'approche naïve de type Baseline simulée. Le modèle hybride basé sur la technique du stacking, combinant efficacement les forces des différents algorithmes a permis d'aller encore plus loin avec **83% de $\mathbf{R^2}$** tout en limitant l'erreur moyenne **(MAE)** à seulement **1,45** unité par jour. Bien que le gain de performance soit modéré par rapport à l'algorithme XGB, il reste significatif et confirme l'intérêt de cette approche dans ce projet.

Outre ces résultats, l'autre valeur ajoutée de ce projet réside dans son application pratique en reliant les prédictions directement aux calculs d'un stock de sécurité **SS** et du Point de Réapprovisionnement **ROP** afin d'adapter le niveau de sécurité aux exigences spécifiques de chaque catégorie de produits. Bien-sûr, ce système prédictif demande à évoluer. Cependant, certaines limites sont à prendre en compte, notamment en ce qui concerne la non prise en compte du météo et des promotions, mais aussi la prédiction des catégories à faible volume ou à forte variabilité, ainsi que la sensibilité de certaines métriques comme le **MAPE**. Au-delà des performances obtenues, les résultats de cette recherche confirment également la pertinence du cadre théorique retenu. En effet, notre positionnement sur la **Théorie des Contraintes** TOC se justifie par le fait que l'amélioration de la précision des prédictions contribue directement à une meilleure maîtrise des niveaux de stock, à la réduction des stocks inutiles et à la diminution des ruptures d'approvisionnement. En fournissant des estimations plus fiables de la demande, le système développé permet ainsi de soutenir les principes fondamentaux de la TOC, qui visent

simultanément à réduire les stocks, à limiter les dépenses opérationnelles et à améliorer le profil de l'entreprise. Par ailleurs, le choix du Modèle **d'Acceptation de la Technologie** TAM apparaît également pertinent dans la mesure où la solution proposée constitue un outil d'aide à la décision destiné aux gestionnaires de stocks et des approvisionnements. En offrant des prédictions automatisées, facilement exploitables et directement intégrées aux décisions de réapprovisionnement, cette solution présente une utilité perçue élevée susceptible d'améliorer les performances des utilisateurs. De plus, sa simplicité d'utilisation favorise une facilité perçue, deux déterminants essentiels de l'acceptation et de l'adoption d'une nouvelle technologie selon le TAM.

Pour l'avenir, quatre principales améliorations se distinguent. D'abord, l'intégration de nouvelles variables externes, comme la météo ou les promotions ce permettraient d'affiner les résultats sur les catégories plus irrégulières comme l'Hygiène ou l'Électronique ou encore la cosmétique. Ensuite, le passage d'un modèle de prédiction global à des modèles indépendants, entraînés spécifiquement par catégorie, offrirait une opportunité sérieuse de gagner encore en précision. Programmer un réentraînement périodique pour capturer les nouvelles tendances et maintenir la précision du modèle est aussi à envisagé. Enfin, le déploiement de cette solution sous forme d'interface applicative, un API connectée en temps réel au logiciel de gestion des approvisionnements, représenterait l'aboutissement de ce travail, transformant notre modèle d'étude en un assistant logistique pertinent.

En termes d'implication managériale notre solution pourra permettre au gestionnaire de stock et approvisionnement de mieux fidéliser sa clientèle avec une diminution des ruptures de stock mais aussi de minimiser le coût de possession des stocks et évitant le sur-stockage. En effet, l'utilisation du Machine Learning permet d'obtenir des prévisions de la demande plus précises que les méthodes traditionnelles. Par ailleurs, une meilleure maîtrise des stocks contribue à limiter les pertes liées aux produits invendus mais en même temps assure la disponibilité des produits pour le client.

Références

1. ELECDSWTROTECHNIQUE, M. E. (2024). *Gestion des stocks par l'intelligence artificielle* (Doctoral dissertation, Université Abdelhamid Ibn Badis Mostaganem).
2. Firoozi, M. (2018). Multi-echelon inventory optimization under supply and demand uncertainty (Doctoral dissertation, Université de Bordeaux).
3. Farel, J. (2017). Amélioration continue du processus d" approvisionnement: stratégies et paramétrages des approvisionnements et des stocks.
4. Arda, Y. (2008). Politiques d'approvisionnement dans les systèmes à plusieurs fournisseurs et optimisation des décisions dans les chaînes logistiques décentralisées (Doctoral dissertation, INSA de Toulouse)
5. Desport, P. (2017). Planification tactique de chaîne d'approvisionnement en boucle fermée: modélisation, résolution, évaluation (Doctoral dissertation, Université d'Angers).
6. Hubert, T. (2013). Prévision de la demande et pilotage des flux en approvisionnement lointain (Doctoral dissertation, Ecole Centrale Paris).
7. Belaid, M. S. (2022). Modélisation des processus de prévision de la chaîne logistique pour l'amélioration des performances industrielles (Doctoral dissertation, Ecole nationale supérieure Mines-Télécom Lille Douai).
8. Massonnet, G. (2013). Algorithmes d'approximation pour la gestion de stock (Doctoral dissertation, Université de Grenoble).
9. PEGUY, A. (1968). UNE MÉTHODE DE PRÉVISION ADAPTATIVE: Application aux Prévisions à court terme dance la Gestion des Stocks. Bulletin mathématique de la Société des Sciences Mathématiques de la République Socialiste de Roumanie, 12(2), 107-129.
10. Hnaien, F. (2008). Gestion des stocks dans des chaînes logistiques face aux aléas des délais d'approvisionnements (Doctoral dissertation, Ecole Nationale Supérieure des Mines de Saint-Etienne).
11. Tlili, M., Moalla, M., Bahroun, Z., & Campagne, J. P. (2010). Gestion de stocks avec transshipment dans un réseau de distribution multi sites et multi échelons. In 8e International Conference on Modeling and Simulating-mosim'Io.
12. Luce, S. (1994). Amélioration de la disponibilité des équipements de production par l'optimisation de la gestion des stocks de maintenance (Doctoral dissertation, Reims).

13. Bouami D. Le Grand Livre de La Gestion Des Stocks et Approvisionnements : Pour Une Maintenance Performante. Afnor éditions; 2019.
14. El Naqa, I., & Murphy, M. J. (2015). What is machine learning? (pp. 3-11). Springer International Publishing.
15. Mitchell, T. M., & Mitchell, T. M. (1997). Machine learning (Vol. 1, No. 9). New York: McGraw-hill.
16. Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., & Fotiadis, D. I. (2015). Machine learning applications in cancer prognosis and prediction. Computational and structural biotechnology journal, 13, 8-17.
17. Candillier, L. (2006). Contextualisation, visualisation et évaluation en apprentissage non supervisé (Doctoral dissertation, Université Charles de Gaulle-Lille III).
18. SOUADDA, L. I., BERGHOUT, Y. M., MENARI, M., & BENILLES, B. La gestion du risque des crédits d'exploitation des PME par le Machine Learning Étude de cas: La Banque CPA.
19. Lu, J., Yan, H., Chang, C., & Wang, N. (2020, June). Comparison of machine learning and deep learning approaches for decoding brain computer interface: an fNIRS study. In International Conference on Intelligent Information Processing (pp. 192-201). Cham: Springer International Publishing.
20. Belaid, M. S. (2022). Modélisation des processus de prévision de la chaîne logistique pour l'amélioration des performances industrielles (Doctoral dissertation, Ecole nationale supérieure Mines-Télécom Lille Douai).
21. Mahesh, B. (2020). Algorithmes d'apprentissage automatique - une revue. Revue internationale de la science et de la recherche (IJSR). [Internet], 9(1), 381-386.
22. Mitchell, T. M., & Mitchell, T. M. (1997). Machine learning (Vol. 1, No. 9). New York: McGraw-hill.
23. Yao, B. (2023). Walmart Sales Prediction Based on Decision Tree, Random Forest, and K Neighbors Regressor. Highlights in Business, Economics and Management.
24. Helmini, S., Jihan, N., Jayasinghe, M., & Perera, S. (2019). Prévision des ventes à l'aide de modèles de réseaux de mémoire à long terme multivariés. PeerJ Prepr., 7, e27712.
25. Patel, R., Choudhary, V., Saxena, D., & Singh, A. K. (2021, June). Review of stock prediction using machine learning techniques. In 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 840-846). IEEE.

26. Vavliakis, KN, Siailis, A., & Symeonidis, AL (2021). Optimisation des prévisions de ventes dans le commerce électronique avec les modèles ARIMA et LSTM. Conférence internationale sur les systèmes et technologies d'information Web.
27. Senihji, Khadija. (2024). L'INTELLIGENCE ARTIFICIELLE : ETATS DES LIEUX ET APPLICATIONS DANS LE DOMAINE DU MARKETING.
28. LeCun, Y. (2016). L'apprentissage profond, une révolution en intelligence artificielle. La lettre du Collège de France, (41), 13.
29. FOUAD, M., Mali, R., & Bousmah, M. (2018). Machine Learning au service de la prédiction de la demande d'énergie dans les Smart Grid. Revue Méditerranéenne des Télécommunications, 8(2).
30. Beutel, AL, & Minner, S. (2012). Planification des stocks de sécurité dans le cadre de la prévision de la demande causale. Revue internationale d'économie de la production, 140 (2), 637-645.
31. He, M., Wang, H. & Thwin, M. Une technique d'apprentissage automatique pour optimiser la prévision de la demande de charge dans les systèmes de climatisation en utilisant le modèle GRU/IASO. Sci Rep 15, 3353 (2025).
32. Douaioui, K., Oucheikh, R., Benmoussa, O., & Mabrouki, C. (2024). Modèles d'apprentissage automatique et d'apprentissage profond pour la prévision de la demande dans la gestion de la chaîne logistique : une revue critique. Applied System Innovation , 7 (5), 93.
33. Bendhi MR. Gestion des stocks : l'apprentissage automatique prédit la demande et réduit les stocks excédentaires jusqu'à 20 %. IJAIDSML [Internet]. 30 juin 2023 [cité le 15 août 2025] ; 4(2) : 67-83.
34. SARR, E. N., & Ousmane, S. A. L. L. La modération automatique des commentaires toxiques dans les réseaux sociaux au Sénégal.
35. Rigatti, SJ (2017). Forêt aléatoire. Journal de médecine d'assurance , 47 (1), 31-39.
36. Hu, J., & Szymczak, S. (2023). A review on longitudinal data analysis with random forest. Briefings in bioinformatics, 24(2), bbad002.
37. Biau, G., & Scornet, E. (2016). Une visite guidée par forêt aléatoire. *Test* , *25* (2), 197-227.
38. Li, W., Yin, Y., Quan, X., & Zhang, H. (2019). Gene expression value prediction based on XGBoost algorithm. *Frontiers in genetics*, *10*, 1077.

39. Li, J., An, X., Li, Q., Wang, C., Yu, H., Zhou, X., & Geng, Y. A. (2022). Application of XGBoost algorithm in the optimization of pollutant concentration. *Atmospheric Research*, *276*, 106238.
40. Ramraj, S., Uzir, N., Sunil, R., & Banerjee, S. (2016). Experimenting XGBoost algorithm for prediction and classification of different datasets. International Journal of Control Theory and Applications, 9(40), 651-662.
41. Chun, W. (2001). *Programmation Python de base* (Vol. 1). Prentice Hall Professional.
42. Harris, F. W. (1913). How many parts to make at once. *Factory, the magazine of management*, *10*(2), 135-136.
43. Ohno, T. (1988). Toyota Production System. Productivity Press. *vol*, *4*, 3-11.
44. Lee, H. L., Padmanabhan, V., & Whang, S. (1997). Information distortion in a supply chain: The bullwhip effect. *Management science*, *43*(4), 546-558.
45. Rogers, E. M. (2003). Diffusion of Innovations, Free Press. *New York*, *551*.
46. Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, *13*(3), 319-340.
47. Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). *User acceptance of information technology: Toward a unified view.* MIS Quarterly, 27(3), 425-478.
48. Chopra, S., & Meindl, P. (2019). Supply chain management. Strategy, planning & operation. In Das Summa Summarum des Management: Die 25 wichtigsten Werke für Strategie, Führung und Veränderung (pp. 265-275). Wiesbaden: Gabler.
49. Christopher, M. (2016). Logistics and supply chain management: logistics & supply chain management. Pearson UK.
50. Ivanov, D. (2021). Introduction à la résilience de la chaîne d'approvisionnement : gestion, modélisation, technologie . Springer Nature.
51. Ivanov, D., & Dolgui, A. (2021). A digital supply chain twin for managing the disruption risks and resilience in the era of Industry 4.0. Production Planning & Control, 32(9), 775-788.

Table des Matières

Annexe 1 : Code de création d'un calendrier de fêtes sénégalaises

```
# Classe Sénégal personnalisée
class Senegal(WesternCalendar):
    """Sénégal"""

    # Fêtes fixes grégoriennes intégrées dans workalendar
    FIXED_HOLIDAYS = WesternCalendar.FIXED_HOLIDAYS + (
        (4,  4,  "Fête nationale"),
        (5,  1,  "Fête du Travail"),
        (8, 15,  "Assomption"),
        (11,  1, "Toussaint"),
    )

    # Fêtes islamiques : (nom_local, mois_hijri, jour_hijri)
    FETES_ISLAMIQUES = [
        ("Nouvel An Hégirien",  1,  1),
        ("Tamkharit (Achoura)", 1, 10),
        ("Magal de Touba",      2, 18),
        ("Gamou / Mawlid",      3, 12),
        ("Korité (Aïd al-Fitr)",10,  1),
        ("Tabaski (Aïd al-Adha)",12,10),
    ]

    def _get_islamic_holidays(self, year):
        """Convertit les dates hijri → grégorien pour une année donnée."""
        result = []
        # Balayer 2 années hégiriennes pour couvrir l'année grégorienne
complète
        for h_year in range(year - 580, year - 576):
            for nom, mois, jour in self.FETES_ISLAMIQUES:
                try:
                    g = Hijri(h_year, mois, jour).to_gregorian()
                    d = date(g.year, g.month, g.day)
                    if d.year == year:
                        result.append((d, nom))
                except Exception:
                    pass
        return result

    def get_variable_days(self, year):
        """Retourne toutes les fêtes variables (islamiques) pour l'année."""
        days = super().get_variable_days(year)
        days += self._get_islamic_holidays(year)
        return days
```

```
#  illustration
cal = Senegal()

# Lister tous les fériés de l'année 2024
print(" Fériés 2024 ")
for d, label in sorted(cal.holidays(2024)):
    print(f"  {d}  →  {label}")

print()

#  Fonctions métier de workalendar
#  Vérifier si un jour est ouvrable
test_date = date(2024, 4, 10)   # Korité 2024
print(f"{test_date} est ouvrable ? {cal.is_working_day(test_date)}")

# 2. Ajouter N jours ouvrables
d_depart = date(2024, 4,  8)
d_fin    = cal.add_working_days(d_depart, 5)
print(f"8 avr + 5 jours ouvrables = {d_fin}")

# 3. Nombre de jours ouvrables entre deux dates
delta = cal.get_working_days_delta(date(2024, 1, 1), date(2024, 12, 31))
print(f"Jours ouvrables en 2024 : {delta}")

# Construire un set de timestamps pour filtrage pandas
def feries_set(annee_debut: int, annee_fin: int) -> set:
    return {
        pd.Timestamp(d)
        for y in range(annee_debut, annee_fin + 1)
        for d, _ in cal.holidays(y)
    }

feries_senegal = feries_set(2009, 2025)
print(f"\nTotal fériés 2009-2025 : {len(feries_senegal)} dates ")
```

Annexe 2 : code de calcul du SS et ROP

```
# Calcul SS et ROP par produit — Z différencié par catégorie
df_preds = df_test_fe.reset_index(drop=True).copy()
df_preds['pred_hybrid'] = y_pred_hybrid
df_preds['erreur_abs']  = np.abs(y_test.values - y_pred_hybrid)

recommandations = []

for produit, grp in df_preds.groupby('Produit'):
    cat  = grp['Categorie'].iloc[0]
    four = grp['Fournisseur'].iloc[0]
    Z    = Z_PAR_CATEGORIE.get(cat, Z_DEFAULT)

    mu    = grp['pred_hybrid'].mean()    # demande moyenne prédite
    sigma = grp['pred_hybrid'].std()     # volatilité de la demande prédite

    if sigma > 0:
        SS  = Z * sigma * np.sqrt(DELAI)
        ROP = mu * DELAI + SS
    else:
        SS  = 0
        ROP = mu * DELAI

    # MAE produit-spécifique (précision du modèle sur ce produit)
    mae_produit = grp['erreur_abs'].mean()

    recommandations.append({
        'Produit'               : produit,
        'Categorie'             : cat,
        'Fournisseur'           : four,
        'Z_Niveau_Service'      : Z,
        'Demande_Moy_Predite'   : round(mu, 2),
        'Ecart_Type_Demande'    : round(sigma, 2),
        'MAE_Modele'            : round(mae_produit, 4),
        'Stock_Securite_SS'     : round(SS, 2),
        'ROP_Point_Reappro'     : round(ROP, 2),
        'Nb_Obs_Test'           : len(grp),
    })

df_recommandations =
pd.DataFrame(recommandations).sort_values('ROP_Point_Reappro',
ascending=False)
print(f' Recommandations calculées pour {len(df_recommandations)} produits')
print(f'\nZ par catégorie dans les recommandations :')
print(df_recommandations.groupby('Categorie')['Z_Niveau_Service'].first().to_s
tring())
df_recommandations.head(10)
```