

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

الجمهورية الجزائرية الديمقراطية الشعبية

MINISTERE DE L'ENSEIGNEMENT
SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

ECOLE SUPERIEURE EN
SCIENCE APPLIQUEES
-TLEMCCEN-



المدرسة العليا للعلوم التطبيقية
Ecole Supérieure en
Sciences Appliquées

وزارة التعليم العالي والبحث العلمي

المدرسة العليا في العلوم التطبيقية
-تلمسان-

Mémoire de fin d'étude

Pour l'obtention du diplôme d'ingénieur

Filière : Génie Industriel

Spécialité : Management Industriel et Logistique

Présenté par Asma HAMZI



Thème

**Développement d'une solution intelligente d'aide à la décision pour la
prédiction de la production : application aux lignes de confection de
l'entreprise TAYAL S.P.A**

Soutenu publiquement, le 22/06/2026 devant le jury composé de :

Mr. Mohammed M'HAMED	MCB ESSA. Tlemcen	Président
Mme. Zoulikha BELMEKKI	MCB ESSA. Tlemcen	Directrice de mémoire
Mr. Fouad MALIKI	MCA ESSA. Tlemcen	Co- Directeur de mémoire
Mr. Mustapha Anwar BRAHAMI	MCA ESSA. Tlemcen	Examinateur 1
Mme. Wahiba FANDI	MCB ESSA. Tlemcen	Examinatrice 2

Année universitaire : 2025/2026

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciement

Avant toute chose, nous rendons grâce à Allah, le Tout-Puissant et le Miséricordieux, de nous avoir accordé la force, la patience et la volonté nécessaires pour mener à bien ce travail. C'est par Sa grâce que ce modeste effort a pu voir le jour.

Nous tenons à exprimer notre profonde gratitude à **Mr. Fouad Maliki**, notre encadrant et chef de filière, pour la confiance qu'il nous a accordée, ses conseils éclairés et son accompagnement indéfectible tout au long de ce parcours. Sa rigueur scientifique, sa pédagogie bienveillante et son engagement constant ont été une boussole précieuse dans la conduite de ce projet.

Nous adressons également nos sincères remerciements à **Mme. Zoulikha KOUDAD**, notre co-encadrante et enseignante, pour son implication précieuse, ses orientations avisées et la générosité avec laquelle elle a partagé son expertise.

Nos remerciements les plus chaleureux s'adressent à **l'entreprise TAYAL SPA** pour son accueil et la qualité de la collaboration qu'elle nous a offerte.

Nous remercions tout particulièrement **Mr. Mehmet Topçu**, Directeur de l'unité de confection, pour ses conseils précieux et les encouragements sincères qu'il nous a prodigués. Ses échanges ont été d'une richesse inestimable pour l'avancement de ce travail.

Nos remerciements vont également à **Mr. Mulazim Ipek**, Manager du département planification, pour son accueil professionnel et la sérénité qu'il a su instaurer dans notre environnement de travail.

Nous exprimons une reconnaissance toute particulière à **Mme. Nour El Houda MOKHTARI**, Ingénieure en méthodologie et notre tutrice au sein de l'Unité 23, dont le suivi attentif, les orientations précises et le soutien constant ont été au cœur de la réussite de notre stage de fin d'études. Travailler à ses côtés a été un véritable honneur.

Nous tenons à exprimer notre gratitude la plus respectueuse et la plus sincère aux membres du jury : **Monsieur Mohammed M'HAMEDI**, **Monsieur Mustapha Anwar BRAHAMI** et **Madame Wahiba FANDI**, qui nous font l'insigne honneur d'évaluer ce mémoire. Leur expertise, leur regard critique et le temps qu'ils y consacrent représentent pour nous bien plus qu'une formalité académique c'est une reconnaissance du travail accompli et une opportunité rare d'en enrichir la valeur. Nous leur en sommes profondément reconnaissants.

Nous remercions l'ensemble de nos enseignants pour les connaissances et les compétences qu'ils nous ont transmises avec dévouement tout au long de notre formation. Chaque cours, chaque échange a posé une pierre de l'édifice que représente ce mémoire.

Nos remerciements s'étendent également au personnel administratif et technique de l'école, pour leur accueil chaleureux, leur disponibilité et leur contribution silencieuse mais essentielle à la qualité de notre cadre d'études.

Enfin, nous remercions du fond du cœur toutes les personnes qui, de près ou de loin, ont accompagné ce parcours. Par une opportunité offerte, un mot d'encouragement, un conseil avisé ou un simple sourire au bon moment, chacune a contribué, à sa manière, à l'aboutissement de ce travail. À toutes et à tous merci.

Dédicace

À la lumière de ma vie, à ceux sans qui rien de tout cela n'aurait eu de sens...

À mes parents bien-aimés, Sihem et Mourad.

Il est des dettes que l'on ne rembourse jamais la vôtre en est une, et je l'assume avec fierté.

Maman, tu es mon héroïne silencieuse. Dans chaque épreuve, tu as incarné la patience et la volonté avec une grâce que j'ai appris, peu à peu, à reconnaître comme ta force la plus profonde.

Papa, tu es ma référence, mon cap. Ta force tranquille et ton courage discret ont été, sans que tu le saches toujours, ma boussole dans les moments de doute.

Vous avez été mes anges gardiens présents derrière mes fragilités, solides quand je vacillais, lumineux quand je ne voyais plus.

Ce travail est le reflet de vos sacrifices invisibles, de vos prières silencieuses et de cette foi inébranlable que vous avez toujours placée en mon avenir. Je vous le dédie avec une reconnaissance qui n'aura pas de fin.

À mon frère Rayane,

Mon complice, mon ami, mon témoin fidèle des moments qui comptent vraiment. Tu as partagé avec moi les pages les plus importantes de mon histoire, avec cette simplicité et cette affection qui me sont précieuses. Merci d'être là simplement, sincèrement, toujours.

À mes grands-parents, Mima Yasmina et Papa Abdelhamid,

J'ai la chance immense de vous avoir à mes côtés, et je ne l'oublie pas. Votre soutien, vos bénédictions et votre amour discret ont été une lumière douce sur ce chemin. Vous représentez pour moi les racines profondes qui permettent à l'arbre de grandir sans craindre la tempête.

À ma chère tante Fatima,

Qui n'a jamais manqué de penser à moi, de près ou de loin, dans les grands moments comme dans les petits. Vos attentions ont compté bien plus que vous ne l'imaginez.

À ma précieuse ami Hasnaa,

Ton amitié est un trésor rare, et les années passées ensemble n'ont fait que le confirmer. Nous avons traversé ce parcours d'ingénieur côte à côte les joies franches, les nuits difficiles, les doutes partagés à voix basse et les victoires célébrées ensemble. Chaque moment, même le plus éprouvant, n'a fait que renforcer ce lien qui nous unit. Ta présence a été, pour moi, un soutien inestimable.

À tous mes amis et mes collègues de promotion,

Même si vos noms ne figurent pas tous ici, vos visages et votre générosité sont gravés dans ma mémoire. Vous avez rendu ce parcours plus léger, plus vivant, plus humain. Merci pour les sourires partagés, l'entraide et cette belle énergie collective qui a porté chacun d'entre nous.

À tous ceux qui, de près ou de loin,

Ont été là par un regard bienveillant, un mot au bon moment, un geste simple et sincère...

Vous avez, chacun à votre façon, posé une pierre sur ce chemin. Merci, du fond du cœur.

Et à moi-même,

Pour avoir choisi de continuer quand tout invitait à s'arrêter. Pour avoir tenu forte, entière et debout malgré tout.

Ce travail est aussi le mien. Et j'en suis fière.

Asma HAMZI

Résumé

Ce travail porte sur l'optimisation de la planification et de l'affectation des ordres de production au sein de l'unité de confection de l'entreprise textile TAYAL SPA, en s'appuyant sur la prédiction de la quantité produite par modèle et par ligne sur une même journée. Dans un environnement marqué par l'instabilité, où le facteur humain en interface directe avec les machines influence fortement la productivité ; retards, absences, pannes sous de fortes contraintes de coûts, l'étude met en évidence l'intérêt d'une gestion anticipative et intelligente des quantités affectées à chaque ligne. Un modèle de gradient boosting ; XGBoost est mobilisé pour estimer la quantité réellement produite d'un article de prêt-à-porter, et une application web rend ces résultats exploitables par les responsables de production en confrontant quantité prédite et quantité planifiée. Les résultats montrent une meilleure anticipation de la sortie du système, la prévention de l'écart entre prévision et réalisation autorisant décisions et marges avant production ainsi qu'une maîtrise accrue des performances réelles de chaque ligne selon le modèle traité. En conclusion, le travail souligne l'importance de la digitalisation de la gestion des ordres de production et propose l'intégration d'outils d'aide à la décision fondés sur l'intelligence artificielle, dans une interface simple et accessible, contribuant à l'amélioration continue des indicateurs clés de performance ; quantité produite par jour et respect des délais.

Abstract

This work addresses the optimization of production-order planning and assignment within the garment unit of the textile company TAYAL SPA, relying on the prediction of the quantity produced per style and per line on the same day. In an environment marked by instability, where the human factor directly interfacing with the sewing machines strongly affects productivity ; delays, absences, breakdowns under tight cost constraints, the study highlights the value of an anticipatory and intelligent management of the quantities assigned to each line. A gradient boosting model ; XGBoost is used to estimate the actual quantity produced for a ready-to-wear article, and a web application makes these results usable by production managers by comparing predicted and planned quantities. The results show better anticipation of the system's output, prevention of the gap between forecast and actual production enabling decisions and safety margins before production and improved control of each line's real performance according to the processed style. In conclusion, the work emphasizes the importance of digitalizing production-order management and proposes integrating artificial-intelligence-based decision-support tools within a simple, accessible interface, contributing to the continuous improvement of key performance indicators ; daily output and on-time delivery.

ملخص

يتناول هذا العمل تحسين تخطيط أوامر الإنتاج وتوزيعها داخل وحدة الخياطة بالشركة النسيجية TAYAL SPA ، اعتماداً على التنبؤ بالكمية المنتجة لكل موديل ولكل خط في اليوم نفسه. ففي بيئة تتسم بعدم الاستقرار، حيث يؤثر العامل البشري المتفاعل مباشرة مع آلات الخياطة تأثيراً كبيراً في الإنتاجية كالتأخر، الغياب والأعطال في ظل قيود صارمة على التكاليف، تُبرز الدراسة أهمية إدارة استباقية وذكية للكميات المُسندة إلى كل خط. وقد استخدم نموذج التعزيز التدرجي XGBoost لتقدير الكمية المنتجة فعلياً لمنتج من الملابس الجاهزة، مع تطوير تطبيق ويب يجعل هذه النتائج قابلة للاستثمار من قِبل مسؤولي الإنتاج عبر مقارنة الكمية المتوقعة بالكمية المخططة. وتُظهر النتائج تحسناً في استباق مخرجات النظام، والوقاية من الفجوة بين التوقع والإنجاز بما يتيح اتخاذ القرارات وتحديد هوامش الأمان قبل الإنتاج إضافةً إلى تحكّم أفضل في الأداء الفعلي لكل خط حسب الموديل المُعالج. وخلاصةً، يؤكد العمل أهمية رقمنة إدارة أوامر الإنتاج، ويقترح دمج أدوات دعم القرار المعتمدة على الذكاء الاصطناعي ضمن واجهة بسيطة وسهلة الاستخدام، بما يساهم في التحسين المستمر لمؤشرات الأداء الرئيسية كالكمية المنتجة يومياً واحترام الأجل.

Mots-clés : planification, affectation, lignes de confection, prédiction, apprentissage automatique, XGBoost, efficience des lignes, quantités produites , KPI, TAYAL SPA.

Table des matières

Remerciement	I
Dédicace.....	II
Résumé	IV
Liste des Figures	X
Listes des tableaux.....	XIII
Liste des sigles et acronymes	XIV
Introduction générale.....	1
CHAPITRE 01 : PRESENTATION DE L'ENTREPRISE TAYAL SPA.....	2
1. Introduction	3
2. Présentation de l'entreprise Tayal SPA	3
2.1 Capacités.....	4
2.2 Mission	4
2.3 Produits.....	4
2.3.1 Fil.....	4
2.3.2 Tissu	5
2.3.3 Prêt-à-porter.....	5
2.4 Unités de l'entreprise.....	5
3. Processus de production textile chez TAYAL	7
3.1 Stockage de matière première (Unité 2)	7
3.2 Unités de filature (unités 3 et 4).....	8
3.2.1 Ouvraison.....	8
3.2.2 Épuration et nettoyage.....	8
3.2.3 Cardage	9
3.2.4 Doublage	9
3.2.5 Peignage	9
3.2.6 Étirement	10
3.2.7 Roving (Banc à broche)	10
3.2.8 Filature Ring (Ring Spinning)	11
3.2.9 Bobinage (Cone Winding)	11
3.2.10 Types de fils produits.....	12
3.3 Indigo (Unité 5)	12
3.3.1 Teinture en Corde (Rope Dyeing)	13
3.3.2 Séchage.....	14
3.3.3 Réorganisation des Fils (Rebeaming).....	15

3.4 Tissage (unité 8 et 9)	15
3.4.1 Préparation du tissu non-denim (Direct Warping)	16
3.4.3 Encollage (Sizing).....	16
3.5 Tricotage (unité 6)	16
3.5 .1 Machines de tricotage	17
3.5.2 Les types de tissus tricotés.....	17
3.6 Teinture (unité 11)	18
3.6.1 Processus de teinture et finition des tissus tricotés	18
3.7 Unité de Confection (Unité 23)	21
3.7.1 Organisation générale de l'unité de confection	22
3.7.2 Traitement de la commande client	22
3.7.3 Rôle du département planning	23
3.7.4 Rôle du département méthodologie.....	24
3.7.5 Préparation des matières premières et des accessoires	24
3.7.6 Flux de production dans l'unité 23	24
4. Conclusion.....	28
<i>CHAPITRE 02 : INTELLIGENCE ARTIFICIELLE : CONCEPTS FONDAMENTAUX ET MÉTHODES POUR LA PRÉDICTION</i>	29
1. Introduction	30
2. Définition des concepts fondamentaux de l'intelligence artificielle.....	30
2.1 Définition de l'intelligence artificielle (IA).....	30
2.2 Définition de l'apprentissage automatique (Machine Learning)	31
2.3 Définition du Deep Learning (apprentissage profond)	31
2.4 Définition de l'IA générative (Generative AI).....	32
2.5 Applications majeures de l'IA	32
2.6 Pourquoi l'IA devient un acteur clé dans les industries ?.....	33
3 Fondements du Machine Learning.....	34
3.1 L'apprentissage.....	34
3.2 Catégories de problèmes en apprentissage automatique	34
3.2.1 Apprentissage supervisé	34
3.2.2 Apprentissage non supervisé.....	34
3.2.3 Apprentissage par renforcement	34
3.3 Typologie des problèmes et méthodes associées.....	35
3.3.1 Apprentissage supervisé	35
3.3.2 Apprentissage non supervisé.....	36

3.3.4 Apprentissage par renforcement	37
4 Fondements de l'apprentissage supervisé pour la prédiction de la performance et l'optimisation de la planification de la production	37
4.1 L'apprentissage supervisé dans un contexte de logistique et de planification de la production	37
4.2 Réseaux de neurones artificiels	38
4.2.1 Évolution historique et émergence des réseaux de neurones artificiels	38
4.2.2 Du neurone biologique au neurone artificiel	38
4.2.3 Structure générale d'un réseau de neurones artificiels	39
4.2.4 Principe de la prédiction par apprentissage supervisé	39
4.2.5 Le perceptron comme modèle de base de l'apprentissage neuronal	40
et la fonction seuil elle-même est définie par :	40
4.2.6 Fonctions d'activation et modélisation des relations non linéaires	41
4.2.7 Optimisation de l'apprentissage par la méthode de descente de gradient	42
4.2.8 Mécanisme d'apprentissage par rétropropagation du gradient	43
4.2.9 Généralisation, validation et performance des modèles neuronaux	44
4.2.10 Apports des réseaux de neurones artificiels dans les tâches de prédiction	44
4.3 XGBOOST	45
4.3.1 Des modèles simples aux modèles ensemblistes : le principe de combinaison d'apprenants faibles	45
4.3.2 L'arbre de décision comme apprenant de base	47
4.3.3 Du Gradient Boosting classique à XGBoost	49
4.3.4 Architecture interne et mécanismes clés de XGBoost	51
4.3.5 Optimisation algorithmique	53
4.3.6 Hyperparamètres fondamentaux et stratégie de réglage	54
4.3.7 Évaluation, généralisation et interprétabilité	56
4.3.8 XGBoost dans l'écosystème des algorithmes de prédiction	58
5. Conclusion	60
<i>CHAPITRE 03 : PREDICTION DE LA QUANTITE JOURNALIERE PRODUITE PAR LIGNE VIA LE MODELE XGBOOST</i>	61
1. Introduction	62
2. Problématique	63
3. Hypothèse	64
4. Méthode de travail	65
4.1 Compréhension du métier	65
4.2 Compréhension des données	65

4.3 Préparation des données	65
4.4 Modélisation et choix du modèle	65
4.5 Résolution	65
4.6 Résultats, interprétation et évaluation.....	65
4.7 Déploiement.....	66
5. 5. Développement d'un Système Intelligent de Prédiction de la Production Journalière en Confection.....	66
5.1 Compréhension du métier.....	66
5.1.1 Qu'est-ce qu'un ordre de production ?	66
5.1.2 Les critères d'affectation vers une ligne.....	67
5.1.3 Règles d'affectation des ordres aux lignes	68
5.2 Compréhension et Préparation des données	69
5.2.1 Compréhension de données.....	69
5.2.2 Préparation de données	74
5.3 Modélisation et choix du modèle	74
5.3.1 Première approche : les réseaux de neurones	75
5.3.2 Approche retenue : le gradient boosting (XGBoost)	76
5.4 Résolution	79
5.4.1 De l'entraînement aux artefacts.....	79
5.4.2 Le moteur de prédiction predict_production.py	79
5.4.3 Le moteur d'affectation prediction_core.py	79
5.5 Résultats, interprétation et évaluation.....	80
5.5.1 Qualité des prédictions	81
5.5.2 Diagnostics de validité du modèle.....	85
5.5.3 Synthèse comparative : XGBoost vs réseaux de neurones	93
5.5.4 Pertinence de la prédiction et de l'affectation	93
5.5.5 Logique d'amélioration continue.....	95
5.5.6 Conclusion de l'évaluation	95
5.6 Déploiement.....	95
5.6.1 Page d'accueil et entrée dans l'application	95
5.6.2 Contrôle d'accès	96
5.6.3 Barre de navigation.....	97
5.6.4 Onglet « Prédiction »	98
5.6.5 Onglet « Plan »	100
5.6.6 Onglet « Rapports ».....	101

5.6.7 Onglet « Réglage »	103
5.6.8 Onglet « Profil »	103
6. Conclusion.....	104
Conclusion générale	105
Bibliographie.....	106

Liste des Figures

Figure	Désignation	Page
Chapitre 1	Présentation de l'entreprise TAYAL SPA	
Figure 1.1	Logo de TAYAL SPA.....	3
Figure 1.2	Géolocalisation de l'entreprise TAYAL SPA.....	3
Figure 1.3	Gamme de produits de l'entreprise.....	4
Figure 1.4	Unités de production de l'entreprise.....	7
Figure 1.5	Unité 2 de stockage de la matière première.....	7
Figure 1.6	Unité 3 de filature.....	8
Figure 1.7	Machine d'épuration Uni clean.....	8
Figure 1.8	Machine d'épuration Uni mix.....	8
Figure 1.9	Carding machine.....	9
Figure 1.10	Banc d'étirage 01.....	9
Figure 1.11	Combing (Peigneuse).....	10
Figure 1.12	Banc d'étirage 02.....	10
Figure 1.13	Banc à broche.....	10
Figure 1.14	Ring Spinning.....	11
Figure 1.15	Bobinage.....	11
Figure 1.16	Combed.....	12
Figure 1.17	Carded.....	12
Figure 1.18	Open-end.....	12
Figure 1.19	Machine de rope dyeing.....	13
Figure 1.20	Dyeing machine (Box 6 – Box 13).....	13
Figure 1.21	Zone d'oxydation.....	13
Figure 1.22	STEAMER.....	14
Figure 1.23	Dryer 1.....	14
Figure 1.24	Dryer 2.....	14
Figure 1.25	Machine de rebeaming.....	15
Figure 1.26	Étape de préparation pour le tissage.....	16
Figure 1.27	Unité 6 de tricotage.....	17
Figure 1.28	Machine de blanchiment.....	18
Figure 1.29	Emballage de rouleaux avec du plastique.....	18
Figure 1.30	Machine de la teinture de non denim.....	19
Figure 1.31	Machine de la teinture de denim.....	19
Figure 1.32	Machine de lavage.....	19
Figure 1.33	Essorage.....	20
Figure 1.34	Machine de séchage.....	20
Figure 1.35	Machine de sanforisage.....	20
Figure 1.36	Unité 23 de confection.....	21
Figure 1.37	Flux de production de l'unité 23.....	25
Figure 1.38	Lignes de repassage.....	27
Chapitre 2	Intelligence artificielle : concepts fondamentaux et méthodes pour la prédiction	
Figure 2.1	Relations hiérarchiques entre l'Intelligence Artificielle et le Machine Learning.....	31
Figure 2.2	Du neurone biologique au neurone artificiel.....	38

Figure 2.3	Structure générale d'un réseau de neurones artificiels.....	39
Figure 2.4	Structure générale d'un perceptron.....	40
Figure 2.5	Fonctions d'activation d'un neurone artificiel.....	41
Figure 2.6	Descente de gradient et problème des minima locaux.....	42
Figure 2.7	Structure de la méthode Bagging.....	47
Figure 2.8	Structure de la méthode Boosting.....	47
Figure 2.9	États possibles du modèle d'apprentissage.....	57
Figure 2.10	Structure de XGBoost, LightGBM et CatBoost.....	59
Chapitre 3	Prédiction de la quantité journalière produite par ligne via le modèle XGBoost	
Figure 3.1	Distribution des postes de travail sur une ligne de confection.....	69
Figure 3.2	Répartition des affectations par catégorie	71
Figure 3.3	Résultats de test des trois réseaux de neurones (POLO, BASIC, CASUAL).....	75
Figure 3.4	Pipeline de modélisation XGBoost par catégorie	77
Figure 3.5	Les quatre modes d'affectation.....	80
Figure 3.6	Classement des modèles par R^2 (POLO, BASIC, CASUAL).....	81
Figure 3.7	R^2 de test et R^2 de validation croisée par ligne de production.....	82
Figure 3.8	Comparaison des métriques d'erreur (MAE et RMSE) par modèle.....	82
Figure 3.9	Distribution des erreurs pour le segment POLO.....	84
Figure 3.10	Courbes d'apprentissage, POLO.....	85
Figure 3.11	Courbes d'apprentissage, BASIC.....	85
Figure 3.12	Courbes d'apprentissage, CASUAL.....	86
Figure 3.13	Prédit vs réel, segment POLO.....	86
Figure 3.14	Prédit vs réel, segment BASIC.....	87
Figure 3.15	Prédit vs réel, segment CASUAL.....	87
Figure 3.16	Analyse des résidus, POLO	88
Figure 3.17	Analyse des résidus, BASIC.....	88
Figure 3.18	Analyse des résidus, CASUAL.....	88
Figure 3.19	QQ-plot des résidus, POLO.....	89
Figure 3.20	QQ-plot des résidus, BASIC.....	89
Figure 3.21	QQ-plot des résidus, CASUAL.....	90
Figure 3.22	Réel vs prédit (série), POLO.....	90
Figure 3.23	Réel vs prédit (série), BASIC.....	90
Figure 3.24	Réel vs prédit (série), CASUAL.....	91
Figure 3.25	Importance SHAP moyenne, POLO.....	91
Figure 3.26	Diagramme SHAP beeswarm, POLO.....	91
Figure 3.27	Importance SHAP moyenne, BASIC.....	92
Figure 3.28	Diagramme SHAP beeswarm, BASIC.....	92
Figure 3.29	Importance SHAP moyenne, CASUAL.....	92
Figure 3.30	Diagramme SHAP beeswarm, CASUAL.....	92
Figure 3.31	Avancement cumulé du PO : production prédite vs objectif planifié.....	94
Figure 3.32	Page d'accueil de l'application.....	96
Figure 3.33	Boîte de dialogue d'authentification par mot de passe.....	96
Figure 3.34	Barre de navigation de l'application.....	97
Figure 3.35	Onglet Prédiction — 1er champ.....	98
Figure 3.36	Onglet Prédiction — 2ème champ.....	98
Figure 3.37	Résultats de prédiction.....	99
Figure 3.38	Résultats de prédiction (suite).....	99

Figure 3.39	Registre des affectations de production.....	100
Figure 3.40	Actions disponibles sur une affectation.....	101
Figure 3.41	Rapport « État des lignes »	101
Figure 3.42	Taux d'atteinte des objectifs par catégorie.....	102
Figure 3.43	Détail par ordre de production sur l'objectif.....	102
Figure 3.44	Onglet Réglage.....	103
Figure 3.45	Onglet Profil.....	103

Listes des tableaux

Tableau	Désignation	Page
Chapitre 2	Intelligence artificielle : concepts fondamentaux et méthodes pour la prédiction	
Tableau 2.1	Critères de comparaison entre le Bagging et le Boosting	47
Tableau 2.2	Signification des paramètres de gain	48
Tableau 2.3	Critères de choix de XGBoost	59
Chapitre 3	Prédiction de la quantité journalière produite par ligne via le modèle XGBoost	
Tableau 3.1	Composantes principales d'un ordre de production	67
Tableau 3.2	Exemple d'ordres de production planifiés et affectés	68
Tableau 3.3	Rapport journalier de la production – Données de balance et de capacités	70
Tableau 3.4	Rapport journalier de la production – Données sur la production et la performance	70
Tableau 3.5	Données d'entrée exploitables pour l'entraînement	71
Tableau 3.6	Données de sortie exploitables pour l'entraînement	71
Tableau 3.7	Variables d'identification	72
Tableau 3.8	Ressources et conditions de la journée	72
Tableau 3.9	Historique de comportement	73
Tableau 3.10	Variables dérivées	73
Tableau 3.11	Variables temporelles et de mémoire	73
Tableau 3.12	Critères d'évaluation du réseau de neurones	75
Tableau 3.13	Hyperparamètres calibrés par famille	78
Tableau 3.14	Bibliothèques mobilisées au cours de la résolution du système	80
Tableau 3.15	Tableau de métriques, segment POLO	83
Tableau 3.16	Tableau de métriques, segment BASIC	84
Tableau 3.17	Tableau de métriques, segment CASUAL	84
Tableau 3.18	Comparaison mesurable des deux approches	93

Liste des sigles et acronymes

Sigle	Signification
IA	Intelligence Artificielle
ML	Machine Learning (apprentissage automatique)
DL	Deep Learning (apprentissage profond)
RN	Réseau de Neurones
NN	Neural Network
MLP	Multi-Layer Perceptron (perceptron multicouche)
RNN	Recurrent Neural Network (réseau de neurones récurrent)
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
CNN	Convolutional Neural Network (réseau de neurones convolutif)
GAN	Generative Adversarial Network
ReLU	Rectified Linear Unit (fonction d'activation)
SVM	Support Vector Machine (machine à vecteurs de support)
SVR	Support Vector Regression
GMM	Gaussian Mixture Model
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
XGBoost	Extreme Gradient Boosting
GBM	Gradient Boosting Machine
LightGBM	Light Gradient Boosting Machine
CatBoost	Categorical Boosting
SHAP	SHapley Additive exPlanations
ARIMA	AutoRegressive Integrated Moving Average
CV	Validation croisée (Cross-Validation)
RMSE	Root Mean Squared Error (racine de l'erreur quadratique moyenne)
MAE	Mean Absolute Error (erreur absolue moyenne)

MAPE	Mean Absolute Percentage Error (erreur pourcentage absolue moyenne)
KPI	Key Performance Indicator (indicateur clé de performance)
PO	Production Order (ordre de production)
SDT	Standard Time / Temps de traitement standard
QQ-plot	Quantile-Quantile plot (droite de Henry)

Introduction générale

La planification de la production s'est imposée, au fil des décennies, comme l'un des leviers décisifs du pilotage de l'entreprise industrielle. Loin d'être une simple opération d'ordonnancement, elle conditionne aujourd'hui la capacité d'une organisation à honorer ses engagements, à utiliser au mieux ses ressources et à tenir ses délais [2]. Une planification rigoureuse se traduit directement par des gains mesurables : moins de gaspillage, une meilleure productivité, une compétitivité renforcée et au bout de la chaîne, la satisfaction d'un client de plus en plus exigeant. À l'inverse, une planification approximative engendre des écarts coûteux entre ce qui est prévu et ce qui est réellement produit. Ce qui n'était longtemps qu'une tâche administrative est ainsi devenu un véritable outil de valeur, qui aide et facilite l'exécution des opérations et structure la performance globale de l'entreprise.

Dans le même temps, l'intelligence artificielle a profondément transformé la manière dont les entreprises exploitent leurs données. Capable d'apprendre des comportements passés pour anticiper les situations futures, elle apporte une valeur considérable en convertissant un historique brut en aide à la décision concrète [3]. C'est dans cette vision que s'inscrit le présent travail : faire de l'intelligence artificielle un acteur actif de la production, et plus précisément de la planification des ordres de production en les affectant aux lignes de confection. L'enjeu est de répondre à une attente claire satisfaire une demande, qui se traduit par une quantité à produire sur un ordre donné dans le respect des délais tout en suivant et en mesurant les indicateurs de performance qui en garantissent l'atteinte, au service de la satisfaction du client.

Pour atteindre cet objectif, nous abordons d'abord les concepts fondamentaux de l'intelligence artificielle, avant de plonger au cœur des modèles d'apprentissage et, plus particulièrement, de celui qui produit la prédiction recherchée. Nous appliquons enfin cette démarche à un cas d'étude concret au sein de l'entreprise TAYAL SPA, en cherchant à prédire la quantité produite par jour, par ligne et par modèle de prêt-à-porter. Cette prédiction est intégrée à une application web qui transforme l'exécution programmable des codes en un outil directement utilisable par les ingénieurs de planification, à travers une interface facilitant l'interaction avec les résultats.

Ce mémoire s'articule autour de trois chapitres : le premier présente l'entreprise TAYAL SPA et son environnement industriel, le deuxième expose les concepts fondamentaux de l'intelligence artificielle et les méthodes de prédiction, et le troisième déroule le cas pratique, de la compréhension des données jusqu'au déploiement de l'application.

À travers cette étude, nous mettons en lumière l'impact de l'intelligence artificielle sur la performance opérationnelle et la compétitivité des entreprises.

***CHAPITRE 01 : PRESENTATION DE L'ENTREPRISE
TAYAL SPA***

1. Introduction

TAYAL S.P.A. est l'une des entreprises de référence dans le secteur textile en Algérie. Elle se distingue par une chaîne de production intégrée couvrant l'ensemble du cycle de fabrication, depuis la production de fil et de tissu jusqu'à la confection de vêtements prêt-à-porter. Grâce à des capacités industrielles importantes et au respect de normes de qualité élevées, l'entreprise occupe une position notable sur son marché et s'inscrit dans une dynamique de performance et de compétitivité.

Ce premier chapitre est consacré à la présentation générale de TAYAL S.P.A. Il retrace l'historique de l'entreprise, précise ses missions et ses objectifs stratégiques, et met en évidence ses principales activités ainsi que ses unités de production. Cette analyse permettra d'établir un cadre de compréhension clair du contexte industriel et des mécanismes de production dans lesquels évolue l'entreprise, constituant ainsi une base essentielle pour la suite de ce travail.

2. Présentation de l'entreprise Tayal SPA

L'Algérienne des Industries Textiles TAYAL S.P.A est une entreprise textile implantée dans la zone industrielle de Sidi Khettab wilaya de Relizane [51].

Créée sous forme de joint-venture algéro-turque depuis novembre 2013, elle résulte d'un partenariat entre C&H Group, E.P.E TEXALG, MADAR Holding et INTERTAY (filiale du groupe textile turc TAY Group).



Figure 1.1 : Logo de TAYAL SPA

Le projet industriel s'étend sur une superficie totale d'environ 2 500 000 m² (première phase), dont 1 000 000 m² sont déjà réalisés et opérationnels, avec plus de 400 000 m² de surface couverte répartie en 30 blocs et unités.

TAYAL se distingue par son intégration verticale : l'entreprise assure un processus complet

« De la fibre au produit fini », en maîtrisant les étapes de filature, tissage/tricotage, teinture/finissage et confection.

Le logo de TAYAL se distingue par l'image du fennec, un animal emblématique de l'Algérie. Ce logo est coloré aux couleurs du drapeau algérien : le blanc, le vert et le rouge, ainsi que le noir, qui symbolise le groupe TAY.



Figure 1.2 : Géolocalisation de l'entreprise TAYAL SPA

2.1 Capacités

Sur le plan capacitaire, l'entreprise affiche des volumes industriels importants : 36 000 tonnes/an de fil, environ 55 millions de mètres/an de tissu tissé, 5 500 tonnes/an de tissu tricoté, ainsi qu'une capacité de confection d'environ 30 millions d'articles/an ; la consommation de coton à pleine capacité est estimée à 40 000 tonnes/an.

En matière d'ouverture à l'international, l'entreprise a engagé ses exportations à partir de 2018 (avec un volume de 750 tonnes sur la période allant jusqu'à la fin de l'année 2018) et communique une progression notable des exportations sur les années suivantes [51].

2.2 Mission

Assurer à l'Algérie une position stratégique sur le marché mondial du textile et du prêt-à-porter en adoptant des techniques de gestion modernes et des méthodes de production respectueuses de l'environnement, au sein d'installations à la pointe de la technologie.

TAYAL se positionne comme un partenaire de confiance pour les marques émergentes, en offrant des solutions sur mesure et en personnalisant chaque création pour répondre aux exigences spécifiques de ses clients [51].

2.3 Produits



Figure 1.3 : Gamme de produits de l'entreprise

2.3.1 Fil

TAYAL utilise principalement la fibre de coton, avec une consommation annuelle annoncée d'environ 40 000 tonnes, issue de plusieurs origines.

La capacité théorique de filature atteint 36 000 tonnes/an, couvrant une large plage de titrages de 6 Ne à 80 Ne ($\approx$ Nm 10 à Nm 135). L'outil industriel comprend 101 184 broches et 2 700 rotors Open-End.

Sur le plan des familles de produits, l'entreprise fabrique notamment des fils :

- Ring cardé, peigné, compact, Open-End, ainsi que des fils retors (plied).
- Des variantes techniques existent également (ex. core spun, slub, dual core, fils teints).

Les mélanges communiqués incluent principalement :

- 100% coton, poly/coton, coton/viscose, coton/lin, coton/tencel.

2.3.2 Tissu

La production textile est structurée autour de deux grands segments :

a) Tissu tissé

La capacité annoncée est de 54 millions de mètres/an, avec un parc de 456 métiers Picanol Optimax-I.

b) Tissu tricoté

La capacité atteint 5 500 tonnes/an, assurée par 48 machines Mayer & Cie.

Principales familles de tissus :

- Denim : environ 36 millions de mètres/an, avec 2 machines Karl Mayer dédiées à la teinture “rope dyeing”. Les paramètres annoncés couvrent un grammage de 5 à 15 oz ($\approx$ 170 à 500 g/m²), une élasticité jusqu'à 90% et des largeurs pouvant atteindre 200 cm (rigide) et 170 cm (stretch).
- Gabardine & shirting : environ 18 millions de mètres/an, avec des grammages annoncés entre 100 et 500 g/m².
- Maille (knitted fabric) : types cités tels que single jersey, interlock, ribana, fleece (2/3 fils), jacquard, piqué, ainsi que des tissus intégrant de l'élasthanne.

2.3.3 Prêt-à-porter

La capacité annuelle de confection est annoncée à 30 millions d'articles/an.

La répartition indicative communiquée est la suivante :

- Denim & non-denim : 12 millions de pièces/an (pantalons, vestes, shorts, workwear, etc.).
- Chemises : 6 millions de pièces/an (manches courtes/longues, tabliers, tenues traditionnelles, etc.).
- Maille / knitwear : 12 millions de pièces/an (T-shirts, polos, sweatshirts, jogging, sous-vêtements, etc.).

2.4 Unités de l'entreprise

Le complexe industriel de TAYAL est organisé en plusieurs blocs fonctionnels, chacun dédié à une activité précise (production, support, services). Cette structuration permet une gestion rationnelle des flux, une meilleure coordination entre les ateliers et une organisation industrielle intégrée.

- Unité 02 : Entrepôt de coton
- Unités 03 et 04 : Deux unités de filature
- Unité 05 : Unité indigo et préparation de chaîne
- Unité 06 : Unité de tricotage
- Unité 07 : Unité énergétique de tri-génération
- Unités 08 et 09 : Deux unités de tissage
- Unité 10 : Cuisine et réfectoire
- Unité 11 : Unités de teinture et de finissage
- Unité 12 : Station de traitement des eaux usées
- Unités 21, 22 et 23 : Unités de confection et laverie industrielle

- Unité 25 : Hébergement
- Unité 26 : Bâtiments administratifs
- Unité 27 : Centre d'application (confection)
- Unité 28 : Centre de formation
- Unité 29 : Vestiaires, infirmerie et salles de prière

3. Processus de production textile chez TAYAL

La production commence à l'unité 02, dédiée au stockage des fibres naturelles et synthétiques (matières premières).

Les fibres sont ensuite envoyées vers les unités 03 et 04 (filature) pour être transformées en différents types de fils.

Les fils sont dirigés vers les unités de fabrication du tissu : unité 06 (tricotage) et unités 08–09 (tissage), pour obtenir des tissus tricotés et tissés (denim et non-denim), avec une préparation spécifique assurée par l'unité 05.

Le tissu est ensuite traité à l'unité 11 (teinture et finissage).

Enfin, la production se termine par la confection du prêt-à-porter dans les unités 21, 22 et 23.

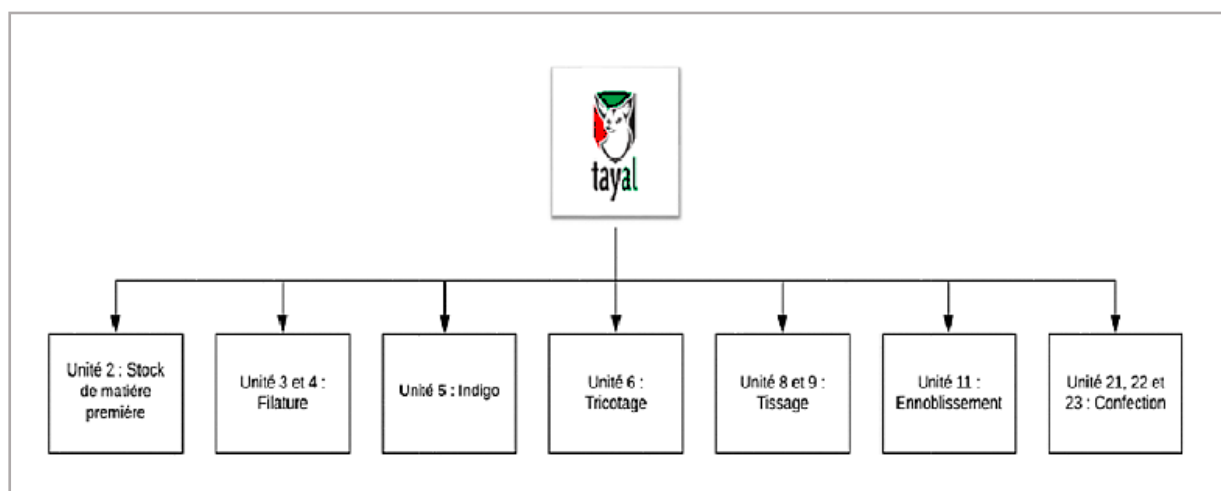


Figure 1.4 : Unités de production de l'entreprise

3. 1 Stockage de matière première (Unité 2)

L'unité 2 est chargée de la réception des matières premières (coton, viscose, polyester, lin, Tencel, etc.) et de leur contrôle afin de vérifier leur conformité en qualité et en quantité. Après validation, les fibres sont stockées par type et par lots pour assurer l'alimentation continue des unités de filature 3 et 4, considérées comme ses clients internes.

Cette unité assure également la gestion des déchets et produits rejetés provenant des unités de filature : selon leur état, ils peuvent être valorisés (vendus) ou éliminés.



Figure 1.5 : Unité 2 de stockage de la matière première

Enfin, le déstockage est réalisé en fonction des besoins de production, des exigences clients et de l'état de conservation des matières disponibles suivant la règle FIFO [4].

3.2 Unités de filature (unités 3 et 4)

L'opération de filature consiste à transformer les fibres (naturelles ou synthétiques) en fils multibrins, retors, câblés ou guipés, en suivant les étapes suivantes :

3.2.1 Ouvraison

Les fibres sont livrées comprimées sous forme de balles.

Ces balles sont ouvertes pour former des flocons.



Figure 1.6 : Unité 3 de filature

3.2.2 Épuration et nettoyage

Cette étape vise à éliminer un maximum d'impuretés et de corps étrangers des fibres.



Figure 1.7 : machine d'épuration Uni clean



Figure 1.8 : machine d'épuration Uni mix

3.2.3 Cardage

Les flocons de fibres sont brossés mécaniquement pour :

- Individualiser et paralléliser les fibres.
- Éliminer les dernières impuretés.

À la fin de cette étape, on obtient un ruban de fibres régulier (nappe de fibres).

Des fibres d'origines différentes peuvent être mélangées.



Figure 1.9 : Carding machine

3.2.4 Doublage

Plusieurs passages permettent d'améliorer la parallélisation des fibres et de régulariser la grosseur du ruban.

À cette étape, différents rubans peuvent être mélangés.



Figure 1.10 : Banc d'étirage 01

3.2.5 Peignage

Cette étape consiste à :

- Séparer les fibres longues des courtes.
- Éliminer les débris restants.

Elle permet d'obtenir un ruban de fibres longues, idéal pour les étapes suivantes.



Figure 1.11 : Combing (Peigneuse)

3.2.6 Étirement

Les rubans sont étirés et affinés pour être présentés sous forme de fromages ou de rondelles.



Figure 1.12 : Banc d'étirage 02

3.2.7 Roving (Banc à broche)

Le produit obtenu à cette étape, appelé mèche, est un brin de fibres moins dense qu'un ruban.

Une légère torsion est appliquée pour maintenir les fibres ensemble.

La mèche est enroulée sur un support adapté pour alimenter les machines de filature Ring.



Figure 1.13 : Banc à broche

3.2.8 Filature Ring (Ring Spinning)

La machine Ring Frame réalise les opérations suivantes :

Étirement : La mèche est étirée jusqu'à atteindre la finesse souhaitée.

Torsion : Le brin est tordu pour former un fil répondant aux exigences de résistance et de numéro.

Enroulage : Le fil torsadé est enroulé sur une bobine pour un stockage et un transport facilité.



Figure 1.14 : Ring Spinning

3.2.9 Bobinage (Cone Winding)

Dernière étape de la filature, elle sert à enrouler le fil sur des cônes pour les prochaines étapes de fabrication.

Ce processus d'emballage est essentiel pour assurer la continuité entre la fabrication du fil et celle du tissu.

Une mauvaise gestion du bobinage peut entraîner des pertes financières importantes et nuire à la fidélisation des clients.



Figure 1.15 : Bobinage

3.2.10 Types de fils produits

L'unité de filature fabrique quatre types de fils, différenciés par leur qualité. Bien que les processus soient similaires, de légères variations influencent la qualité finale.

— Compact /Combed

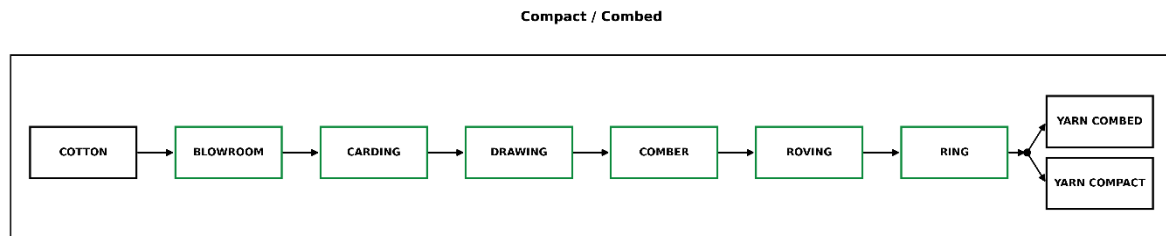


Figure 1.16 : Combed

— Carded

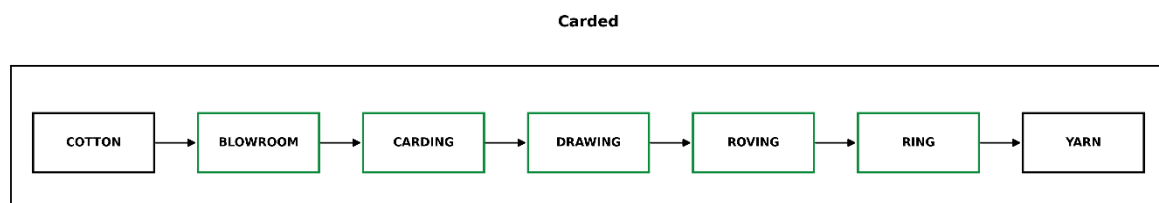


Figure 1.17 :Carded

— Open-end

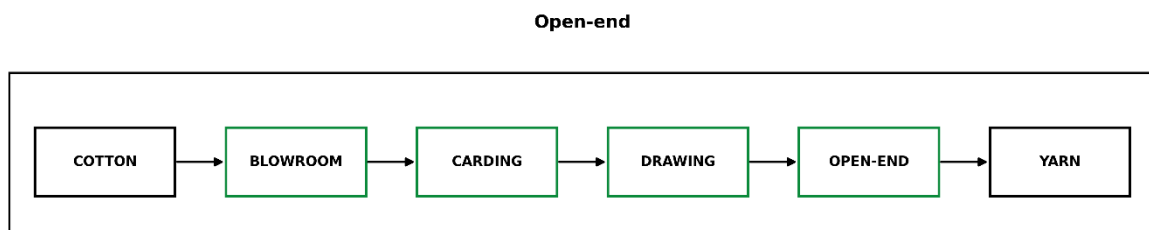


Figure 1.18 : Open-end

Ce processus est essentiel pour garantir une production organisée et efficace de vêtements prêt-à-porter. Une connaissance approfondie des matières premières et de leur transformation est indispensable pour optimiser la qualité et la production.

3.3 Indigo (Unité 5)

L'unité 5 est dédiée à la teinture des fils de chaîne pour la production de tissus denim, notamment en utilisant le colorant indigo. Cette étape est essentielle pour obtenir la couleur caractéristique des jeans et autres tissus similaires.

3.3.1 Teinture en Corde (Rope Dyeing)

Les fils de chaîne issus du procédé Ball Warping sont regroupés en cordes.



Figure 1.19 : machine de rope dyeing

Ces cordes passent successivement par plusieurs bacs de teinture contenant de l'indigo, ainsi que des produits chimiques spécifiques pour renforcer la couleur et ses propriétés.



Figure 1.20 : dyeing machine (Box 6 – Box 13)

Après chaque bain, les cordes sont exposées à l'air, ce qui permet à l'indigo de s'oxyder et de fixer sa teinte bleue caractéristique.



Figure 1.21 : zone d'oxydation

Le steamer sert à fixer la teinture sur le fil grâce à l'action de la vapeur et d'un temps de traitement contrôlé, afin de stabiliser la couleur.



Figure 1.22 : STEAMER

3.3.2 Séchage

Les cordes teintées passent par un dispositif de séchage afin de stabiliser la teinture.



Figure 1.23 : Dryer 1



Figure 1.24 : Dryer 2

3.3.3 Réorganisation des Fils (Rebeaming)

Une fois teintes, les cordes sont démêlées et transformées en feuilles de fils alignés, prêtes pour les étapes suivantes de préparation au tissage.



Figure 1.25 : machine de rebeaming

L'unité comprend deux chaînes de production en parallèle, assurant les mêmes opérations pour augmenter la capacité et garantir la continuité de production.

À TAYAL SPA, l'Unité 5 (Indigo) est une étape centrale de la filière denim, car la teinture à l'indigo donne au tissu son aspect caractéristique et participe directement à sa valeur ajoutée. Cette opération s'insère dans l'organisation intégrée du groupe, de la fabrication des fils jusqu'au produit fini.

3.4 Tissage (unité 8 et 9)

Dans cette étape, les fils sont convertis en tissu selon des procédés de fabrication adaptés au produit visé. La production se répartit en trois grandes familles : denim, non-denim et tissu pour chemise.

Le schéma associé présente l'enchaînement des étapes nécessaires à la réalisation de chaque type de tissu, tout en soulignant les spécificités techniques propres à chaque procédé. Les principales opérations de production sont présentées ci-après.

Denim :

- Les fils de chaîne sont teints à l'indigo par le procédé de Rope Dyeing.
- Ils subissent un encollage pour protéger les fils lors du tissage.
- Le tissage donne au denim sa structure robuste.

Non-Denim :

- Les fils ne sont pas teints mais alignés et préparés par un procédé de Beam Warping.
- L'encollage est également utilisé pour assurer leur résistance.

Chemise :

- Les fils suivent un processus similaire au non-denim, mais les tissus produits sont généralement plus légers et adaptés pour des vêtements comme les chemises.

Ces opérations sont adaptées à chaque type de tissu pour garantir les qualités spécifiques nécessaires, comme la durabilité pour le denim ou la légèreté pour les chemises.

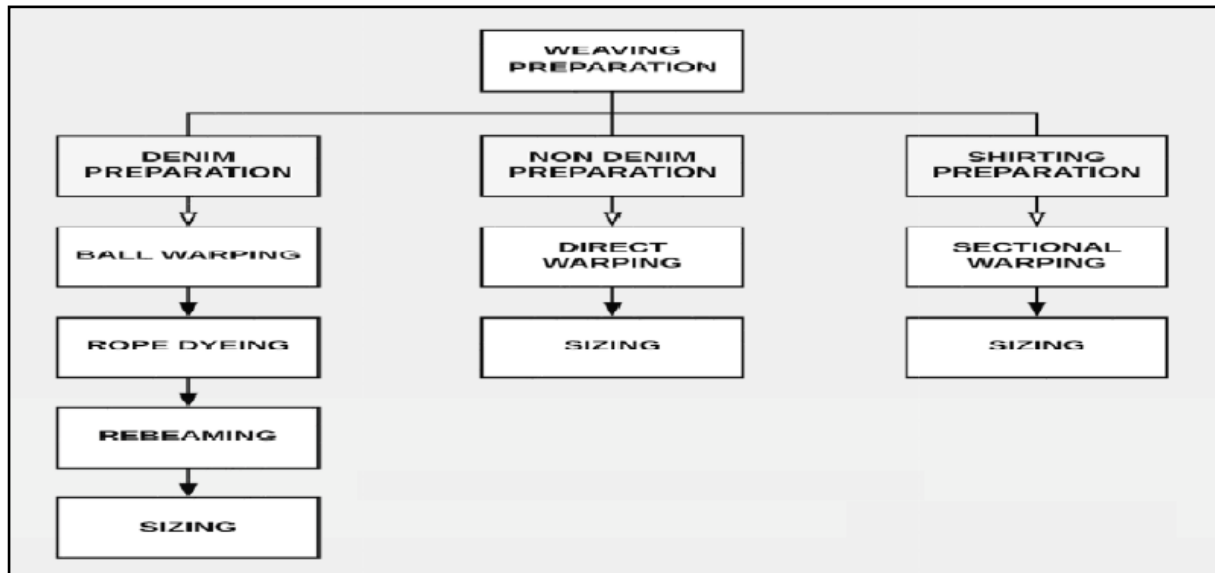


Figure 1.26 : Etape de préparation pour le Tissage

3.4.1 Préparation du tissu non-denim (Direct Warping)

Le Direct Warping consiste à enrouler directement les fils de chaîne sur un cylindre, sans les traiter par teinture, pour les préparer au tissage. Ce processus aligne les fils parallèlement et les place sous forme de feuille ouverte, prêts pour l'étape de sizing.

3.4.2 Préparation du tissu shirting (Sectional Warping)

Le Sectional Warping consiste à enrouler les fils de chaîne sur un cylindre en sections, chaque section étant enroulée séparément. Ce processus permet d'obtenir une distribution uniforme des fils, idéale pour les tissus légers comme les chemises. Les fils sont ensuite prêts pour le traitement de sizing.

3.4.3 Encollage (Sizing)

Cette étape consiste à appliquer un revêtement protecteur sur les fils de chaîne non teints pour les rendre plus résistants à l'abrasion et limiter la formation de pilosité pendant le tissage.

Le fil denim, déjà teint à l'indigo dans l'unité 5, entre directement dans l'unité de 6 pour recevoir un traitement protecteur sizing.

3.5 Tricotage (unité 6)

Il existe deux principaux types de tricotage, distingués par la manière dont les mailles sont reliées :

- Tricotage à mailles cueillies : Utilise un seul fil provenant d'une bobine pour former des boucles, et travaille dans le sens des rangées (largeur de l'étoffe).

- Tricotage à mailles jetée : Utilise simultanément 2000 à 3000 fils provenant d'ensouples, et travaille dans le sens des colonnes (longueur de l'étoffe).

3.5.1 Machines de tricotage

Les machines de tricotage sont spécialisées pour un type de produit et sont caractérisées par :

- Système : Nombre de bobines utilisées pour le fonctionnement de la machine.
- Diamètre (inch) : Diamètre du cylindre de la machine.
- Gauge (Fine) : Nombre d'aiguilles par pouce.

Chaque machine est dédiée à un type spécifique de tricotage, en fonction de ces paramètres.



Figure 1.27 : Unité 6 de tricotage

3.5.2 Les types de tissus tricotés

3.5.2.1 Machine circulaire

- Le single jersey comprend plusieurs types de tissus, notamment le suprême, le piqué, le two thread et le three thread.
- Le double jersey, quant à lui, inclut des structures comme l'interlock, la ribana et le jacquard.

3.5.2.2 Machine rectiligne

- Collar
- Cuff

Le processus de production débute par la création du tissu témoin, qui est contrôlé par le service qualité pour détecter d'éventuels défauts. Si le tissu est conforme, la production peut commencer. Sinon, le fil est remplacé.

Une fois l'échantillon témoin validé, l'opérateur de tricotage veille au bon fonctionnement de la machine et signale toute anomalie. L'ingénieur prépare le tableau de bord et les fiches de production spécifiques pour chaque machine.

Chaque rouleau produit est contrôlé par l'opérateur qualité pour minimiser les défauts.

Les produits conformes sont classés en première qualité, tandis que les non conformes sont classés en deuxième qualité ou déchet.

Les produits finis sont stockés dans des zones désignées par le superviseur des stocks, et chaque rouleau est étiqueté conformément à la procédure de gestion de stock.

3.6 Teinture (unité 11)

L'unité 11 se compose de plusieurs sous-unités essentielles :

- Un dépôt de produits chimiques.
- Un dépôt pour les rouleaux de tissus finis (tissus tissés et tricotés).
- Une unité de teinture pour le denim, le non-denim et les tissus tricotés (neiting).
- Un laboratoire pour les analyses physiques et chimiques.
- Un bureau de maintenance électrique et mécanique.

3.6.1 Processus de teinture et finition des tissus tricotés

3.6.1.1 Planification de la teinture

La teinture est organisée selon un "plan de lot", prenant en compte le numéro de lot, la couleur, la largeur et le style. Une préparation rigoureuse est essentielle, car tout défaut peut affecter la qualité du tissu teint.

3.6.1.2 Étapes du processus d'ennoblissement

- **Blanchiment** : Nettoie le tissu des salissures (graisses, cires, huiles, etc.) pour le préparer à la teinture. Une fois le traitement terminé, le tissu est enveloppé dans du plastique et laissé reposer pendant 24 heures pour fixer la couleur.



Figure 1.28 : machine de blanchiment



Figure 1.29 : emballage de rouleaux avec

- **Teinture** : Coloration du tissu :

Discontinue (petits lots) : Par épuisement.

Continue (grands lots) : Par imprégnation.

L'unité dispose de 16 machines de teinture, adaptées à des capacités de 25 kg à 1200 kg.

Le denim est reteint pour obtenir la couleur souhaitée, puis lavé pour fixer la teinte. Ce processus garantit la stabilité et la durabilité de la couleur sur le tissu.



Figure 1.30 : machine de la teinture de non denim



Figure 1.31 : machine de la teinture de denim

- **Lavage :** Lave le tissu pour qu'il prenne son état final.



Figure 1.32 : machine de lavage

- **Essorage :** Élimine partiellement l'eau par compression à l'aide de machines spécifiques pour les tissus tubulaires et open width.



Figure 1.33 : essorage

- **Séchage :** Supprime l'eau résiduelle en chauffant le tissu et contrôle le rétrécissement.



Figure 1.34 : Machine de Séchage

- **Sanforisage :** Garantit la stabilité dimensionnelle (4-5%) et améliore la douceur et l'aspect du tissu, conformément aux normes internationales.



Figure 1.35 : Machine de Sanforisage

3.6.1.3 Contrôle de qualité

Après chaque étape, un contrôle qualité est réalisé pour évaluer les défauts et classer le tissu selon son niveau de qualité.

3.7 Unité de Confection (Unité 23)

L'unité de confection, appelée également unité 23, représente l'une des étapes principales dans la chaîne de production de Tayal. Elle est spécialisée dans la transformation des tissus en produits finis de prêt-à-porter, tels que les polos, les t-shirts, les chemises, les articles basiques et les articles casual.

Cette unité intervient après la fabrication du tissu. En effet, le tissu utilisé comme matière première provient principalement de l'unité 11 de Tayal. Ainsi, l'unité 23 peut être considérée comme un client interne de l'unité de production du tissu. En revanche, les accessoires nécessaires à la confection, comme les fils, les boutons, les étiquettes, les zips et les éléments d'emballage, sont généralement achetés auprès de fournisseurs externes.



Figure 1.36 : Unité 23 de Confection

La production au sein de l'unité 23 est réalisée selon les exigences des clients. Chaque commande possède ses propres caractéristiques, notamment le modèle, la quantité, les tailles, les couleurs, les accessoires, les délais et les exigences de qualité. Pour cette raison, l'unité fonctionne selon une organisation de type job shop, caractérisée par une production organisée en ateliers spécialisés.

L'unité fonctionne également selon une logique de flux tiré, car la production est lancée à partir d'une demande client validée. Cette organisation permet de réduire les stocks inutiles, de limiter les encours de production et d'optimiser l'utilisation des ressources. Elle peut aussi intégrer certains principes du Juste-à-Temps, notamment à travers la préparation des matières et des accessoires selon les besoins réels de la production.

Actuellement, l'unité 23 fonctionne avec un seul shift, ce qui rend la planification très importante pour assurer le respect des délais et la bonne affectation des commandes aux lignes de confection.

3.7.1 Organisation générale de l'unité de confection

L'unité 23 est composée de plusieurs départements et ateliers qui interviennent de manière coordonnée pour transformer une commande client en produit fini prêt à être expédié.

Les principaux départements et ateliers intervenant dans le processus sont :

- le département merchandising ;
- le département planning ;
- le département méthodologie ;
- la Sample Room ;
- l'entrepôt des matières premières et accessoires ;
- l'atelier de découpe ;
- l'atelier d'impression et/ou de broderie ;
- l'atelier de couture ;
- la zone de repassage, contrôle et emballage ;
- le contrôle qualité ;
- le département logistique.

Chaque département joue un rôle précis dans le traitement de la commande. La bonne coordination entre ces services permet d'assurer la disponibilité des matières, le lancement de la production dans les délais, la conformité du produit et la préparation de la commande pour l'expédition.

3.7.2 Traitement de la commande client

Le processus de production commence par une demande exprimée par le client. Cette demande concerne généralement un modèle précis, une quantité à produire, des tailles, des couleurs, des accessoires particuliers et un délai de livraison souhaité.

3.7.2.1 Réception de la demande par le service commercial

Le service commercial constitue le premier point de contact avec le client. Il reçoit la demande, discute les conditions générales et collecte les informations nécessaires concernant le produit demandé.

Après la validation des principaux éléments avec le client, notamment le modèle, la quantité, le prix et le délai, le client dépose un bon de commande auprès du service commercial.

3.7.2.2 Intervention du département merchandising

Après la réception du bon de commande, le service commercial transmet les informations au département merchandising. Le merchandiser assure la coordination entre le client, le service commercial, la planification et les autres départements concernés par la production.

À cette étape, le merchandiser crée un document interne appelé SO, c'est-à-dire un Sample Order. Ce document contient les informations principales liées à la commande et sert de base pour la suite du traitement.

3.7.2.3 Création de l'ordre de production

Le Sample Order est ensuite transmis au département planning. À partir de ce document, le planificateur crée un ordre de production, appelé aussi PO.

Cet ordre de production contient toutes les informations nécessaires pour organiser la fabrication, notamment :

- le modèle demandé ;
- le style code ;
- les quantités à produire ;
- la répartition par taille ;
- la répartition par couleur ;
- les exigences spécifiques du client ;
- les matières premières nécessaires ;
- les accessoires demandés ;

Le PO devient ainsi le document de référence pour le lancement et le suivi de la production.

3.7.3 Rôle du département planning

Le département planning joue un rôle essentiel dans l'organisation de la production. Il permet de transformer la commande client en un programme de production réalisable, en tenant compte des capacités disponibles et des délais à respecter[8].

Le planificateur affecte chaque ordre de production à une ou plusieurs lignes de confection. Cette affectation dépend de plusieurs paramètres, tels que :

- la catégorie du produit ;
- le modèle à fabriquer ;
- la quantité demandée ;
- le temps standard de production ;
- la disponibilité des lignes ;
- les compétences des opérateurs ;
- la charge actuelle de production ;
- le délai exigé par le client.

Lorsqu'une commande est importante, elle peut être répartie sur plusieurs lignes en parallèle. De même, la fabrication d'un ordre de production peut s'étaler sur plusieurs jours selon la quantité planifiée par jour.

Le planning permet donc d'assurer une meilleure utilisation des lignes de confection et d'éviter les retards dans le traitement des commandes.

3.7.4 Rôle du département méthodologie

Le département méthodologie intervient avant le lancement de la production. Son rôle consiste à définir les méthodes de production adaptées à chaque modèle.

Pour chaque article, ce département analyse les opérations nécessaires, les machines à utiliser, le nombre d'opérateurs requis et le temps standard de production. Il prend également en considération les exigences du client et les standards internes de l'entreprise.

Chaque modèle possède ses propres caractéristiques techniques. Il peut se différencier par :

- le nombre d'opérations ;
- le opérations de couture ;
- les machines nécessaires ;
- le nombre d'opérateurs ;
- le temps standard ;
- les accessoires utilisés ;
- le niveau de difficulté du modèle.

Ces informations sont importantes pour aider le département planning à affecter chaque PO à la ligne la plus adaptée.

3.7.5 Préparation des matières premières et des accessoires

Avant le lancement effectif de la production, les matières premières et les accessoires doivent être préparés.

Le tissu, qui constitue la matière principale, provient de l'unité 11 de Tayal. Il est ensuite mis à disposition de l'atelier de découpe. En parallèle, l'équipe chargée des accessoires prépare les éléments nécessaires à la commande selon les quantités demandées par le planning.

Une fois le tissu découpé et les accessoires préparés, l'ensemble est regroupé dans des chariots dédiés. Ces chariots sont ensuite placés devant les lignes de production concernées. Ils représentent les entrées principales de la ligne de confection.

3.7.6 Flux de production dans l'unité 23

Le flux de production dans l'unité 23 suit plusieurs étapes successives. Chaque étape contribue à la transformation progressive de la matière première en produit fini.

Le flux général peut être présenté comme suit :

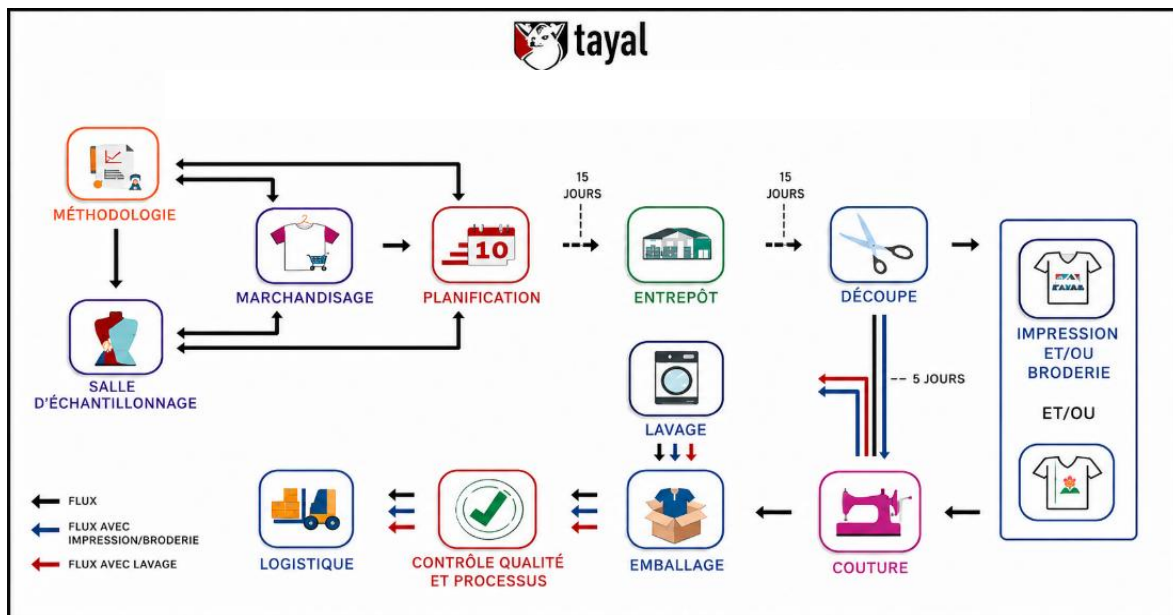


Figure 1.37 : Flux de production de l'unité 23

3.7.6.1 Salle d'échantillonnage

La Salle d'échantillonnage est chargée de réaliser les échantillons correspondant aux modèles demandés par les clients. Ces échantillons servent de prototypes avant le lancement de la production en série.

Cette étape permet de vérifier que le produit correspond bien aux attentes du client en termes de modèle, de dimensions, de finition et de qualité. Après validation de l'échantillon, la commande peut passer aux étapes de préparation et de production.

3.7.6.2 Entrepôt des matières premières et accessoires

L'entrepôt assure la réception, le stockage et la gestion des matières premières et des accessoires. Il garantit leur disponibilité au bon moment afin d'éviter les retards dans la production.

Cette étape est importante, car l'absence d'un tissu ou d'un accessoire peut empêcher le lancement de la commande ou provoquer un arrêt de production.

3.7.6.3 Découpe du tissu

L'atelier de découpe prépare les pièces nécessaires à la confection du produit. Le tissu est découpé selon le patron validé et selon les spécifications du modèle demandé par le client.

Après la découpe, les pièces sont contrôlées, identifiées et étiquetées. Cette identification permet de différencier les commandes, les clients, les tailles et les modèles.

Les pièces coupées sont ensuite placées dans des chariots. Selon le besoin du produit, elles peuvent être envoyées directement vers les lignes de couture ou passer d'abord par l'atelier d'impression ou de broderie.

3.7.6.4 Impression et broderie

Lorsque le modèle demandé comporte un logo, un motif, une broderie ou un design spécifique, les pièces coupées sont orientées vers l'atelier d'impression ou de broderie.

Cette opération est réalisée avant la couture afin d'appliquer les éléments décoratifs ou personnalisés conformément aux exigences du client.

Après cette étape, les pièces sont regroupées de nouveau dans des chariots et préparées pour entrer dans les lignes de confection.

3.7.6.5 La confection

La couture représente l'étape principale de transformation des pièces coupées en articles assemblés. Les pièces préparées sont placées devant les lignes de confection avec les accessoires nécessaires.

Actuellement, l'atelier de couture de l'unité 23 est composé de 20 lignes de confection fonctionnant en parallèle. Ces lignes sont organisées selon les catégories de produits fabriqués.

La répartition actuelle est la suivante :

- les lignes 1 à 7 sont dédiées aux modèles polo ;
- les lignes 8 à 15 sont dédiées aux modèles basiques ;
- les lignes 16 à 20 sont dédiées aux modèles casual.

Cette organisation permet de spécialiser les lignes selon les familles de produits. Les opérateurs affectés à chaque groupe de lignes maîtrisent mieux les opérations liées à leur catégorie et chaque ligne fonctionne comme une chaîne de production.

Le produit passe par plusieurs postes successifs, et chaque opérateur réalise une opération précise. À la fin de la ligne, les différentes pièces sont assemblées pour obtenir un article complet.

L'affectation d'un ordre de production à une ligne dépend donc de la catégorie du produit, du modèle, des compétences des opérateurs, de la disponibilité des lignes, du temps standard et du délai demandé par le client.

3.7.6.6 Le Repassage

Après la couture, les articles passent par l'étape de repassage. Cette opération permet de préparer les vêtements avant le contrôle final et l'emballage.

Le repassage améliore l'aspect visuel du produit et facilite la détection des défauts éventuels lors du contrôle qualité.



Figure 1.38 : lignes de repassage

3.7.6.7 Le Contrôle qualité

Le contrôle qualité est une étape essentielle dans le processus de production. Il permet de vérifier la conformité des articles par rapport aux exigences du client et aux standards de l'entreprise.

Les contrôles peuvent être réalisés à plusieurs niveaux : après la découpe, après l'impression ou la broderie, pendant la couture et avant l'emballage.

Après la couture et le repassage, les articles sont contrôlés pièce par pièce. Ils sont ensuite classés selon plusieurs catégories :

- Premier choix : articles conformes et validés ;
- Deuxième choix : articles présentant des défauts mineurs pouvant être retouchés ;

Les articles classés en deuxième choix peuvent être retouchés afin d'être récupérés et reclassés en premier choix lorsque cela est possible.

3.7.6.8 Emballage

Après le contrôle qualité, les articles conformes sont emballés selon les exigences du client. L'emballage peut varier selon le type d'article, la quantité par carton, les tailles, les couleurs et les instructions spécifiques du client.

Les articles sont placés dans des chariots identifiés par couleur afin de faciliter le suivi :

- chariot vert : articles conformes et validés ;
- chariot rouge : articles contrôlés mais non validés ;
- chariot jaune : articles en cours de contrôle.

Cette organisation visuelle facilite la circulation des produits et permet de distinguer rapidement l'état d'avancement des articles dans la zone de finition.

3.7.6.9 Logistique et expédition

Après l'emballage, les produits finis sont transférés vers le département logistique. Ce département est responsable du comptage, du tri, de l'étiquetage et de la préparation des cartons.

La logistique vérifie les quantités produites et les compare avec les quantités demandées par le client. Elle s'assure également que les articles sont emballés selon les exigences définies, notamment le nombre de pièces par carton, les références, les tailles et les couleurs. Les cartons sont ensuite stockés temporairement afin d'être enregistrés dans le système d'information de l'entreprise. Après cette étape, ils sont transférés vers le stock final pour être expédiés au client.

4. Conclusion

TAYAL S.P.A. se présente comme un acteur industriel intégré et structuré, maîtrisant l'ensemble de la chaîne de valeur textile, De la réception des fibres jusqu'à l'expédition des articles finis, chaque étape s'enchaîne de manière cohérente au sein d'un complexe organisé et fonctionnel. Cette maîtrise du cycle complet de production confère à l'entreprise une certaine autonomie et lui permet de répondre directement aux attentes de ses clients. C'est dans ce contexte industriel précis que s'inscrit la suite de ce travail.

CHAPITRE 02 : INTELLIGENCE ARTIFICIELLE : CONCEPTS FONDAMENTAUX ET MÉTHODES POUR LA PRÉDICTION

1. Introduction

Dans un environnement industriel, le pilotage d'une ligne de production repose sur des indicateurs de performance KPI) qui permettent de suivre l'efficacité, la qualité et le respect des objectifs. La normalisation de ces KPI aide à garantir une lecture cohérente et comparable des résultats, ce qui facilite l'analyse et le reporting au sein de l'entreprise.

Avec la multiplication des données issues de la production et des ressources, l'enjeu n'est plus seulement de mesurer, mais aussi d'expliquer et d'anticiper les écarts. L'intelligence artificielle, en particulier l'apprentissage automatique, apporte des méthodes capables d'apprendre à partir de l'historique pour produire des prévisions et soutenir une prise de décision plus rapide et plus objective. Dans la pratique industrielle actuelle, la partie la plus utilisée est l'apprentissage automatique (machine learning) : on entraîne un modèle sur des données historiques afin qu'il puisse prédire un résultat futur à partir de variables explicatives.

Ce chapitre présente donc les définitions et concepts essentiels de l'IA, les principales familles de méthodes utilisées en prédiction, ainsi que leur place dans une démarche structurée allant de la compréhension du besoin à l'exploitation des résultats pour l'aide à la décision.

2. Définition des concepts fondamentaux de l'intelligence artificielle

2.1 Définition de l'intelligence artificielle (IA)

Le terme intelligence artificielle ne renvoie pas à une définition unique : selon les auteurs, l'accent peut être mis sur la création de machines capables d'agir intelligemment, sur la notion d'agent qui perçoit et agit, ou encore sur une définition plus opérationnelle orientée "entrées→sorties"(prédictions,recommandations,décisions)[18].

John McCarthy présente l'IA comme une discipline visant à concevoir des machines capables de comportements intelligents. Il résume l'idée par une formule devenue classique : l'IA est

« la science et l'ingénierie de la création de machines intelligentes ». Cette définition insiste sur l'objectif : construire des systèmes qui accomplissent des tâches habituellement associées à l'intelligence (raisonner, résoudre des problèmes, atteindre des buts).

Dans Artificial Intelligence : A Modern Approach, Russell et Norvig structurent l'IA autour de la notion d'agent : « un agent reçoit des informations de son environnement (perceptions) et choisit des actions » [16]. Ils résument l'IA comme l'étude de ces agents qui perçoivent et agissent.

Cette approche est très utilisée en ingénierie, car elle relie naturellement un système intelligent à un objectif mesurable (performance attendue) et à des décisions à prendre.

L'OCDE propose une définition plus opérationnelle : un système d'IA est un système basé sur une machine qui, pour des objectifs explicites ou implicites, "inférera" à partir d'entrées comment produire des sorties telles que prédictions, contenus, recommandations ou décisions, susceptibles d'influencer un environnement réel/physique ou virtuel [17].

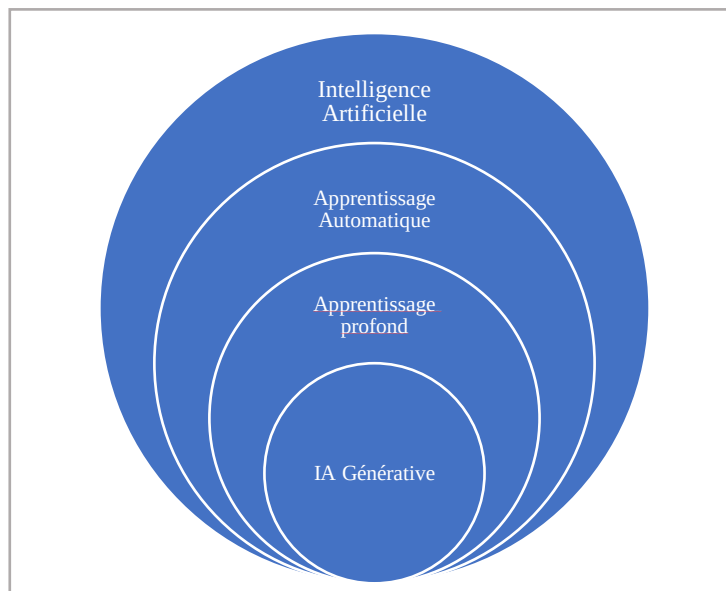


Figure 2.1 : Relations hiérarchiques entre l'Intelligence Artificielle et le Machine Learning

2.2 Définition de l'apprentissage automatique (Machine Learning)

L'apprentissage automatique (ML) est généralement présenté comme une branche de l'IA centrée sur l'idée suivante : au lieu d'écrire toutes les règles "à la main", on laisse le système apprendre à partir d'exemples (données historiques) afin d'améliorer sa performance sur une tâche.

Arthur Samuel est souvent cité parmi les premiers à formaliser l'idée que l'on peut donner à une machine la capacité d'apprendre sans programmer explicitement toutes les règles : le ML est un champ qui donne aux ordinateurs « la capacité d'apprendre sans être explicitement programmés ».

Cette formulation met en évidence l'intérêt industriel du ML : réduire la dépendance à des règles fixes, parfois insuffisantes face à la variabilité réelle.

Mitchell définit le ML comme l'étude d'algorithmes qui s'améliorent automatiquement par l'expérience.

Il propose aussi une définition plus "ingénieur" : un programme apprend à partir d'une expérience E pour une tâche T et une mesure de performance P si sa performance mesurée par P s'améliore avec E .

2.3 Définition du Deep Learning (apprentissage profond)

Le deep learning est une famille de méthodes de ML fondée sur des réseaux de neurones à plusieurs couches, capables d'apprendre des représentations de plus en plus abstraites à partir des données [21] [22].

Dans leur ouvrage de référence, les auteurs décrivent le deep learning comme une forme de ML qui apprend à comprendre les données via une hiérarchie de concepts, construits progressivement à partir de concepts plus simples, ce qui correspond à l'idée de "plusieurs couches" (profondeur) [21].

Un point essentiel dans la littérature : les caractéristiques (features) utiles ne sont pas toujours conçues manuellement ; dans le deep learning, elles sont apprises automatiquement à plusieurs niveaux. Par exemple, dans des tâches perceptives, les couches apprennent progressivement des motifs simples puis des structures plus complexes, sans que l'ingénieur n'écrive explicitement ces motifs.

Cette idée explique pourquoi le deep learning est très performant lorsque les données sont riches et peu structurées (images, texte, audio), et pourquoi la qualité/quantité de données est un facteur clé.

2.4 Définition de l'IA générative (Generative AI)

L'IA générative désigne une classe de modèles dont la finalité n'est pas seulement de classer ou de prédire une valeur, mais de produire un contenu nouveau qui ressemble aux données d'apprentissage.

Le NIST définit l'IA générative comme une classe de modèles d'IA qui reproduisent la structure et les caractéristiques des données d'entrée afin de générer un contenu synthétique dérivé (texte, images, audio, vidéo, etc...).

Sur le plan conceptuel, l'IA générative peut être comprise comme l'apprentissage d'une "forme" de distribution des données afin de pouvoir produire de nouveaux exemples plausibles ; les GAN (Generative Adversarial Networks) illustrent bien cette logique en entraînant un modèle génératif à capturer la distribution des données via un processus adversarial. Enfin, comme ces modèles sont conçus pour générer du contenu, leur utilisation en contexte organisationnel doit s'accompagner d'une attention particulière aux risques et à la fiabilité, ce qui motive l'existence de cadres dédiés à la gestion des risques de l'IA, y compris pour l'IA générative.

2.5 Applications majeures de l'IA

Dans l'industrie, les applications majeures de l'intelligence artificielle visent surtout à convertir des données opérationnelles (production, qualité, maintenance, supply chain, etc.) en décisions plus rapides et plus fiables. Cette évolution explique sa place d'acteur clé : l'IA ne sert plus seulement à analyser le passé, mais aussi à anticiper les écarts, détecter des anomalies et optimiser des actions ayant un impact direct sur la productivité. Cette diffusion est visible dans les usages : l'enquête mondiale de McKinsey indique qu'en 2025, 88% des répondants déclarent un usage régulier de l'IA dans au moins une fonction de l'entreprise.

Concrètement, une première catégorie d'applications est la prédiction (demande, délais, volumes, KPI), utilisée pour réduire l'incertitude et améliorer la planification ; McKinsey rapporte qu'en supply chain, la prévision pilotée par l'IA peut réduire les erreurs de prévision de 20 à 50%, avec des effets possibles sur l'indisponibilité produit (jusqu'à 65%) et des baisses de coûts (entreposage 5 à 10%, administration 25 à 40%). Une deuxième catégorie, très répandue en usine, est la maintenance prédictive, où l'exploitation des historiques ou capteurs permet d'anticiper les défaillances : Deloitte mentionne des ordres de grandeur de +10 à 20% de disponibilité, -5 à 10% de coûts de maintenance et -20 à 50% de temps de planification. Une troisième famille concerne le contrôle qualité et la détection de défauts souvent via vision par ordinateur : le World Economic Forum rapporte, par exemple, un cas

de contrôle en boucle fermée réduisant les rebuts avec 12,5% d'économies de coût matière et un modèle réduisant le taux de défauts de 66% dans un cas industriel.

Au-delà de ces cas d'usage, l'IA est de plus en plus mobilisée pour l'optimisation et pour l'automatisation intelligente des tâches.

Cette création de valeur reste toutefois conditionnée par la qualité des données, l'intégration dans les processus et la maîtrise des risques, d'où l'intérêt de cadres de gouvernance comme le NIST AI Risk Management Framework.

2.6 Pourquoi l'IA devient un acteur clé dans les industries ?

L'intelligence artificielle devient un acteur clé dans les industries d'abord parce que les environnements industriels génèrent aujourd'hui une quantité massive de données issues de sources très diverses : capteurs, systèmes ERP/MES, contrôle qualité, maintenance, logistique ou encore ressources humaines [14]. Là où l'analyse humaine atteint vite ses limites, l'IA permet d'extraire des relations et des régularités à grande échelle, en traitant des volumes d'informations que l'on ne peut pas exploiter efficacement avec des méthodes manuelles ou des règles simples.

Elle est également devenue incontournable parce que les systèmes industriels sont, par nature, complexes : la performance d'une ligne ou d'un atelier dépend simultanément du produit fabriqué, des ressources disponibles, des contraintes de planification, des aléas techniques et parfois même de facteurs externes (variations de demande, délais fournisseurs). Dans ce type de contexte, des règles "si... alors..." deviennent insuffisantes, car elles ne capturent ni les interactions entre variables, ni la variabilité réelle du terrain. L'IA apporte alors des modèles capables d'apprendre ces interactions à partir des données et de s'adapter à des situations multiples [1].

Un autre élément décisif est la capacité de l'IA à transformer des données en décisions. Elle permet d'évoluer d'une logique descriptive "que s'est-il passé ?" vers une logique diagnostique "pourquoi cela s'est-il produit ?", puis vers une logique prédictive "que risque-t-il de se passer?" et enfin prescriptive "quelle action faut-il prendre maintenant ?". Cette progression donne une valeur directe à l'IA, car elle ne se limite pas au constat : elle aide à anticiper et à orienter les actions correctives ou d'optimisation.

L'IA s'impose parce qu'elle améliore le triptyque industriel classique : coûts, qualité et délais. En réduisant les arrêts non planifiés, en détectant plus tôt les défauts et en rendant la planification plus robuste, elle contribue à stabiliser la productivité et à accélérer la prise de décision, ce qui renforce la performance globale de l'organisation [15].

3 Fondements du Machine Learning

3.1 L'apprentissage

L'apprentissage désigne la capacité d'un système à améliorer automatiquement ses performances sur une tâche en s'appuyant sur l'expérience, généralement sous forme de données. Autrement dit, on dit qu'un système apprend lorsqu'il adapte son comportement à partir des données pour produire des résultats de plus en plus précis [19].

3.2 Catégories de problèmes en apprentissage automatique

En apprentissage automatique, on distingue généralement trois grands paradigmes selon la manière dont le modèle apprend à partir des données : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement. Cette distinction dépend principalement de la présence ou non d'une information cible et de la façon dont le système est guidé pendant l'apprentissage.

3.2.1 Apprentissage supervisé

L'apprentissage supervisé consiste à entraîner un modèle à partir de données étiquetées ou labellisées. Concrètement, chaque observation est décrite par un ensemble de variables d'entrée X et associée à une sortie cible Y connue. L'objectif est d'apprendre une fonction f telle que $f(X) \approx Y$, afin de pouvoir prédire correctement la sortie Y pour de nouvelles observations jamais vues.

Dans ce cadre, on prépare et structure les données de manière à fournir au modèle des exemples représentatifs : il apprend progressivement les relations entre les variables d'entrée et la cible, puis généralise ces relations pour produire des prédictions sur de nouveaux cas.

3.2.2 Apprentissage non supervisé

Contrairement au cas supervisé, l'apprentissage non supervisé s'applique lorsque les données ne sont pas étiquetées : il n'existe donc pas de cible Y fournie au modèle. L'objectif n'est pas de prédire une sortie connue, mais plutôt de mettre en évidence la structure des données en identifiant des régularités, des regroupements ou des comportements atypiques.

Ce paradigme est particulièrement utile pour :

- regrouper des observations similaires,
- résumer ou simplifier un grand nombre de variables,
- détecter des points rares ou incohérents.

3.2.3 Apprentissage par renforcement

L'apprentissage par renforcement concerne des situations où un agent doit prendre des décisions successives. L'agent interagit avec un environnement : il choisit une action, observe le résultat, puis reçoit un signal de retour appelé récompense : positive ou négative. À travers des essais successifs, il apprend progressivement une stratégie visant à maximiser une récompense cumulée sur le long terme.

Ce paradigme est adapté aux problèmes de pilotage, contrôle et optimisation dynamique, où la notion de "bonne décision" dépend des conséquences futures.

3.3 Typologie des problèmes et méthodes associées

3.3.1 Apprentissage supervisé

3.3.1.1 Régression

La régression vise à prédire une valeur numérique continue tel que le rendement, productivité, temps opératoire, coût, etc... Le modèle apprend la relation entre des variables d'entrée X et une cible Y afin d'estimer une valeur $\hat{Y}$ la plus proche possible de la valeur réelle [6].

On présente ci-après les algorithmes les plus utilisés pour ce problème et les situations dans lesquelles ils sont généralement pertinents :

- Régression linéaire : Un baseline simple et interprétable ; utile lorsque la relation entre variables et cible est approximativement linéaire.
- Régressions régularisées : améliorent la robustesse et limitent le surapprentissage, surtout en présence de variables nombreuses ou corrélées.
- Arbre de décision : modélise des relations non linéaires via des règles ; interprétable mais peut sur-apprendre si trop complexe.
- Forêt aléatoire : ensemble d'arbres qui améliore la stabilité et la précision ; très utilisée sur des données tabulaires industrielles.
- Gradient Boosting (XGBoost, LightGBM, CatBoost) : souvent parmi les meilleures performances sur données tabulaires ; capture bien les interactions complexes.
- SVR ; Support Vector Regression : utile sur des jeux de données de taille moyenne ; nécessite un réglage précis des hyperparamètres.
- Réseaux de neurones (MLP) : adaptés aux données volumineuses et complexes, mais moins interprétables.

3.3.1.2 Classification

La classification consiste à prédire une classe binaire ou multi-classes, par exemple défaut / non défaut ou faible / moyen / élevé. Le modèle apprend à attribuer une étiquette correcte à partir des variables d'entrée[27] :

- Régression logistique : Un baseline efficace et interprétable, souvent utilisée pour la classification binaire.
- k-plus proches voisins; k-NN : méthode simple basée sur la similarité ; sensible à l'échelle des variables.
- Naive Bayes : rapide et efficace dans certains contextes, notamment lorsque les variables sont relativement indépendantes.
- SVM ; Support Vector Machines : performant pour séparer des classes avec une bonne marge ; utile sur des datasets de taille moyenne.
- Arbres, forêts, boosting : robustes sur données hétérogènes et souvent performants en pratique.

3.3.1.3 Série temporelle (Forecasting)

La prévision temporelle concerne des données indexées dans le temps et vise à prédire des valeurs futures à partir de l'historique [7]. Elle impose de respecter l'ordre chronologique et d'intégrer des informations telles que les retards ; lags, la tendance et la saisonnalité :

- ARIMA / ETS : méthodes statistiques adaptées lorsque la dynamique temporelle tel que la tendance, la saisonnalité est dominante [12].
- Approche Machine Learning : création de variables temporelles ; lags, moyennes glissantes, variables calendrier puis régression avec Random Forest ou Gradient Boosting[27].
- RNN / LSTM : utiles lorsque la dépendance temporelle est forte et que le volume de données est important [13] [25].

3.3.2 Apprentissage non supervisé

3.3.2.1 Clustering (segmentation)

Le clustering regroupe automatiquement des observations similaires en clusters, sans variable cible. Il est utile pour identifier des profils de production, des familles de produits ou des comportements similaires.

On présente ci-après les algorithmes les plus utilisés pour ce problème et les situations dans lesquelles ils sont généralement pertinents :

- K-means : simple et rapide, efficace pour des clusters compacts.
- Clustering hiérarchique : utile pour explorer des regroupements multi-niveaux.
- Mean Shift : basé sur la densité, détecte des zones denses.
- DBSCAN : robuste au bruit, détecte des clusters de forme complexe.
- GMM : probabiliste, adapté aux clusters qui se chevauchent.

3.3.2.2 Réduction de dimension

La réduction de dimension vise à représenter les données avec moins de variables, tout en conservant l'essentiel de l'information. Elle sert à visualiser, réduire le bruit et accélérer certains traitements :

- PCA / ACP : technique linéaire, conserve la variance maximale.
- t-SNE : principalement utilisé pour visualisation 2D/3D.
- UMAP : réduction non linéaire moderne, utile pour visualisation et embeddings.
- Autoencoders : représentation compacte apprise par réseau de neurones.

3.3.2.3 Détection d'anomalies

La détection d'anomalies vise à identifier des observations rares ou incohérentes ; dérives, capteurs défectueux[24], journées atypiques :

- Isolation Forest : efficace sur données tabulaires, isole les points rares[26].
- Local Outlier Factor: basé sur la densité locale, détecte des anomalies relatives au voisinage[26].

- One-Class SVM : apprend la frontière du normal, sensible au paramétrage[24].
- Méthodes robustes basées covariance : utiles lorsque les données suivent une structure statistique stable[24].

3.3.4 Apprentissage par renforcement

L'apprentissage par renforcement traite des problèmes de décision séquentielle, Les algorithmes suivants sont fréquemment mobilisés pour traiter ce problème :

- Q-learning / TD learning : apprennent des valeurs d'action ; adaptés à des environnements discrets.
- Policy Gradients : apprennent directement une politique ; utiles dans des environnements complexes.
- Actor-Critic : combine estimation de valeur et politique, utilisé pour des problèmes continus ou avancés.

4 Fondements de l'apprentissage supervisé pour la prédiction de la performance et l'optimisation de la planification de la production

4.1 L'apprentissage supervisé dans un contexte de logistique et de planification de la production

L'apprentissage supervisé permet d'exploiter les données historiques de production pour prédire des résultats futurs à partir de variables déjà observées. Il repose sur des données étiquetées, où chaque observation est associée à une valeur cible connue. Le modèle apprend ainsi la relation entre les variables d'entrée et la sortie attendue afin de produire des prédictions sur de nouvelles données.

Dans la logistique et la planification de la production, cette approche permet d'anticiper le rendement d'une ligne de production, la quantité réellement produite et la probabilité d'atteinte des objectifs de production. Elle facilite ainsi l'exploitation des ordres de fabrication et des données opérationnelles pour améliorer la planification, réduire les écarts entre prévision et réalisation, et renforcer le pilotage de la production.

Dans le cadre de ce travail, l'apprentissage supervisé s'inscrit donc comme un outil d'aide à la décision, permettant de transformer les données de production en informations prédictives utiles pour ajuster les ressources, mieux organiser les lignes de production et soutenir la performance opérationnelle.

4.2 Réseaux de neurones artificiels

4.2.1 Évolution historique et émergence des réseaux de neurones artificiels

L'émergence des réseaux de neurones artificiels s'inscrit dans l'histoire de l'intelligence artificielle. Le premier modèle mathématique et informatique du neurone biologique est attribué à Warren McCulloch et Walter Pitts en 1943. Par la suite, Donald Hebb a proposé en 1949 une règle simple de mise à jour des connexions, ouvrant la voie aux mécanismes d'apprentissage [5]. Les premiers travaux sur les réseaux de neurones ont ensuite évolué vers des modèles plus élaborés, avant de connaître un regain d'intérêt important au milieu des années 1980 grâce à la redécouverte de l'algorithme de rétro-propagation du gradient [5]. Cette évolution a marqué une étape décisive dans l'utilisation des réseaux pour des tâches d'apprentissage et de prédiction.

Cette trajectoire historique montre que les réseaux de neurones ne constituent pas une méthode isolée, mais le prolongement d'une approche scientifique cherchant à modéliser l'apprentissage à partir des données. Leur essor s'explique notamment par leur capacité à dépasser certaines limites des modèles linéaires simples et à traiter des problèmes pour lesquels la structure de décision est difficile à exprimer analytiquement [5].

4.2.2 Du neurone biologique au neurone artificiel

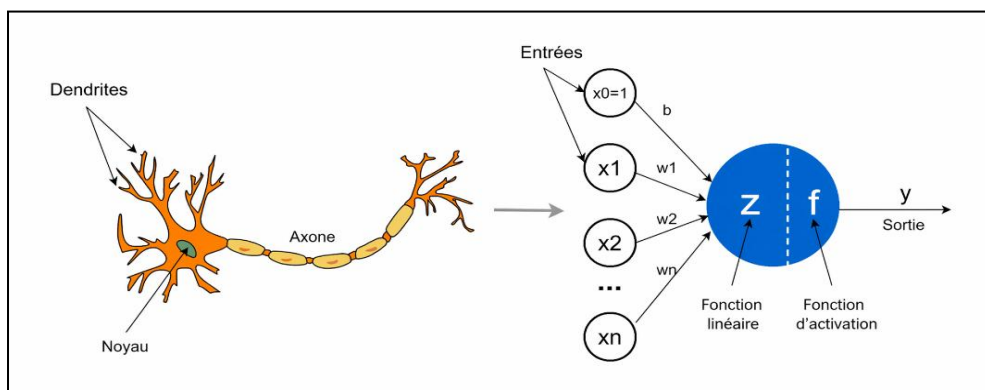


Figure 2.2 : De neurone biologique au neurone artificiel

Les réseaux de neurones artificiels sont inspirés du fonctionnement simplifié du neurone biologique. Le neurone biologique se compose principalement du soma, qui constitue son centre, des dendrites, par lesquelles arrivent les signaux d'entrée, et de l'axone, par lequel est transmis le signal de sortie. La communication entre neurones s'effectue au niveau des synapses[50].

À partir de cette inspiration biologique, le neurone artificiel a été conçu comme une unité de calcul élémentaire. Il reçoit plusieurs valeurs d'entrée $x_1, x_2, \dots, x_n$, chacune associée à un poids $w_1, w_2, \dots, w_n$. Ces poids traduisent l'importance relative de chaque entrée dans la décision du neurone. Le neurone effectue ensuite une combinaison de ces entrées par une somme pondérée. Dans sa forme la plus courante, cette combinaison s'écrit :

$$z = \sum_{i=1}^n x_i w_i + b$$

où b représente le biais [50]. Le biais joue un rôle très important : il permet de décaler la décision du neurone et d'éviter que celle-ci soit toujours centrée sur l'origine. Autrement dit, il donne plus de flexibilité au modèle. Le support sur le perceptron illustre d'ailleurs que le biais peut être traité comme une entrée supplémentaire de valeur constante égale à 1.

Après le calcul de la somme pondérée, le neurone applique une fonction d'activation afin de produire sa sortie. Cette étape est essentielle, car elle transforme une combinaison numérique brute en une réponse exploitable par le modèle.

4.2.3 Structure générale d'un réseau de neurones artificiels

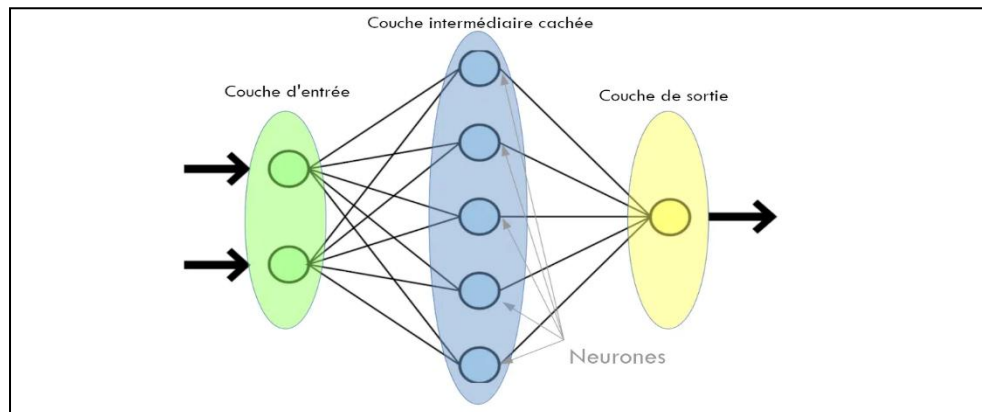


Figure 2.3 : Structure générale d'un réseau de neurones artificiels

Un réseau de neurones artificiels est formé d'un ensemble de neurones organisés en couches. On distingue généralement une couche d'entrée, une ou plusieurs couches cachées, et une couche de sortie. Dans un perceptron multicouche, l'information circule de la couche d'entrée vers la couche de sortie, sans connexion entre les neurones d'une même couche. Les connexions pondérées entre couches successives assurent la transformation progressive des données [5].

Cette organisation en couches permet au réseau de construire des représentations intermédiaires de plus en plus riches. Alors qu'un neurone isolé reste limité dans la nature des relations qu'il peut modéliser, l'empilement de plusieurs couches permet d'exprimer des structures non linéaires beaucoup plus complexes. Le support fourni indique d'ailleurs qu'un perceptron multicouche peut produire des frontières de décision convexes, concaves, ouvertes ou fermées, et qu'il permet d'approximer n'importe quelle fonction [5]. Par extension, cette propriété en fait un outil pertinent pour les problèmes de prédiction, notamment lorsque la variable à estimer dépend de plusieurs facteurs interdépendants.

4.2.4 Principe de la prédiction par apprentissage supervisé

La prédiction par réseau de neurones s'appuie sur le principe de l'apprentissage supervisé. Le réseau est entraîné à partir d'exemples constitués d'un vecteur d'entrée et d'une sortie attendue. Dans le cas du perceptron multicouche, les données d'entraînement sont notées sous la forme $X(t)$ pour les entrées et $y(t)$ pour les sorties désirées, tandis que $s(t)$ représente la sortie effectivement produite par le réseau. L'apprentissage consiste alors à ajuster les poids afin de réduire l'écart entre la sortie attendue et la sortie observée.

Dans cette logique, la prédiction ne résulte pas d'une règle écrite manuellement, mais d'un processus d'ajustement progressif des paramètres internes du réseau. Le modèle apprend donc à partir des données historiques, puis utilise les poids appris pour estimer la sortie associée à de nouvelles observations. Cela permet de capturer des relations complexes entre variables explicatives et variable cible, y compris lorsque ces relations ne sont ni strictement linéaires ni immédiatement visibles.

4.2.5 Le perceptron comme modèle de base de l'apprentissage neuronal

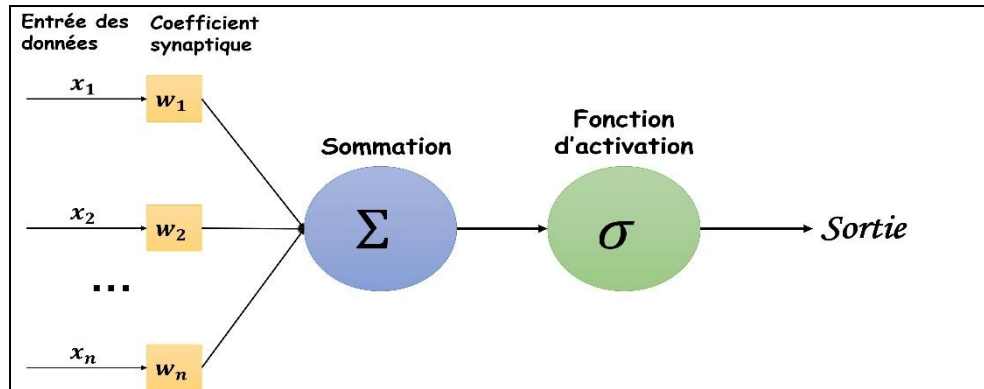


Figure 2.4 : Structure générale d'un perceptron

Le perceptron est l'un des premiers modèles opérationnels de neurone artificiel. Il repose sur une somme pondérée des entrées suivie d'une fonction d'activation seuil. Le support donne la forme suivante :

$$h_w(x) = \text{seuil} \left(\sum_{i=1}^p w_i x_i + b \right)$$

et la fonction seuil elle-même est définie par :

$$h_w(x) = \begin{cases} 1 & \text{si } w_1 x_1 + w_2 x_2 + b \geq 0 \\ 0 & \text{sinon} \end{cases}$$

Cela signifie que le perceptron prend une décision binaire. Si la somme pondérée dépasse un certain seuil, la sortie vaut 1 ; sinon, elle vaut 0. Ce mécanisme est simple à comprendre : on peut le voir comme une règle de décision élémentaire [50].

L'apprentissage du perceptron consiste à corriger les poids lorsqu'une observation est mal classée.

$$w_i \leftarrow w_i + \alpha (y_t - h_w(X_t)) x_{t,i}$$

où α est le taux d'apprentissage. Ce paramètre contrôle l'amplitude des corrections. Si α est trop petit, l'apprentissage devient lent ; s'il est trop grand, l'algorithme peut devenir instable ou diverger [50].

Le perceptron présente cependant une limite importante : il fonctionne bien lorsque les données sont linéairement séparables, c'est-à-dire lorsqu'une droite ou un hyperplan suffit à séparer les classes. Le support rappelle que, dans ce cas, le perceptron converge vers une solution à erreur nulle. En revanche, pour des problèmes plus complexes, il devient insuffisant, d'où la nécessité de passer au perceptron multicouche [23].

4.2.6 Fonctions d'activation et modélisation des relations non linéaires

La fonction d'activation détermine la sortie du neurone à partir de la somme pondérée reçue. Les supports présentent plusieurs types de fonctions : la fonction seuil, la fonction linéaire, la fonction linéaire saturée, la sigmoïde, la tangente hyperbolique, entre autres.

La fonction seuil est la plus simple. Elle produit une sortie binaire, 0 ou 1. Elle est adaptée à des tâches de décision simples, mais elle présente un inconvénient majeur : elle n'est pas dérivable en certains points, en particulier au niveau du seuil. La régression logistique précise que la descente de gradient s'applique aux fonctions dérivables, ce qui rend l'entraînement du perceptron à seuil parfois instable.

Pour cette raison, dans les réseaux multicouches, on utilise souvent la fonction sigmoïde [50] :

$$h(x) = \frac{1}{1 + \exp(-x)}$$

Cette fonction transforme toute valeur réelle en une sortie comprise entre 0 et 1. Elle possède deux avantages essentiels. D'abord, elle introduit une non-linéarité, ce qui permet au réseau de modéliser des relations complexes. Ensuite, elle est dérivable partout, ce qui la rend compatible avec l'apprentissage par descente de gradient. Les supports montrent également que sa dérivée peut s'écrire sous une forme simple [50] :

$$h'(x) = h(x)(1 - h(x))$$

Cette propriété est très utile dans les calculs de rétropropagation.

En résumé, la fonction d'activation est ce qui donne au réseau sa capacité à apprendre autre chose qu'une simple relation linéaire. Sans elle, même un réseau comportant plusieurs couches se comporterait comme un modèle linéaire global.

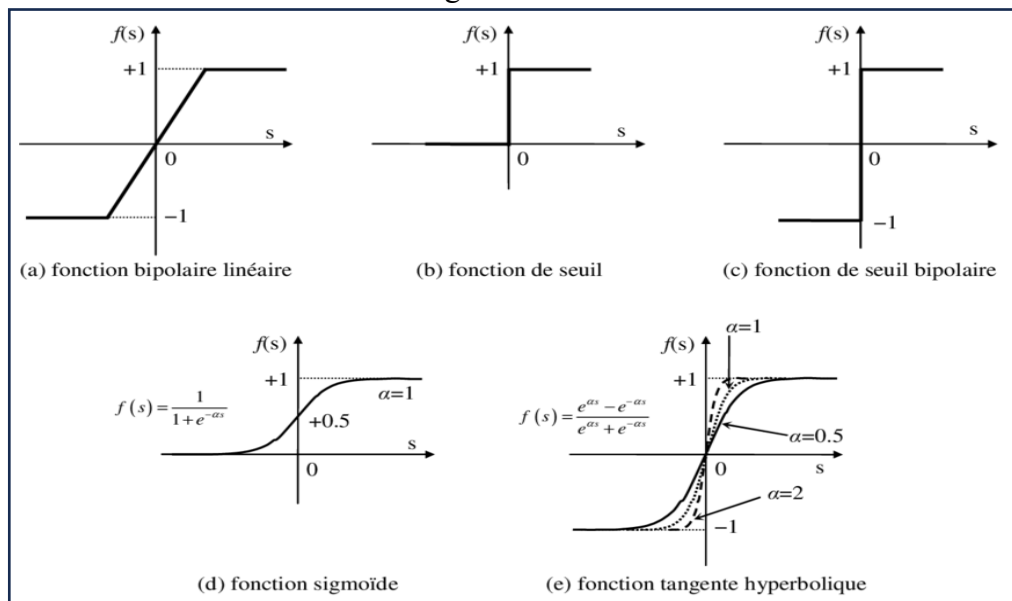


Figure 2.5 : Fonctions d'Activation d'un neurone artificiel

4.2.7 Optimisation de l'apprentissage par la méthode de descente de gradient

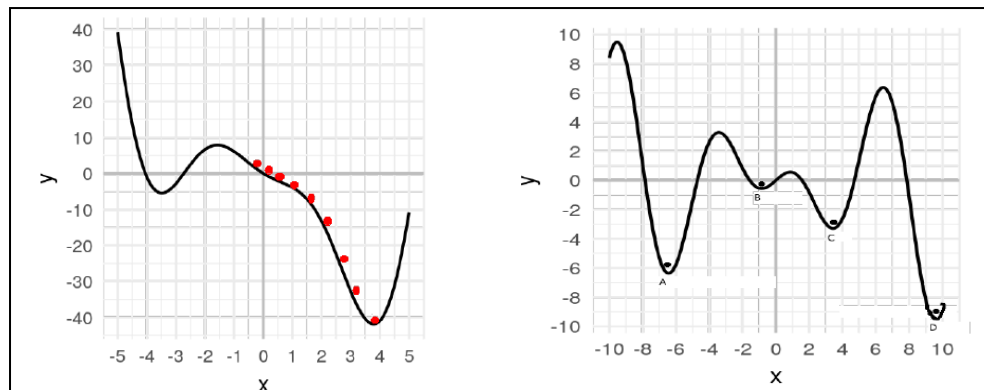


Figure 2.6 : Descente de gradient et problème des minima locaux

Pour qu'un réseau apprenne, il faut disposer d'un critère permettant de mesurer la qualité de ses prédictions. Ce critère est généralement une fonction d'erreur ou fonction de coût. Le but de l'apprentissage est alors de trouver les poids qui minimisent cette erreur. C'est précisément le rôle de la descente de gradient[23].

Avant d'introduire cette méthode, les supports rappellent la notion de dérivée. Pour une fonction à une variable, la dérivée indique si la fonction augmente ou diminue. Pour une fonction à plusieurs variables, on utilise les dérivées partielles. Le gradient est alors le vecteur formé par toutes ces dérivées partielles :

$$\nabla f(x, y, z) = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right]$$

Intuitivement, le gradient indique la direction dans laquelle la fonction augmente le plus rapidement. Donc, pour la minimiser, il faut aller dans la direction opposée. La règle générale de mise à jour par descente de gradient s'écrit :

$$x_{i+1} = x_i - \alpha f'(x_i)$$

Dans le cas des réseaux de neurones, cette idée est appliquée aux poids. Le support indique que pour le perceptron, on ajuste les poids selon :

$$w_i = w_i - \alpha \frac{\partial E}{\partial w_i}$$

Le taux d'apprentissage α détermine la taille du pas de correction. Un taux trop faible ralentit la convergence, tandis qu'un taux trop élevé risque de provoquer des oscillations ou d'empêcher l'atteinte d'un minimum. Le support mentionne également le problème des minima locaux :

l'algorithme peut parfois s'arrêter dans un minimum qui n'est pas le meilleur possible. Une solution proposée consiste à relancer l'apprentissage à partir de plusieurs points initiaux[50].

4.2.8 Mécanisme d'apprentissage par rétropropagation du gradient

La rétropropagation du gradient est la méthode la plus classique pour entraîner un réseau multicouche. Elle se déroule en deux grandes étapes. La première est la propagation avant[23].

À ce stade, les données d'entrée traversent le réseau couche par couche jusqu'à produire une sortie observée $s(t)$. Cette sortie est ensuite comparée à la sortie souhaitée $y(t)$. L'erreur pour le neurone de sortie j est donnée par :

$$e_j(t) = y_j(t) - s_j(t)$$

L'erreur globale du réseau peut alors être exprimée sous la forme :

$$E(t) = \frac{1}{2} \sum_{j=1}^q e_j^2(t)$$

Cette expression correspond à une erreur quadratique globale. La seconde étape est la rétropropagation. L'idée est de répartir l'erreur en sens inverse dans le réseau afin de déterminer comment modifier chaque poids. Pour cela, on utilise la dérivation en chaîne. Le support montre que, pour la couche de sortie, la variation du poids s'écrit :

$$\Delta w_{ij}(t) = -\alpha \frac{\partial E(t)}{\partial w_{ij}(t)}$$

Après développement, on obtient :

Avec
$$\Delta w_{ij}(t) = \alpha \delta_j(t) s_i(t)$$

$$\delta_j(t) = e_j(t) s_j(t) (1 - s_j(t))$$

Le terme $\delta_j(t)$ est appelé gradient local. Il mesure la contribution du neurone j à l'erreur globale [50].

Pour les couches cachées, le calcul est un peu plus complexe, car il n'existe pas directement de sortie désirée pour ces neurones. Le support donne alors la relation suivante :

$$\delta_j(t) = s_j(t)(1 - s_j(t)) \sum_{k=1}^q \delta_k(t) w_{jk}(t)$$

Cette formule signifie que l'erreur d'un neurone caché dépend des erreurs des neurones de la couche suivante. Ainsi, l'erreur est propagée progressivement depuis la sortie jusqu'aux premières couches du réseau. C'est ce mécanisme qui permet à un réseau profond d'apprendre [50].

Enfin, l'algorithme global d'apprentissage consiste à initialiser les poids aléatoirement, présenter les données d'apprentissage, calculer les sorties, corriger les poids, puis répéter ce processus jusqu'à atteindre un critère d'arrêt, par exemple un nombre maximal d'itérations ou une erreur quadratique moyenne suffisamment faible.

4.2.9 Généralisation, validation et performance des modèles neuronaux

Un réseau de neurones ne doit pas seulement apprendre les exemples qu'on lui présente ; il doit également être capable de bien répondre à de nouvelles données. Cette propriété est appelée généralisation. Le support sur le perceptron précise qu'une erreur nulle sur la base d'apprentissage ne garantit pas forcément la qualité du modèle sur des données non vues. Le système peut avoir simplement mémorisé les exemples.

Pour éviter ce problème, il est nécessaire de séparer les données en plusieurs ensembles. On distingue généralement un ensemble d'apprentissage, utilisé pour ajuster les poids, et un ensemble de test, utilisé pour évaluer les performances du modèle sur des données nouvelles. La validation croisée comme stratégie pour mieux apprécier la capacité de généralisation.

Ainsi, l'évaluation de la performance d'un réseau ne doit pas se limiter à l'erreur d'apprentissage. Il faut également vérifier sa robustesse, sa stabilité et sa capacité à produire de bonnes prédictions sur des observations inédites. C'est cette étape qui permet de juger si le réseau a réellement appris le phénomène étudié.

4.2.10 Apports des réseaux de neurones artificiels dans les tâches de prédiction

Les réseaux de neurones artificiels présentent plusieurs avantages pour les tâches de prédiction. D'abord, ils peuvent traiter un grand nombre de variables d'entrée et apprendre automatiquement leurs relations avec la sortie. Ensuite, grâce aux couches cachées et aux fonctions d'activation non linéaires, ils peuvent représenter des interactions complexes que des modèles linéaires simples ne capturent pas. Le perceptron multicouche permet notamment de construire des frontières de décision non linéaires et d'approximer des fonctions très variées.

De plus, l'approche connexionniste de l'IA repose sur l'idée de laisser le système trouver par lui-même une solution satisfaisante à partir des données, au lieu de lui imposer toutes les règles de décision. Cela rend les réseaux de neurones particulièrement intéressants pour la prédiction dans des contextes où les relations entre variables sont nombreuses, évolutives ou difficilement modélisables de manière explicite.

Leur principal intérêt réside donc dans leur capacité d'apprentissage, leur souplesse de modélisation et leur aptitude à la généralisation, sous réserve d'un bon choix des données, de l'architecture et des paramètres d'apprentissage.

4.3 XGBOOST

Dans les environnements de production, les données sont le plus souvent tabulaires, hétérogènes et issues de systèmes ERP, MES ou de relevés opérationnels. Face à ces caractéristiques, XGBoost s'est imposé comme l'un des algorithmes les plus fiables pour prédire des indicateurs tels que les quantités produites, les rendements ou les délais. Cette section vise donc à expliquer, de manière progressive et rigoureuse, son fonctionnement, ses mécanismes d'optimisation, et les propriétés qui justifient son utilisation dans un contexte de prédiction de la performance de production.

4.3.1 Des modèles simples aux modèles ensemblistes : le principe de combinaison d'apprenants faibles

4.3.1.1 Limites d'un modèle unique et motivation des méthodes ensemblistes

Dans la pratique de l'apprentissage automatique appliqué à des contextes opérationnels, planification de la production, gestion des flux logistiques, pilotage des performances nous cherchons à construire des modèles capables de produire des prédictions fiables et stables sur des données nouvelles. Un modèle unique, aussi sophistiqué soit-il, se heurte à une difficulté fondamentale : le compromis biais-variance.

Le biais mesure l'écart systématique entre les prédictions d'un modèle et les valeurs réelles. Un modèle trop simple par exemple une régression linéaire appliquée à un phénomène non linéaire aura un biais élevé : il ne capture pas suffisamment la complexité du problème. À l'inverse, la variance désigne la sensibilité du modèle aux fluctuations des données d'entraînement. Un modèle trop complexe un arbre de décision très profond, par exemple aura une variance élevée : il apprend par cœur les exemples d'entraînement, mais se révèle incapable de généraliser sur de nouvelles observations.

Ce dilemme est bien connu dans le domaine de la prédiction industrielle. Un modèle qui prédit parfaitement les quantités produites sur les données historiques, mais échoue à anticiper les variations sur une nouvelle période de production, n'a aucune valeur opérationnelle. Il faut donc disposer d'une approche qui maintient simultanément un biais faible et une variance contrôlée.

C'est précisément la motivation des méthodes ensemblistes. Plutôt que de s'appuyer sur un seul modèle, elles combinent les prédictions de plusieurs modèles, appelés apprenants faibles ou weak learners. L'idée centrale est intuitive : plusieurs estimateurs imparfaits mais diversifiés peuvent, lorsqu'ils sont agrégés de façon intelligente, produire une prédiction globale bien meilleure que celle de l'un d'eux pris isolément. Cette logique rappelle celle d'une équipe de planification : une décision collective, issue de plusieurs points de vue complémentaires, est souvent plus robuste qu'un jugement individuel.

4.3.1.2 Le Bagging : réduire la variance par agrégation parallèle

La première grande famille de méthodes ensemblistes est le Bagging, acronyme de Bootstrap AGGREGatING, introduit par Breiman en 1996 [30]. Son principe repose sur deux mécanismes : le rééchantillonnage bootstrap et l'agrégation des prédictions.

Le rééchantillonnage bootstrap consiste à créer, à partir du jeu de données original, plusieurs sous-ensembles d'entraînement obtenus par tirage avec remise. Chaque sous-ensemble contient autant d'observations que le jeu original, mais certaines observations apparaissent plusieurs fois, tandis que d'autres sont absentes. Sur chacun de ces sous-ensembles, un apprenant faible est entraîné de façon indépendante. Enfin, les prédictions de tous ces apprenants sont agrégées par vote majoritaire pour la classification, par moyenne pour la régression.

L'intérêt du Bagging réside dans sa capacité à réduire la variance sans augmenter le biais. En entraînant des modèles sur des échantillons différents, nous obtenons des estimateurs diversifiés dont les erreurs individuelles tendent à se compenser lors de l'agrégation. La Forêt Aléatoire (Random Forest), développée par Breiman en 2001, est l'application la plus connue de ce principe [30].

4.3.1.3 Le Boosting : corriger l'erreur par apprentissage séquentiel

Le Boosting constitue la seconde grande famille de méthodes ensemblistes, et c'est celle dont XGBoost est issu. Contrairement au Bagging, les apprenants ne sont pas construits en parallèle de façon indépendante : ils le sont de manière séquentielle, chaque nouvel apprenant cherchant spécifiquement à corriger les erreurs commises par les précédents [32].

L'intuition fondamentale du Boosting est la suivante : lorsqu'un modèle se trompe sur certaines observations, ces observations méritent d'être examinées plus attentivement lors de l'étape suivante. Le nouvel apprenant se concentre donc davantage sur les cas difficiles, ceux que le modèle courant prédit mal. Au fil des itérations, la combinaison des apprenants successifs permet de couvrir progressivement l'ensemble des difficultés du problème.

Sur le plan formel, un modèle ensembliste par boosting construit une fonction de prédiction $F(x)$ comme une somme de K apprenants faibles f_k :

$$F(x) = \sum_{k=1}^K f_k(x)$$

Chaque apprenant f_k est ajouté pour corriger les résidus laissés par la somme des apprenants précédents.

C'est cette logique additive et séquentielle qui distingue fondamentalement le boosting du bagging [32].

4.3.1.4 Comparaison conceptuelle Bagging / Boosting

Dans le contexte logistique et industriel, le Boosting s'est révélé particulièrement efficace sur des jeux de données où les relations entre variables sont hétérogènes et non linéaires comme c'est souvent le cas lorsque nous cherchons à prédire les rendements d'une ligne de production à partir de variables opérationnelles variées.

Ces deux familles se distinguent sur plusieurs dimensions importantes, que nous pouvons résumer ainsi :

Critère	Bagging	Boosting
Ordre de construction	Parallèle	Séquentiel
Dépendance entre apprenants	Indépendance totale	Dépendance forte (chaque apprenant corrige les erreurs du précédent)
Principal effet	Réduit la variance	Réduit le biais tout en maîtrisant la variance
Sensibilité au surapprentissage	Naturellement robuste	Risque de surapprentissage sans régularisation
Parallélisation	Facile et naturelle	Séquentielle par nature, avec optimisations dans XGBoost

Tableau 2.1 : Critères de comparaison entre le Bagging et le Boosting

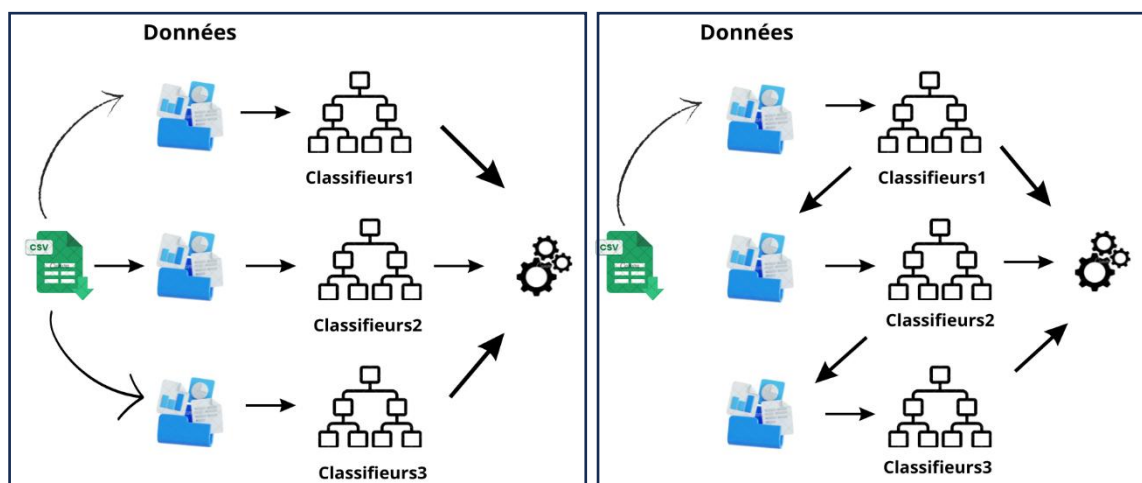


Figure 2.7 : Structure de la méthode Bagging Figure 2.8 : Structure de la méthode Boosting

4.3.2 L'arbre de décision comme apprenant de base

4.3.2.1 Structure et principe de l'arbre de régression

L'apprenant faible utilisé dans XGBoost est l'arbre de décision, et plus précisément l'arbre de régression dans le cas de problèmes à cible continue. Comprendre cet élément de base est indispensable pour saisir la mécanique interne de XGBoost.

Un arbre de décision est une structure hiérarchique qui partitionne l'espace des variables d'entrée en régions successives au moyen de règles binaires. Chaque nœud interne de l'arbre correspond à une condition de type (variable $X_j \leq \text{seuil } s$?), qui oriente chaque observation vers

la branche gauche ou droite. Les feuilles de l'arbre, situées aux extrémités, correspondent aux régions finales dans lesquelles une valeur de prédiction est attribuée.

Formellement, un arbre de régression partitionne l'espace des variables X en T régions $R_1, R_2, \dots, R_T$, et prédit pour chaque région une valeur constante [20] :

$$f(x) = \sum_{t=1}^T w_t \cdot 1(x \in R_t)$$

Où w_t est la valeur prédite dans la région R_t . Cette valeur est généralement la moyenne des valeurs cibles des observations tombant dans cette région.

4.3.2.2 Critères de découpage : réduction de l'impureté et gain d'information

La construction d'un arbre de régression repose sur le choix de la variable et du seuil qui produisent le meilleur découpage à chaque nœud. Ce choix est guidé par un critère de gain : nous cherchons la partition qui réduit le plus possible l'hétérogénéité des valeurs cibles au sein de chaque groupe.

Pour la régression, le critère le plus courant est la réduction de la somme des erreurs quadratiques. Étant donné un ensemble d'observations S à un nœud, la partition en sous-ensembles gauche SG et droit SD est retenue si elle maximise :

$$\text{Gain} = \sum_{i \in S} (y_i - \bar{y}_S)^2 - [\sum_{i \in SG} (y_i - \bar{y}_{SG})^2 + \sum_{i \in SD} (y_i - \bar{y}_{SD})^2]$$

Ce gain mesure la réduction d'impureté apportée par le découpage. Plus ce gain est élevé, plus la partition sépare efficacement des groupes homogènes.

Où

Paramètre	Signification
SG	Sous-ensemble gauche après la coupure
SD	Sous-ensemble droit après la coupure
$\bar{y}_{SG}$	Moyenne des y_i dans le groupe gauche
$\bar{y}_{SD}$	Moyenne des y_i dans le groupe droit
$\sum_{i \in SG} (y_i - \bar{y}_{SG})^2$	Variance résiduelle gauche après découpage
$\sum_{i \in SD} (y_i - \bar{y}_{SD})^2$	Variance résiduelle droite après découpage

Tableau 2.2 : Signification de paramètres de gain

4.3.2.3 Le compromis biais-variance dans un arbre

Un arbre très peu profond ; un seul niveau de découpage, par exemple produit des prédictions très simplistes. Il ne capte que les relations les plus grossières entre les variables et la cible, ce qui génère un biais important. En contrepartie, sa variance est faible : ses prédictions ne fluctuent pas beaucoup d'un échantillon d'entraînement à l'autre.

À l'opposé, un arbre très profond, dont chaque feuille ne contient qu'une seule observation, atteint une erreur nulle sur les données d'entraînement. Mais cette précision est illusoire : l'arbre

a mémorisé les données plutôt qu'appris la relation sous-jacente. Sa variance est alors très élevée, et ses prédictions sur de nouvelles données sont médiocres.

4.3.2.4 Pourquoi l'arbre peu profond est l'apprenant idéal pour le boosting

Dans le cadre du Boosting, et en particulier dans XGBoost, les arbres utilisés sont intentionnellement peu profonds souvent limités à 4 à 8 niveaux, parfois moins. Ce choix est délibéré et théoriquement motivé.

Un arbre peu profond est un apprenant faible au sens strict : il prédit mieux que le hasard, mais imparfaitement. C'est exactement ce que le Boosting recherche. En enchaînant de nombreux arbres peu profonds, chacun corrigeant les erreurs du précédent, nous construisons progressivement un modèle de plus en plus précis, tout en gardant le contrôle sur la complexité globale. Si chaque arbreau contraire était très profond et déjà très précis, le mécanisme de correction séquentielle perdrait son sens, et le risque de surapprentissage augmenterait considérablement.

4.3.3 Du Gradient Boosting classique à XGBoost

4.3.3.1 Principe général du Gradient Boosting Machine (GBM)

Le Gradient Boosting Machine, formalisé par Friedman en 2001, est le fondement théorique sur lequel repose XGBoost. Son idée centrale est d'unifier le mécanisme de boosting avec le cadre de la descente de gradient. Dans le Gradient Boosting, la prédiction est construite additivement [33]. À l'itération m , la prédiction courante $F_m(x)$ est mise à jour en ajoutant un nouvel arbre f_m :

$$F_m(x) = F_{m-1}(x) + \eta \cdot f_m(x)$$

Où η est le taux d'apprentissage, un paramètre qui contrôle l'amplitude de chaque correction. La question clé est : quel arbre f_m doit-on ajouter ? La réponse du Gradient Boosting est : l'arbre qui approxime le mieux les pseudo-résidus, c'est-à-dire le gradient négatif de la fonction de perte par rapport aux prédictions courantes.

En choisissant une perte quadratique $L(y, F) = (y - F)^2 / 2$, le gradient négatif correspond exactement aux résidus classiques $y - F_{m-1}(x)$. Mais la beauté du Gradient Boosting est sa généralité : en changeant la fonction de perte, nous pouvons l'adapter à n'importe quel problème de régression ou de classification, sans modifier la logique algorithmique.

4.3.3.2 Descendre le gradient dans l'espace des fonctions

La différence conceptuelle entre le Gradient Boosting et une descente de gradient classique est subtile mais fondamentale. Dans une descente de gradient ordinaire, nous optimisons un vecteur de paramètres dans un espace numérique. Dans le Gradient Boosting, nous optimisons une fonction dans un espace de fonctions : à chaque étape, nous cherchons la fonction ici un arbre de décision qui représente la meilleure direction de descente dans cet espace fonctionnel.

Cette reformulation nous permet de "descendre le gradient" non pas en modifiant des paramètres réels, mais en ajoutant des structures qui pointent vers la direction de réduction de l'erreur. C'est cette idée qui confère au Gradient Boosting sa flexibilité et sa généralité.

4.3.3.3 Formalisation de l'objectif XGBoost

Introduit par Chen et Guestrin en 2016, XGBoost se démarque par une reformulation soignée de la quantité à optimiser, dans laquelle un terme de régularisation est explicitement incorporé [34]. La fonction objective que nous minimisons prend la forme suivante :

$$\text{Obj} = \sum_{i=1}^n L(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k)$$

Le premier terme quantifie l'écart entre prédictions et valeurs observées sur les n exemples d'apprentissage : il s'agit de la fonction de perte. Le second, $\Omega(f_k)$, sanctionne la complexité de chaque arbre :

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \|w\|^2$$

Ici, T désigne le nombre de feuilles de l'arbre, w le vecteur regroupant les poids de ces feuilles, γ une pénalité appliquée au nombre de feuilles, et λ le coefficient de régularisation L2 portant sur les poids.

C'est précisément cette construction qui sépare XGBoost du GBM traditionnel. En inscrivant la régularisation au cœur même de l'objectif, nous contraignons le modèle à poursuivre deux buts conjoints : réduire l'erreur de prédiction, mais aussi conserver des arbres de structure simple. Il en résulte une maîtrise accrue du surapprentissage un atout d'autant plus appréciable dans les contextes de prédiction industrielle, où les jeux de données sont fréquemment réduits ou entachés de bruit.

4.3.3.4 Approximation du second ordre (Taylor) : le saut qualitatif par rapport au GBM

Le ressort technique propre à XGBoost tient à l'emploi d'un développement de Taylor au second ordre pour approcher la fonction de perte. Là où le GBM classique se limite au gradient de premier ordre la dérivée première de la perte, XGBoost mobilise en plus la courbure locale par l'intermédiaire du Hessien, autrement dit la dérivée seconde.

En désignant par g_i le gradient et par h_i le Hessien de la perte associés à l'exemple i , nous posons :

$$g_i = \frac{\partial L(y_i, \hat{y}_{i(m-1)})}{\partial \hat{y}_{i(m-1)}}$$

$$h_i = \frac{\partial^2 L(y_i, \hat{y}_{i(m-1)})}{\partial (\hat{y}_{i(m-1)})^2}$$

L'objectif réécrit à l'étape m s'exprime alors sous une forme simplifiée :

$$\text{Obj}(m) \approx \sum_{i=1}^n [g_i \cdot f(x_i) + \left(\frac{1}{2}\right) h_i \cdot f(x_i)^2] + \Omega(f)$$

Le recours au second ordre nous procure deux bénéfices déterminants. Premièrement, l'optimisation gagne en finesse : connaître la courbure, et non plus seulement la pente, nous renseigne sur la forme réelle de la perte et débouche sur des corrections plus justes à chaque itération. Deuxièmement, la formulation devient générique : toute l'information utile se trouve condensée dans g_i et h_i , indépendamment de la perte retenue. Adapter XGBoost à des problèmes très divers revient dès lors à modifier la seule définition de L .

4.3.3.5 Score de gain et choix du découpage optimal

À partir de l'expression précédente, le poids idéal de chaque feuille j se déduit par voie analytique :

$$w_j^* = \frac{-\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$$

Où I_j rassemble les exemples acheminés vers la feuille j . Ce poids se calcule directement à partir des gradients et des Hessiens, sans qu'aucune optimisation numérique additionnelle ne soit requise.

De la même manière, le critère permettant de juger un découpage candidat s'obtient analytiquement :

$$\text{Gain} = (1/2) \cdot \left[\frac{GL^2}{(HL+\lambda)} + \frac{GR^2}{(HR+\lambda)} - \frac{(GL+GR)^2}{(HL+HR+\lambda)} \right] - \gamma$$

Cette grandeur traduit la diminution de l'objectif qu'apporte la partition, diminuée d'une pénalité γ destinée à brider la profondeur de l'arbre. Un gain négatif conduit au rejet du découpage : on dispose ainsi d'un élagage intrinsèque, intégré au fil de la construction de l'arbre.

4.3.4 Architecture interne et mécanismes clés de XGBoost

4.3.4.1 Anatomie d'un arbre : feuilles, poids et score de structure

Au sein de XGBoost, un arbre se décrit par deux objets complémentaires : une structure q qui, à toute observation x , associe la feuille où elle aboutit, et un vecteur de poids w renfermant les valeurs prédites par chacune des feuilles. La sortie de l'arbre pour une observation x se réduit alors à :

$$f(x) = w \times q(x)$$

Pour apprécier la qualité d'une structure q , nous lui attribuons un score de structure défini par :

$$\text{Score}(q) = -\frac{1}{2} \times \sum_{j=1}^T \left[\frac{G_j^2}{(H_j+\lambda)} \right] + \gamma T$$

Plus ce score est bas (donc plus il est négatif), meilleure est la structure. La construction procède de façon gloutonne : à chaque nœud, l'algorithme parcourt les partitions envisageables et retient celle qui maximise le gain.

4.3.4.2 Régularisations L1 et L2

Deux formes de régularisation cohabitent directement dans l'objectif de XGBoost :

- La régularisation L2 (ridge), pilotée par λ , frappe la norme au carré des poids des feuilles. Elle interdit aux poids de dériver vers des valeurs excessives et atténue la réactivité du modèle face aux variations des données d'apprentissage.
- La régularisation L1 (lasso), pilotée par α , frappe la norme absolue des poids. Elle pousse vers zéro les poids des feuilles peu porteuses d'information et opère de la sorte une sélection implicite de variables.

Dans un cadre industriel où certaines variables se révèlent peu discriminantes ou redondantes, ces deux leviers nous aident à façonner des modèles plus sobres et mieux aptes à généraliser.

4.3.4.3 Le taux d'apprentissage (η)

Le coefficient de rétrécissement η tempère l'apport de chaque nouvel arbre au modèle cumulé. La mise à jour après l'étape m s'écrit :

$$\hat{f}_m(x) = \hat{f}_{m-1}(x) + \eta \cdot f_m(x)$$

Une valeur modeste de η couramment comprise entre 0,01 et 0,3 implique que chaque arbre ne corrige l'erreur que par petites touches. Cela impose d'empiler davantage d'arbres pour viser une précision donnée, mais le modèle obtenu se révèle en général plus stable et moins enclin au surapprentissage [20].

Une image logistique éclaire le procédé : pour réguler un flux de production, mieux vaut multiplier les petits ajustements successifs qu'asséner une correction unique et massive, susceptible de déstabiliser tout l'atelier.

4.3.4.4 Sous-échantillonnage des lignes et des colonnes :

La robustesse par la diversité XGBoost dispose de deux mécanismes de sous-échantillonnage qui injectent de la diversité d'un arbre à l'autre :

- Sur les observations (subsample) : chaque itération ne s'appuie que sur une portion tirée au hasard des exemples d'apprentissage, ce qui freine le surapprentissage tout en allégeant le temps de calcul.
- Sur les variables (colsample_bytree, colsample_bylevel) : à l'échelle de l'arbre ou du nœud, seul un sous-ensemble aléatoire de variables est mis en concurrence pour le découpage. Hérité de la Forêt Aléatoire, ce procédé contraint les arbres successifs à

explorer des combinaisons distinctes, renforçant d'autant la diversité et la solidité de l'ensemble.

4.3.4.5 Prise en charge native des valeurs manquantes

Parmi les avantages concrets de XGBoost en milieu industriel figure son aptitude à traiter directement les valeurs manquantes. Au lieu d'imputer ces valeurs au préalable, l'algorithme apprend de lui-même vers quelle branche gauche ou droite diriger les observations dont une variable fait défaut.

Concrètement, lorsqu'une observation ne renseigne pas la variable servant au découpage, les deux orientations sont testées et celle qui produit le meilleur gain l'emporte ; cette direction par défaut est ensuite consignée dans la structure de l'arbre.

Sur le terrain de la production, où les relevés manuels font parfois défaut, où des capteurs tombent en panne et où les changements de gamme perturbent la collecte, cette propriété se traduit par un gain opérationnel tangible.

4.3.5 Optimisation algorithmique

4.3.5.1 Weighted Quantile Sketch

Bâtir un arbre suppose, à chaque nœud, de soupeser l'ensemble des seuils candidats pour chaque variable afin d'isoler le meilleur découpage. Dès lors que le jeu de données réunit des millions d'observations et des centaines de variables, cette énumération exhaustive devient impraticable du point de vue du temps de calcul.

XGBoost contourne l'obstacle grâce à l'algorithme Weighted Quantile Sketch. Plutôt que de balayer la totalité des seuils, nous générons un ensemble de seuils répartis de façon approximativement uniforme sur l'étendue de chaque variable, en pondérant chaque observation par son Hessien hi. La recherche se concentre ensuite sur cet échantillon réduit de candidats, sans dégradation notable de la précision.

4.3.5.2 Column Block

L'apport architectural le plus décisif pour la montée en charge réside dans le stockage en colonnes triées, baptisé Column Block. Les données d'apprentissage sont conservées sous forme de colonnes compressées, chacune ordonnée par valeurs croissantes. Cette disposition autorise, lors de la quête du meilleur découpage, un parcours linéaire et favorable au cache, en s'affranchissant de tout retri à chaque nœud.

Construite une fois pour toutes avant l'entraînement, cette structure est ensuite réutilisée à chaque itération. Elle ouvre par ailleurs la voie à une parallélisation aisée de la recherche de découpage sur plusieurs variables à la fois.

4.3.5.3 Accès optimisés au cache et parallélisme au niveau des feuilles

XGBoost a été pensé pour tirer parti de la hiérarchie mémoire des processeurs récents. Lors de l'examen des découpages, les gradients et Hessiens sont consultés selon un ordonnancement qui maximise les succès de cache, limitant d'autant les accès à la mémoire centrale.

Notons que la parallélisation ne s'applique pas aux arbres eux-mêmes leur enchaînement étant par nature séquentiel, mais bien à la recherche du meilleur découpage à l'intérieur d'un arbre donné. Les différentes variables candidates sont évaluées simultanément sur plusieurs threads, ce qui réduit sensiblement le temps de construction.

4.3.5.4 Calcul distribué et mode out-of-core

Lorsque le volume excède la mémoire vive disponible, XGBoost propose un fonctionnement out-of-core : les données résident sur disque et sont chargées par blocs en cours d'entraînement, la compression et le pré-chargement en arrière-plan réduisant les temps d'attente. Le recours au calcul distribué sur grappes de machines, via des frameworks tels qu'Apache Spark ou Dask, reste également envisageable, ce qui rend l'outil exploitable jusqu'à des volumétries industrielles considérables.

4.3.6 Hyperparamètres fondamentaux et stratégie de réglage

3.3.6.1 Une cartographie en quatre familles

Les hyperparamètres que nous avons à ajuster se laissent regrouper en quatre familles fonctionnelles :

Structure de l'arbre

- `max_depth` : profondeur maximale autorisée. Plus elle est grande, plus la capacité de modélisation s'élève, mais le risque de surapprentissage croît d'autant. Plage usuelle : 3 à 8.
- `min_child_weight` : poids hessien minimal exigé dans une feuille. En interdisant les feuilles trop peu peuplées, il joue un rôle de régulateur de profondeur.
- `Gamma` : réduction de perte minimale requise pour valider un découpage. Une valeur positive rend l'arbre plus prudent.

Apprentissage

- `n_estimators` : nombre total d'arbres édifiés.
- `learning_rate` (`eta`) : intensité du rétrécissement appliqué à chaque arbre.

Régularisation

- `lambda` : régularisation L2 sur les poids des feuilles.
- `alpha` : régularisation L1 sur les poids des feuilles.

Sous-échantillonnage

- subsample : proportion d'observations mobilisées par arbre.
- colsample_bytree, colsample_bylevel : proportion de variables retenues par arbre ou par niveau.

4.3.6.2 Des hyperparamètres en interaction

Les hyperparamètres ne produisent pas leurs effets isolément : ils se conjuguent et se répondent. Ainsi, l'association d'un learning_rate faible et d'un n_estimators élevé surpasse le plus souvent le couple inverse taux fort, peu d'arbres, pour peu que la profondeur reste sous contrôle. Dans le même esprit, abaisser max_depth et relever min_child_weight concourent l'un et l'autre à brider la complexité, mais en agissant sur des facettes différentes de la structure.

En pratique industrielle, nous recommandons de fixer d'abord un learning_rate modérément bas (0,05 à 0,1), de déterminer le nombre d'arbres optimal par arrêt précoce, puis de peaufiner la structure et la régularisation.

4.3.6.3 L'arrêt précoce comme régulateur automatique

L'arrêt précoce consiste à suivre la performance du modèle sur un jeu de validation au fil des itérations et à stopper l'entraînement dès qu'aucune amélioration ne survient sur un nombre prédéfini d'étapes consécutives (early_stopping_rounds).

Son intérêt est double : il fixe automatiquement le nombre d'arbres pertinent et nous préserve aussi bien du sous-apprentissage (trop peu d'arbres) que du surapprentissage (trop d'arbres). Le suivi s'opère sur une métrique calculée sur un ensemble de validation tenu à l'écart de l'apprentissage.

4.3.6.4 Explorer l'espace des hyperparamètres

Identifier les meilleurs hyperparamètres constitue une étape charnière du développement. Trois démarches principales s'offrent à nous :

- Recherche par grille (Grid Search) : balayage exhaustif d'une grille prédéfinie de combinaisons. La méthode garantit de localiser le meilleur point de la grille, mais son coût explose dès que le nombre de paramètres augmente.
- Recherche aléatoire (Random Search) : tirage au hasard de combinaisons dans l'espace des hyperparamètres. Bergstra et Bengio ont établi qu'à budget de calcul fixé, cette stratégie couvre l'espace plus efficacement qu'une grille.
- Optimisation bayésienne : on modélise de manière probabiliste la performance en fonction des hyperparamètres, puis on s'appuie sur ce modèle pour orienter judicieusement les points suivants à tester. C'est l'approche la plus économe en nombre d'évaluations nécessaires pour aboutir à un réglage satisfaisant.

4.3.7 Évaluation, généralisation et interprétabilité

4.3.7.1 Mesurer la performance en régression : MAE, RMSE, R²

Apprécier un modèle de régression XGBoost passe par des métriques chiffrées, calculées entre prédictions et valeurs réelles :

- Erreur absolue moyenne (MAE) :

$$\text{MAE} = \left(\frac{1}{n}\right) \times \sum |y_i - \hat{y}_i|$$

Elle exprime l'écart moyen dans l'unité de la variable cible par exemple 35 pièces par jour et se montre peu réactive aux erreurs extrêmes [20].

- Racine de l'erreur quadratique moyenne (RMSE) :

$$\text{RMSE} = \sqrt{\left[\left(\frac{1}{n}\right) \times \sum (y_i - \hat{y}_i)^2\right]}$$

En amplifiant les grands écarts, elle s'impose lorsque les fortes erreurs s'avèrent particulièrement coûteuses sur le plan opérationnel [20].

- Coefficient de détermination R² :

$$R^2 = 1 - \left[\frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}\right]$$

Il indique la part de variance de la cible captée par le modèle ; une valeur proche de 1 signale un bon ajustement [20].

4.3.7.2 Validation croisée et découpage temporel : ne pas trahir la chronologie

La validation croisée demeure la méthode de référence pour jauger la capacité de généralisation[49]. Les données sont scindées en K partitions ; le modèle s'entraîne sur K-1 d'entre elles et s'évalue sur la dernière, le procédé étant répété K fois avant moyennage des performances [10].

Toutefois, dès lors que les données de production sont indexées dans le temps, une validation croisée ordinaire qui brasse les observations au hasard devient inadaptée. Elle provoquerait une fuite d'information temporelle : le modèle apprendrait sur des données futures pour prédire le passé, ce qui gonflerait artificiellement les estimations de performance.

Il nous faut donc imposer un découpage temporel strict, dans lequel l'ensemble des données d'apprentissage précède chronologiquement celles de validation. La causalité est ainsi respectée et l'évaluation reproduit fidèlement les conditions réelles d'exploitation.

4.3.7.3 Surapprentissage et sous-apprentissage

Le diagnostic repose sur la confrontation des performances entre apprentissage et validation :

- Sous-apprentissage : l'erreur reste élevée des deux côtés. Le modèle manque de capacité. On y remédie en approfondissant les arbres, en allégeant la régularisation ou en multipliant les arbres.
- Surapprentissage : l'erreur est faible à l'apprentissage mais grimpe en validation. Le modèle est trop riche. La correction consiste à diminuer `max_depth`, à renforcer `lambda` et `alpha`, à abaisser `subsample` et `colsample_bytree`, ou encore à activer l'arrêt précoce.

Afin de mieux appréhender ces comportements, la figure suivante présente une représentation graphique comparative des trois états possibles du modèle surapprentissage, équilibre et sous-apprentissage appliqués aux tâches de classification et de régression.

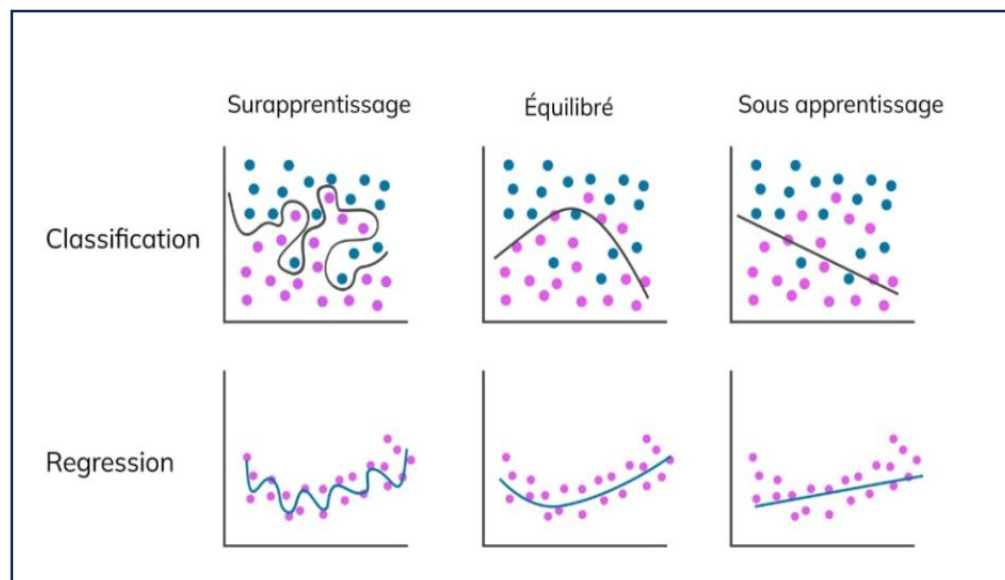


Figure 2.9 : Etats possibles du modèle d'apprentissage

4.3.7.4 Importance des variables : gain, couverture, fréquence

XGBoost livre d'office, pour chaque variable, une mesure d'importance selon trois angles complémentaires :

- Le gain : contribution moyenne à la réduction de la perte imputable aux découpages s'appuyant sur cette variable. C'est l'indicateur le plus parlant quant à l'utilité prédictive effective.
- La couverture : nombre moyen d'observations concernées par les découpages mobilisant cette variable.
- La fréquence : nombre de fois où la variable intervient dans les découpages, tous arbres confondus.

Ces indicateurs nous aident à repérer les variables les plus déterminantes et peuvent éclairer des arbitrages opérationnels : quels indicateurs surveiller en priorité, quelles données collecter avec le plus grand soin.

4.3.7.5 Interprétabilité locale avec SHAP : entrouvrir la boîte noire

Les scores d'importance globaux fournis par XGBoost souffrent d'une limite : leur caractère agrégé les empêche d'expliquer pourquoi le modèle aboutit à une prédiction précise pour une observation donnée. Pour combler ce manque, Lundberg et Lee ont proposé en 2017 les valeurs SHAP (SHapley Additive exPlanations) [40].

Adossées à la théorie des jeux coopératifs de Shapley, ces valeurs décomposent, observation par observation, la prédiction du modèle en contributions propres à chaque variable. La somme de ces contributions, ajoutée à la valeur de référence [40], restitue exactement la prédiction :

$$f(x_i) = E[f(x)] + \sum_{j=1}^p \phi_{ij}$$

Où ϕ_{ij} représente la valeur SHAP de la variable j pour l'observation i .

XGBoost intègre une implémentation native et performante du calcul de ces valeurs. La propriété s'avère précieuse en milieu industriel : elle nous permet à la fois de confronter le modèle au jugement des experts métier et de dialoguer en toute transparence avec les responsables de production sur les facteurs qui pèsent sur les prédictions.

4.3.8 XGBoost dans l'écosystème des algorithmes de prédiction

4.3.8.1 XGBoost face à la Forêt Aléatoire

Forêt Aléatoire et XGBoost comptent parmi les méthodes ensemblistes les plus répandues en pratique ; il vaut la peine d'en cerner les divergences de fond.

La Forêt Aléatoire élève ses arbres en parallèle, chacun sur un sous-échantillon bootstrap, la prédiction finale n'étant que la moyenne des prédictions individuelles. Robuste, peu sensible au réglage et rarement victime de surapprentissage, elle souffre néanmoins d'un plafonnement : les arbres restants indépendants, la précision demeure bornée par la qualité de chacun d'eux [31].

XGBoost, à l'inverse, construit ses arbres en cascade, chacun s'attaquant aux erreurs laissées par le précédent. Cette logique corrective lui vaut, sur des données tabulaires structurées, une précision généralement supérieure. La contrepartie est une sensibilité plus marquée aux hyperparamètres, qui réclame de notre part un réglage plus minutieux pour écarter le surapprentissage.

4.3.8.2 XGBoost, LightGBM et CatBoost : les héritiers du boosting

Depuis la parution de XGBoost en 2016, plusieurs variantes sont venues en prolonger les performances :

- LightGBM s'appuie sur deux idées : le GOSS, qui privilégie la conservation des observations à fort gradient, et l'EFB, qui fusionne des variables mutuellement exclusives afin de réduire la dimensionnalité. Sur des jeux de données larges, LightGBM se montre habituellement plus rapide que XGBoost [35] [36].
- CatBoost tire son originalité d'une gestion native et optimisée des variables catégorielles, qui désamorce la fuite d'information liée à leur encodage. Nous le préconisons tout particulièrement face à de nombreuses variables qualitatives [35] [36].

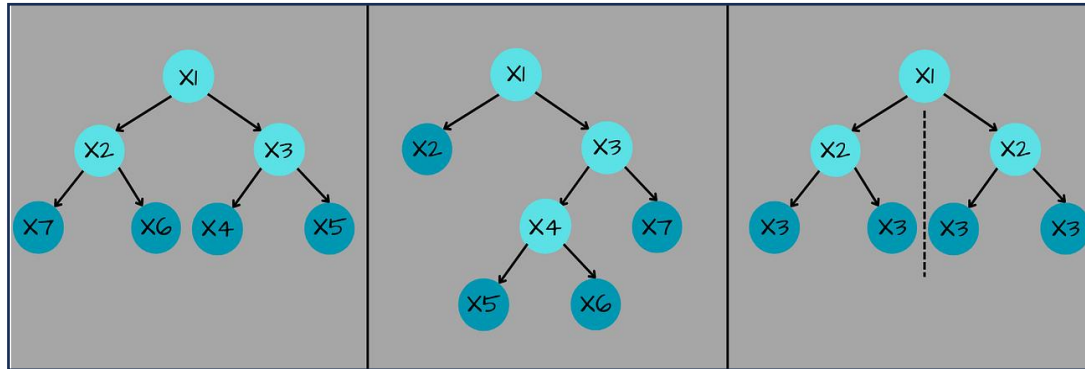


Figure 2.10 : Structure de XGBoost, LightGBM et CatBoost

Dans les faits, le classement de ces trois outils fluctue selon les données. XGBoost conserve néanmoins un statut de valeur sûre, et la maturité de son écosystème lui assure une intégration fluide avec Scikit-learn ainsi qu'un support natif de SHAP [35].

4.3.8.3 XGBoost face aux réseaux de neurones

Le face-à-face entre XGBoost et les réseaux de neurones profonds nourrit l'un des débats les plus vifs de l'apprentissage automatique appliqué. La synthèse de Grinsztajn et al. y apporte un éclairage empirique solide : sur des données tabulaires de taille moyenne à grande, les méthodes à base d'arbres XGBoost au premier chef devancent régulièrement les réseaux profonds [41].

Plusieurs raisons l'expliquent : les données tabulaires mêlent des variables hétérogènes dont les relations ne sont pas invariantes par rotation, et les arbres saisissent spontanément les effets non linéaires comme les interactions entre variables, sans exiger d'immenses volumes de données.

Les réseaux de neurones gardent l'ascendant sur les données non structurées images, texte, signaux ou lorsqu'une architecture spécialisée peut exploiter une structure temporelle ou séquentielle [41] [42].

4.3.8.4 Quand privilégier XGBoost ?

Le choix d'un algorithme doit épouser les caractéristiques du problème et des données. Le tableau ci-dessous récapitule les situations où XGBoost s'impose comme premier réflexe :

Critère	XGBoost recommandé si...
Type de données	Tabulaires, structurées, hétérogènes
Volume de données	Moyen à grand
Valeurs manquantes	Présentes
Besoin d'interprétabilité	Élevé
Précision recherchée	Maximale sur données tabulaires
Ressources de calcul	Limitées
Variables catégorielles	Après encodage approprié

Tableau 2.3 : Critères de choix de XGBoost

Au regard de ces critères, XGBoost se révèle particulièrement adapté lorsque les données sont tabulaires et hétérogènes, disponibles en volume suffisant mais pas nécessairement massif, et lorsque l'on recherche à la fois une précision élevée et une interprétabilité fine, le tout sous des contraintes de calcul raisonnables.

5.Conclusion

Ce chapitre a établi le socle théorique sur lequel repose la suite de ce travail. Après une présentation des grandes familles de l'intelligence artificielle ; apprentissage automatique, apprentissage profond et IA générative, nous avons situé leur rôle croissant dans l'industrie, où elles soutiennent la prédiction, l'optimisation, la maintenance prédictive et l'aide à la décision.

L'étude s'est ensuite resserrée sur les méthodes pertinentes pour notre problématique. Les réseaux de neurones artificiels offrent une capacité avérée à modéliser des relations non linéaires, mais exigent généralement des volumes de données importants et un réglage délicat. Les méthodes à base d'arbres ont été présentées selon une progression cohérente : l'arbre de décision, lisible mais sensible au surapprentissage ; la Forêt Aléatoire, qui en atténue l'instabilité par agrégation parallèle ; et XGBoost, qui exploite un boosting séquentiel régularisé pour affiner la précision au prix d'un réglage plus exigeant.

Cette comparaison ne désigne pas un algorithme universellement supérieur ; elle permet plutôt d'orienter le choix en fonction des caractéristiques du problème et des données disponibles. Chaque méthode présente un équilibre propre entre précision, robustesse, interprétabilité et exigences de réglage, de sorte qu'aucune ne s'impose indépendamment du contexte. Le cadre théorique posé dans ce chapitre fournit ainsi les critères nécessaires pour justifier, sur des bases objectives, l'approche retenue dans la partie appliquée, dont la pertinence sera évaluée empiriquement dans le chapitre suivant.

***CHAPITRE 03 : PREDICTION DE LA QUANTITE
JOURNALIERE PRODUITE PAR LIGNE VIA LE MODELE
XGBOOST***

1. Introduction

Après avoir posé, dans les chapitres précédents, le cadre théorique de notre travail et présenté l'environnement industriel de TAYAL SPA, nous abordons ici le cœur de notre contribution : la conception et le développement d'une application web destinée à prédire la quantité journalière produite par ligne de confection de l'unité 23, à partir de modèles d'apprentissage automatique.

L'idée de ce projet n'est pas née d'un choix abstrait. Elle est le fruit d'une analyse du processus de production que nous avons menée sur le terrain lors d'un premier stage de découverte, puis approfondie, au cours de notre seconde intégration dans le cadre de ce projet de fin d'études, en travaillant directement avec l'équipe de production. C'est en observant le quotidien de l'atelier de confection, en discutant avec les planificateurs et en suivant le cheminement d'un ordre de production, que nous avons identifié un besoin concret : aider à anticiper ce qu'une ligne est réellement capable de produire, et mieux répartir les commandes en conséquence.

Le système que nous proposons se veut un véritable outil d'aide à la décision. Plutôt que de s'appuyer uniquement sur l'expérience et l'estimation manuelle, il exploite l'historique de production pour apprendre les relations entre les caractéristiques d'une commande, les ressources d'une ligne et le rendement effectivement observé. À partir de ces apprentissages, l'application estime la production attendue pour chaque ligne, puis propose une affectation raisonnée des ordres de production qui tient compte des objectifs à atteindre, de la disponibilité des lignes et des contraintes réelles de l'atelier. L'ensemble est mis à la disposition de l'utilisateur à travers une interface simple, pensée pour le planificateur, sans qu'il ait besoin de comprendre la mécanique des modèles qui travaillent en arrière-plan.

Dans ce chapitre, nous ne nous limitons donc pas à décrire un programme : nous présentons une démarche complète, allant de la compréhension du besoin métier jusqu'à un système intelligent, fonctionnel et directement utilisable dans le contexte de l'entreprise. Les détails techniques de cette réalisation feront l'objet des sections qui suivent.

Si nous avons choisi d'investir ce sujet, c'est parce qu'il réunit ce qui nous tient à cœur : mettre l'intelligence artificielle au service d'un problème industriel tangible, et transformer une difficulté observée sur le terrain en une solution concrète. Au-delà de l'exercice académique, nous espérons que ce travail illustre comment des outils modernes, bien intégrés au métier, peuvent réellement soutenir la performance d'une chaîne de production.

2. Problématique

L'unité 23 de confection fonctionne actuellement avec un seul shift, ce qui rend la planification quotidienne déterminante pour le respect des délais et la bonne affectation des ordres de production (PO). Le planificateur doit répartir chaque PO sur les vingt lignes en tenant compte de la catégorie du produit, du temps standard, des compétences des opérateurs et de la disponibilité des lignes. La pertinence de cette affectation dépend directement de la capacité réelle de chaque ligne à produire une quantité donnée sur une journée.

Or, dans la pratique actuelle, la quantité planifiée par ligne et par jour repose essentiellement sur l'expérience des employés, et non sur une analyse rigoureuse des données historiques. Cette estimation empirique présente plusieurs limites :

- une incohérence récurrente entre les quantités planifiées et celles réellement produites en fin de journée ;
- une absence de maîtrise objective de l'efficacité des lignes, faute d'outil capable de la quantifier à partir des données ;
- Une dépendance forte au savoir-faire individuel : les nouveaux ingénieurs ne connaissent pas le comportement réel des lignes, et la consultation manuelle des archives reste une méthode lente, peu adaptée à une décision opérationnelle.

Ces limites se traduisent directement par des objectifs d'ordres non atteints, des alertes fréquentes, une utilisation non optimale des ressources et une difficulté à fiabiliser les engagements pris vis-à-vis des clients.

C'est dans ce contexte que s'inscrit le présent projet : concevoir une solution d'aide à la décision fiable et directement exploitable, fondée sur l'apprentissage automatique. En estimant à l'avance la quantité supportable par chaque ligne, l'outil permet de planifier les PO sur des bases objectives, de réduire les écarts entre prévision et réalisation et de rendre l'efficacité des lignes lisible et transmissible, indépendamment de l'expérience individuelle.

Ce constat et cette finalité conduisent à formuler la question centrale de ce travail :

Comment exploiter les données historiques de production de l'unité 23 afin de prédire de manière fiable la quantité réellement produite par modèle et par ligne, et d'appuyer une planification réaliste, supportable par les lignes de confection et conforme aux objectifs fixés ?

3. Hypothèse

À partir de la problématique posée et des méthodes retenues pour ce travail, plusieurs hypothèses peuvent être formulées. Elles orientent la démarche et seront confirmées ou nuancées par les résultats obtenus.

— Un modèle d'apprentissage automatique, correctement entraîné sur les données historiques de production et capable de capturer la relation entre les variables explicatives et la quantité réellement produite, peut fournir des prédictions suffisamment fiables et satisfaisantes pour servir de base à la planification quotidienne des lignes de confection.

Cette hypothèse centrale se décline en plusieurs hypothèses secondaires :

— **Pertinence des données historiques** : Les données historiques de l'unité 23 contiennent une information suffisante et exploitable pour expliquer la quantité réellement produite par ligne et par jour, à condition d'être correctement préparées et structurées.

— **Existence d'une relation apprenable** : Il existe entre les variables d'entrée (catégorie de produit, modèle, nombre d'opérateurs, temps standard, ligne.) et la quantité produite une relation non triviale, souvent non linéaire, qu'un modèle supervisé est capable d'apprendre mieux qu'une simple estimation empirique.

— **Supériorité de l'approche par apprentissage** : Une prédiction fondée sur un modèle entraîné sur l'historique est plus fiable et plus reproductible que l'estimation actuelle, basée sur l'expérience individuelle des employés et la consultation manuelle des archives.

— **Adéquation du modèle choisi** : Un modèle d'apprentissage supervisé adapté à la nature tabulaire des données de production, capable de capturer les interactions complexes et non linéaires entre les variables, devrait fournir des prédictions plus fiables et plus stables qu'une approche moins adaptée à ce type de données.

— **Exploitabilité opérationnelle** : La capacité de prédiction, une fois intégrée dans un outil d'aide à la décision, permet de planifier des quantités réellement supportables par les lignes, de réduire les écarts entre quantités planifiées et produites, et de rendre l'efficacité des lignes accessible aussi bien aux planificateurs expérimentés qu'aux nouveaux ingénieurs.

La vérification de ces hypothèses repose sur l'évaluation de la performance du modèle ; qualité des prédictions sur des données non vues et sur sa capacité à appuyer une planification cohérente et réaliste au sein de l'unité 23.

4. Méthode de travail

4.1 Compréhension du métier

La première phase consiste à analyser le fonctionnement général de l'unité de confection afin de cerner l'organisation des lignes, leur spécialisation par catégorie de produit (polo, basique, casual) et le cheminement complet d'un ordre de production, de sa réception jusqu'à son affectation effective sur les lignes. Cette compréhension permet de préciser les besoins réels du service planning, d'identifier les contraintes propres à l'atelier et de traduire le problème industriel en une formulation exploitable, à la fois par un modèle de prédiction et par une logique d'affectation.

4.2 Compréhension des données

Une fois le contexte métier maîtrisé, l'étape suivante porte sur la collecte et l'exploration des données historiques de production. Il s'agit de recenser les sources disponibles, d'examiner la structure et la qualité des fichiers, et de clarifier la signification des différentes variables ainsi que leur lien avec la quantité réellement produite par ligne et par jour. Cette exploration permet également de détecter les incohérences, les anomalies et les valeurs manquantes susceptibles d'affecter la suite des traitements.

4.3 Préparation des données

La préparation constitue une étape charnière, car des données brutes inadaptées compromettent la qualité du résultat final, quelle que soit la sophistication du modèle. Cette phase regroupe le nettoyage des données, le traitement des valeurs manquantes, l'uniformisation des formats, l'encodage des variables catégorielles et la création d'attributs dérivés pertinents. Les informations sont ainsi organisées dans une structure cohérente, directement exploitable pour l'entraînement des modèles de prédiction.

4.4 Modélisation et choix du modèle

La modélisation consiste à formuler le problème comme une tâche d'apprentissage supervisé visant à estimer la quantité réellement produite d'un article par ligne à partir des variables explicatives. Plusieurs approches sont envisagées et comparées afin de retenir le modèle le plus adapté à la nature des données et à l'objectif de prédiction. Cette phase comprend la définition des variables d'entrée et de la cible, le choix de l'algorithme, l'entraînement des modèles par famille de produits et le réglage de leurs paramètres.

4.5 Résolution

La résolution consiste à mobiliser les modèles entraînés pour estimer la production attendue de chaque ligne, puis à exploiter ces prédictions au sein d'une logique d'affectation des ordres de production. Les ordres sont répartis sur les lignes en tenant compte des capacités effectivement disponibles et des contraintes de l'atelier, dans une démarche itérative permettant d'explorer différentes configurations d'affectation au regard des objectifs fixés.

4.6 Résultats, interprétation et évaluation

Cette phase analyse les sorties produites et mesure leur pertinence opérationnelle. La qualité des prédictions est évaluée à l'aide d'indicateurs adaptés aux problèmes de régression, tandis que la pertinence de l'affectation est appréciée à travers le taux d'utilisation des lignes, l'écart

entre les quantités planifiées et prédites, et l'atteinte des objectifs des ordres. Lorsque les résultats apparaissent insuffisants, certains éléments de la modélisation ou de la préparation sont réajustés dans une logique d'amélioration continue.

4.7 Déploiement

La dernière phase consiste à rendre le système opérationnel dans son environnement réel d'utilisation, à travers la conception d'une interface interactive accessible aux utilisateurs métiers, l'automatisation du processus de prédiction et d'affectation, ainsi que la génération de plans de production et de tableaux de bord. L'objectif final est de mettre l'outil entre les mains des planificateurs et des responsables de production afin de faciliter la prise de décision et d'améliorer le pilotage opérationnel des lignes de confection.

5. Développement d'un Système Intelligent de Prédiction de la Production Journalière en Confection

Avant d'engager toute démarche de modélisation prédictive, nous avons jugé indispensable de comprendre en profondeur la réalité opérationnelle qui sous-tend notre problème : comment naît un ordre de production, de quoi est-il constitué, et selon quelle logique est-il acheminé vers une ligne de confection plutôt qu'une autre ?

5.1 Compréhension du métier

5.1.1 Qu'est-ce qu'un ordre de production ?

Dans le cadre de ce mémoire, nous utiliserons le terme ordre de production (PO) équivalent de l'ordre de fabrication (OF) dans la littérature scientifique pour désigner toute demande formalisée de fabrication émise par le service planification à destination de l'atelier de confection.

Un ordre de production ne naît pas isolément. Il découle d'une confirmation client : un ordre d'échantillonnage valide d'abord le modèle, puis le PO est créé. Dès lors, il devient simultanément planifiable et affectable. C'est un document de référence interne qui traduit un besoin qu'il provienne d'une commande client confirmée ou d'un besoin de réapprovisionnement en une instruction concrète adressée à l'atelier, contenant toutes les informations structurées nécessaires à son exécution :

Composante du PO	Description
ID du PO	Il s'agit du code ou numéro unique qui permet d'identifier l'ordre de production. Cet identifiant facilite le suivi du PO depuis sa création jusqu'à sa réalisation.
Date du PO	Elle correspond à la date de création, de lancement ou de planification de l'ordre de production. Elle permet de situer le PO dans le temps et de suivre son avancement.

Client	Il désigne le client concerné par l'ordre de production. Cette information permet de relier la production à une demande client précise et de respecter ses exigences.
Produit	Il représente l'article à fabriquer. Il peut s'agir d'un produit fini, d'un semi-produit ou d'un composant destiné à entrer dans un processus de fabrication.
Type de produit	Il permet de classer le produit selon sa nature générale.
Catégorie de produit	Elle correspond à une classification plus détaillée du produit selon ses caractéristiques, sa famille, son usage ou son processus de fabrication.
Quantité à produire	Elle indique le nombre d'unités qui doivent être fabriquées dans le cadre de l'ordre de production. Cette donnée permet d'estimer la charge de travail nécessaire.
Exigences du produit	Elles correspondent aux caractéristiques attendues du produit, comme les dimensions, les matières utilisées, les spécifications techniques, les couleurs, les modèles ou les particularités de fabrication.
Ressources requises	Elles regroupent les moyens nécessaires pour exécuter le PO, tels que les machines, les postes de travail, les opérateurs, les matières premières, les outillages et les compétences techniques.

Tableau 3.1 : Composantes principales d'un ordre de production

5.1.2 Les critères d'affectation vers une ligne

L'affectation des ordres de production ne s'effectue pas de manière arbitraire. Elle repose sur une analyse croisée de plusieurs critères interdépendants que le planificateur doit impérativement concilier :

- La disponibilité des ressources : une ligne, un poste ou un opérateur ne peut être mobilisé que s'il est effectivement libre sur la période concernée, en tenant compte des horaires, maintenances et affectations déjà réalisées.
- La capacité de production : chaque ligne dispose d'une capacité nominale exprimée en minutes productives. L'ordre ne peut lui être affecté que si la charge générée reste dans les limites de cette capacité ; dans le cas contraire, il sera fractionné ou décalé.
- Les compétences et qualifications techniques : certains modèles exigent des lignes spécialisées ou des opérateurs qualifiés. L'affectation doit respecter cette compatibilité entre le produit et la ressource.

- Les délais et priorités : tous les ordres n'ont pas la même urgence. Ceux liés à des commandes clients avec pénalités de retard sont traités prioritairement devant ceux destinés au réapprovisionnement du stock.
- Les contraintes de gamme opératoire : la fabrication d'un article suit une séquence logique d'opérations dont l'ordre doit être respecté. L'affectation ne peut ignorer ces contraintes de précédence.
- Le coût de production : dans une logique d'optimisation, affecter un ordre à une ressource coûteuse alors qu'une ressource équivalente et moins onéreuse est disponible constitue un gaspillage que nous cherchons à éviter.

Numéro d'Ordre	Modèle	Type	Quantité d'ordre	Quantité Planifiée	Ligne	Date de début	Date de fin
G23-013564	MP135	POLO	5 250	2288	line 6	07/02/2026	11/02/2026
G23-013858	42ZTAPOH04	POLO	594	601	line 2	17/02/2026	20/02/2026
G23-013535	MP135	POLO	3 938	2117	line5	17/02/2026	24/02/2026
G23-013570	MP135	POLO	3 938	2235	line5	02/03/2026	09/03/2026
G23-013584	MP135	POLO	5 250	1449	ligne7	09/04/2026	14/04/2026
G23-013584	MP135	POLO	5 250	1449	line 1	09/04/2026	14/04/2026
G23-014056	MT224	T-SHIRT	1 050	1063	line 12	12/04/2026	15/04/2026
G23-014212	MT351	T-SHIRT	2 142	1057	line 11	13/04/2026	16/04/2026
G23-013959	MT164	T-SHIRT	11 550	3778	line 8	13/04/2026	17/04/2026
G23-013981	MP135	POLO	6 825	2403	line5	14/04/2026	20/04/2026
G23-013942	WR108	Pyjama-Long	6 930	3051	line 13	14/04/2026	20/04/2026
G23-013804	BT152	T-SHIRT	3 150	3153	line 14	16/04/2026	22/04/2026
G23-013658	BT139	T-SHIRT	3 150	3053	line 8	16/04/2026	22/04/2026
G23-013588	MT301	T-SHIRT	10 500	10503	line 15	16/04/2026	27/04/2026
G21-000735	MG188	JOGGER	2 625	2653	line 17	16/04/2026	27/04/2026
G23-014369	DRICA	TANK TOP	4 091	4103	line 10	19/04/2026	27/04/2026
G23-014101	MO170	Short	1 050	1053	line 19	19/04/2026	22/04/2026
G23-013963	BT139	T-SHIRT	4 725	4728	line 9	19/04/2026	23/04/2026

Tableau 3.2 : Exemple sur des Ordres de Production planifiés et affectés

Le planificateur occupe ici un rôle central. Pour chaque ordre, il détermine les quantités à produire, les lignes concernées et les journées de fabrication, en conciliant deux impératifs : honorer les quantités commandées et respecter les délais de livraison. C'est l'ensemble de cette logique décisionnelle que nous avons cherché à capturer et à enrichir à travers notre modèle prédictif.

5.1.3 Règles d'affectation des ordres aux lignes

— Priorité à la ligne spécialisée : chaque ordre est d'abord orienté vers la ligne dédiée à sa catégorie (polo, basique, casual).

— Flexibilité encadrée : en cas d'urgence ou de saturation des lignes spécialisées, l'ordre peut être redirigé vers une ligne non spécialisée.

— Contrainte de charge : la charge d'un ordre sur une journée ne peut pas dépasser la capacité de la ligne.

— Contrainte de capacité : chaque ligne compte 50 postes ; lorsque deux ordres partagent une ligne, la somme de leurs opérateurs doit rester ≤ 50 .

— Affectation en parallèle : un ordre réparti sur plusieurs lignes le même jour pour tenir le délai, ou deux ordres traités simultanément de part et d'autre d'une même ligne (côtés A/B).

— Affectation en série : un second ordre enchaîné sur une même ligne le même jour lorsque le premier sature les postes tout en sous-utilisant la ligne.

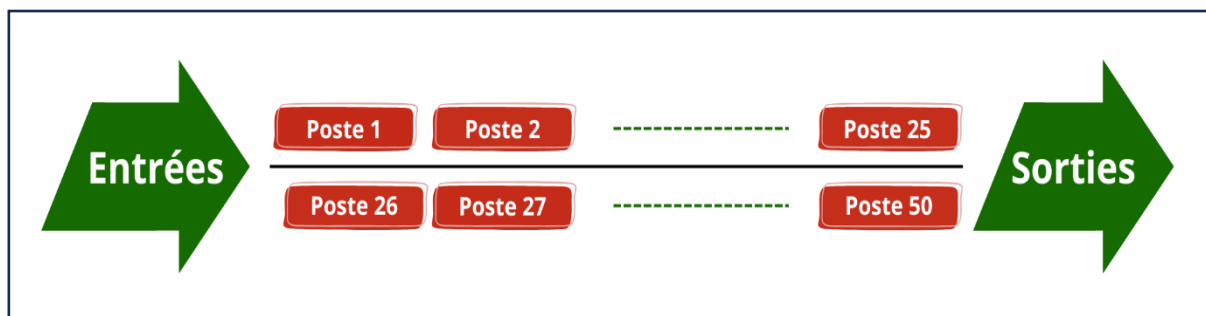


Figure 3.1 : Distribution des postes de travail sur une ligne de Confection

La difficulté de l'affectation tient à la combinaison de ces règles priorité aux lignes spécialisées, flexibilité maîtrisée, plafond de capacité et arbitrage entre série et parallèle. Un déséquilibre mal géré crée aussitôt des lignes sous-employées et des goulots, au détriment des délais. Ce constat justifie la solution que nous proposons : un outil d'aide à la décision qui, en s'appuyant sur la prédiction des quantités produites, répartit les ordres dans le respect de ces règles métier afin d'atteindre l'efficacité recherchée.

5.2 Compréhension et Préparation des données

5.2.1 Compréhension de données

Les données proviennent des archives Excel du département de méthodologie de l'unité 23 et couvrent la période du 5 janvier 2025 au 30 mars 2026, fenêtre suffisamment représentative des cycles de production tout en restant homogène en conditions d'exploitation. Nous raisonnons au niveau du modèle plutôt que de l'ordre de production : c'est la quantité produite d'un modèle donné sur une ligne donnée qui porte l'information utile, le couple (ligne, modèle) décrivant un comportement de production stable et exploitable. Le corpus assure ainsi la traçabilité de chaque modèle (ligne d'affectation, quantité planifiée par jour, sortie réelle en fin de journée, influencée par l'efficacité de la ligne) et constitue la base d'apprentissage permettant de prédire la production des futurs lancements.

Le département méthodologie met à disposition un rapport de production journalier, renseigné quotidiennement pour chaque ligne. Ce fichier, conçu pour le suivi opérationnel et le contrôle de gestion, est riche d'une trentaine de paramètres couvrant l'identification de la production, les ressources engagées, les mesures de performance constatées et des indicateurs de coût. Toutes ces informations ne sont cependant pas pertinentes pour notre objectif de prédiction.

La capture suivante présente un extrait de ce rapport source tel qu'il est tenu par l'entreprise :

GRUP/ GROUP	Hat Numar ası/ Line N°	Model	Tip / Type	Müşteri / Customer	Denge Bilgisi/ Balance Information			Detay/ Line Person Detail		Kapasite Dakika Detay/ Capacity Minute Detail					
					Operatör Sayısı Op.Count	Standart Time	Hedef %100	Makine / Operatörü	+/-	Çalışma Süresi/ Working Time	Verimlilik Hedefi/ Efficiency Target	Kapasite Dakikası/ Capacity Min	Kayıp Dakika/ Lost Min	Gerçek Kapasite Dakikası/ Reel Cap. Min	Üretim Dakikası/ Production Minute
POLO	1	MP252	POLO	SOL'S	28	10,22	1233	32	4	450	100%	14 400	60	14 340	9 198
POLO	2	MP127	POLO	EPA	25	9,25	1216	26	1	450	100%	11 700	0	11 700	8 325
CASUAL	3	GJ107	JOGGER	DRAOUZA	32	14,32	1006	33	1	450	75%	11 138	0	11 138	8 248
CASUAL	4	312360	SWEAT	DECATHLON/MODA-S	29	15,81	825	29	0	450	100%	13 050	0	13 050	2 972
POLO	5	MP135	POLO	MODA STYLE	25	9,25	1216	27	2	450	100%	12 150	120	12 030	9 250
POLO	6	MP182	POLO	EPA/MODA STYLE	26	9,61	1217	27	1	450	100%	12 150	0	12 150	8 649
POLO	7	MP127	POLO	EPA	25	9,25	1216	26	1	450	75%	8 775	0	8 775	4 625
BASIC	8	MT301	T SHIRT	MODA-STYLE	30	5,54	2437	33	3	450	100%	14 850	360	14 490	9 003
BASIC	9	MT301	T SHIRT	MODA-S/LE PRINTEMPS	30	5,54	2437	32	2	450	100%	14 400	360	14 040	9 418
BASIC	10	MT301	T SHIRT	MODA-S/LE PRINTEMPS	15	5,54	1218	17	2	450	100%	7 650	120	7 530	5 573
BASIC	11	MT301	T SHIRT	MODA-S/LE PRINTEMPS	30	5,54	2437	33	3	450	100%	14 850	180	14 670	11 135
BASIC	12	BT157	T SHIRT	LE PRINTEMPS/MODA-S	23	8,22	1259	30	7	450	100%	13 500	0	13 500	7 776
CASUAL	13	MC161	SWEAT	JACKET	38	8,08	2116	30	-8	450	100%	13 500	0	13 500	7 280
BASIC	14	MT301	T SHIRT	MODA STYLE	30	5,54	2437	31	1	450	100%	13 950	0	13 950	11 135
CASUAL	15	MC145	SWEAT	MODA STYLE	32	14,19	1015	28	-4	450	100%	12 600	0	12 600	6 811
CASUAL	16	MH165	HOODY	MODA STYLE/EMPREINTE	29	13	991	29	0	450	100%	13 050	0	13 050	7 902
CASUAL	17	121265	SWEAT	DECATHLON	31	16,05	869	38	7	450	100%	17 100	0	17 100	7 544
CASUAL	18	MH175	HOODY	LUMBERJACK/MODA-S	36	16,10	1006	35	-1	450	100%	15 750	0	15 750	2 093
CASUAL	19	MJ156	JOGGER	LUMBERJACK	46	20,54	1008	41	-5	450	100%	18 450	0	18 450	10 270

Tableau 3.3 : Rapport journalier de la production – Données de balance et de capacités

GRUP/ GROUP	Hat Numar ası/ Line N°	Model	Tip / Type	Müşteri / Customer	Üretim Miktarı Detay/ Production Quantity Detail			Performans Detay/ Performans Detail			Kayıp Departman/ Department Of Losing
					Planlanan Miktar/ Planned Qty	Giriş Miktarı/ Input Qty.	Çıkış Miktarı/ Output Qty.	Giriş Verimliliği/ Input Efficiency	Çıkış Verimliliği/ Output Efficiency	Çıkış Performansı/ Output Performans	
POLO	1	MP252	POLO	SOL'S	900	850	900	69%	64%	64%	30min problem of lockstitch machine
POLO	2	MP127	POLO	EPA	1 000	900	900	74%	71%	71%	
CASUAL	3	GJ107	JOGGER	DRAOUZA	525	620	576	82%	74%	74%	
CASUAL	4	312360	SWEAT	DECATHLON/MODA-S	440	336	188	41%	23%	23%	1hr problem of overlock machine
POLO	5	MP135	POLO	MODA STYLE	750	1 100	1 000	90%	76%	77%	
POLO	6	MP182	POLO	EPA/MODA STYLE	900	1 000	900	82%	71%	71%	
POLO	7	MP127	POLO	EPA	500	500	500	55%	53%	53%	2hrs problem of automat bottom
BASIC	8	MT301	T SHIRT	MODA-STYLE	2000	2 000	1 625	82%	61%	62%	
BASIC	9	MT301	T SHIRT	MODA-S/LE PRINTEMPS	2000	2 000	1 700	82%	65%	67%	
BASIC	10	MT301	T SHIRT	MODA-S/LE PRINTEMPS	1000	1 000	1 006	82%	73%	74%	1hr problem of coverstitch machine
BASIC	11	MT301	T SHIRT	MODA-S/LE PRINTEMPS	2040	2 100	2 010	86%	75%	76%	
BASIC	12	BT157	T SHIRT	LE PRINTEMPS/MODA-S	1750	1 006	946	80%	58%	58%	
CASUAL	13	MC161	SWEAT	JACKET	700	600	901	28%	54%	54%	
BASIC	14	MT301	T SHIRT	MODA STYLE	2000	2 000	2 010	82%	80%	80%	
CASUAL	15	MC145	SWEAT	MODA STYLE	350	450	480	44%	54%	54%	
CASUAL	16	MH165	HOODY	MODA STYLE/EMPREINTE	575	830	600	84%	61%	61%	
CASUAL	17	121265	SWEAT	DECATHLON	450	0	470	0%	44%	44%	
CASUAL	18	MH175	HOODY	LUMBERJACK/MODA-S	440	483	130	48%	13%	13%	
CASUAL	19	MJ156	JOGGER	LUMBERJACK	400	567	500	56%	56%	56%	

Tableau 3.4 : Rapport journalier de la production – Données sur la production et la performance

À partir de ce rapport, nous construisons une base d'apprentissage dédiée, base_rn.xlsx, en ne conservant que les variables exploitables pour la prédiction. Le tri repose sur un principe simple : une variable n'est retenue que si elle est disponible au moment de la planification et porte une information utile sur la production à venir.

Nous conservons ainsi les variables d'identification (groupe, ligne, modèle, type, client), les ressources et conditions de la journée (nombre d'opérateurs requis, temps de travail, temps standard, efficacité, capacité-minute) et la quantité planifiée. La variable cible est la quantité de sortie réellement produite en fin de journée.

Cette sélection écarte tout particulièrement les mesures de sortie (efficacité et performance constatées), dont la disponibilité postérieure à la production les rendrait inutilisables en situation réelle de planification. Les variables retenues sont ensuite réorganisées par catégorie dans les feuilles POLO_X, BASIC_X et CASUAL_X, avec leur cible respective POLO_Y, BASIC_Y et CASUAL_Y, avec une répartition des affectations de 2 053 pour POLO ,2858 pour BASIC et 1199 pour CASUAL.

La figure suivante présente la répartition des données étudiées par catégorie de produit sur la même période retenue :

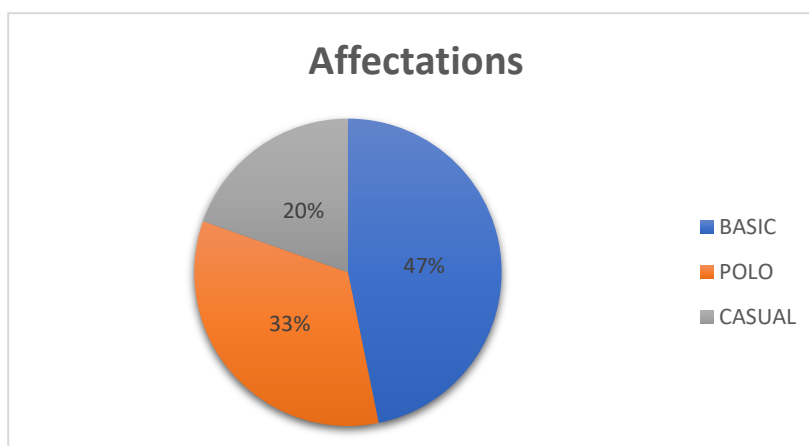


Figure 3.2 : Répartition des affectations par catégorie

La capture suivante présente un extrait de la feuille d'entrées POLO_X de la base ainsi constituée, qui rassemble les variables explicatives décrivant chaque journée de production d'un modèle sur une ligne :

Date	ligne_id	modele_id	Client	categorie	type	qte_planifiee_jour	nbr_opérateurs	temps_traitement_standard	temps_disponible_jour
05/01/2025	1	MP135	HPS	POLO	POLO	800	25	9,25	450
05/01/2025	2	MP135	HPS	POLO	POLO	800	25	9,25	450
05/01/2025	3	MP135	HPS	POLO	POLO	750	25	9,25	450
05/01/2025	4	MP135	HPS	POLO	POLO	750	25	9,25	450
05/01/2025	5	MP135	WAFAA	POLO	POLO	750	25	9,25	450
05/01/2025	6	MP127	EPA	POLO	POLO	750	25	9,46	450
05/01/2025	7	GT136	LE PRINTEMPS	POLO	T SHIRT	1750	22	4,06	450
06/01/2025	1	MP135	HPS	POLO	POLO	800	25	9,25	450
06/01/2025	2	MP135	HPS	POLO	POLO	800	25	9,25	450
06/01/2025	3	MP135	HPS	POLO	POLO	750	25	9,25	450
06/01/2025	4	MP135	HPS	POLO	POLO	750	25	9,25	450
06/01/2025	5	MP135	WAFAA/HPS	POLO	POLO	750	25	9,25	450
06/01/2025	6	MP135	HPS	POLO	POLO	750	25	9,25	450
06/01/2025	7	GT136	LE PRINTEMPS	POLO	T SHIRT	1750	22	4,06	450
07/01/2025	1	MP135	HPS	POLO	POLO	675	25	9,25	450
07/01/2025	2	MP135	HPS	POLO	POLO	630	25	9,25	450
07/01/2025	3	MP135	HPS	POLO	POLO	678	25	9,25	450

Tableau 3.5 : Données d'entrées exploitables pour l'entraînement

La capture ci-dessous présente la feuille cible POLO_Y, qui contient la variable à prédire, à savoir la quantité réellement produite par jour (qte_reelle_produite_jour).

qte_reelle_produite_jour
489
422
500
644
500
650
1047
720
730
630

Tableau 3.6 : Données de sortie exploitables pour l'entraînement

Choix des données :

Les modèles d'entraînement, de test et de prédiction reposent sur un même ensemble de 54 variables d'entrée, garantissant la cohérence entre l'apprentissage et l'utilisation réelle. La cible est, dans tous les cas, la quantité réellement produite par jour. Les variables se répartissent en cinq familles :

1. Variables d'identification

Variable	Choix justifié
ligne_id	Identifie la ligne ; porte son comportement productif propre
modele_id	Le modèle fabriqué conditionne la difficulté et la cadence
Client	Le client est lié à des exigences et un niveau de finition
categorie, type	Situent le produit dans sa famille (polo, basique, casual)
specialite_ligne	Indique si la ligne est sur sa catégorie dédiée ou en polyvalence

Tableau 3.7 : Variable d'identification

2. Ressources et conditions de la journée

Variable	Choix justifié
qte_planifiee_jour, target_jour	Objectif fixé : référence directe de la production attendue
Nbr_opérateurs, nb_operateurs_presentes, ecart_operateurs	L'effectif engagé est un déterminant majeur de la sortie
Temps_disponible, temps_traitement_standard	Temps disponible et temps standard du modèle
efficience_target, capacité_min	Capacité théorique et efficience visée de la ligne

Tableau 3.8 : Ressources et conditions de la journée

Certaines de ces variables, comme l'effectif réellement présent ou l'écart d'opérateurs, ne sont pas connues au moment de la planification. À la prédiction, elles ne sont donc pas saisies depuis la journée à venir mais estimées à partir du comportement historique de la ligne, ou renseignées par le planificateur à titre d'hypothèse. Cette distinction garantit que le modèle ne s'appuie jamais sur une information indisponible au moment de la décision, et écarte tout risque de fuite d'information.

3. Historique de comportement

Variable	Choix justifié
qte/performance/efficience_moyenne_hist_ligne	Résumé le rendement habituel de la ligne
..._hist_ligne_categorie	Même logique, affinée par catégorie de produit
qte/performance_moyenne_hist_ligne_modele	Comportement du couple précis ligne-modèle (variable la plus influente)
nb_fois_modele_realise	Mesure l'expérience accumulée sur le modèle (effet d'apprentissage)

Tableau 3.9 : Historique de comportement

4. Variables dérivées

Variable	Choix justifié
capacité_x_efficience	Croise capacité et efficience (capacité réellement mobilisable)
opérateurs_ratio	Effectif rapporté à la capacité de la ligne
temps_ratio	Temps de travail rapporté au temps standard
plan_vs_capacité	Ambition du plan au regard de la capacité
ratio_plan_vs_hist, delta_plan_vs_hist	Écart entre objectif planifié et production historique
ratio_eff_vs_perf_hist	Rapport entre efficience visée et performance réelle passée

Tableau 3.10 : Variables dérivées

5. Variables temporelles et de mémoire

Variable	Choix justifié
jour_semaine, mois, annee, jour_mois, semaine, jour, semaine	Capturent les effets de calendrier
est_mois_début, est_mois_fin	Repèrent les débuts/fins de mois (effets de cadence)
jour_sin/cos, mois_sin/cos, semaine_sin/cos	Encodage cyclique : préserve la continuité (fin de période proche du début)
lag_target_1/2/3	Production des journées précédentes (inertie de la ligne)
roullant_mean_3/7, roullant_std_3	Tendance et stabilité récentes de la production
lag_plan_1, roullant_plan_7	Mémoire de la planification récente

Tableau 3.11 : Variables temporelles et de mémoire

Les variables historiques, les retards (lags) et les moyennes glissantes sont calculés séparément pour chaque couple ligne-modèle, de sorte que chaque ligne ne mobilise que son propre passé, sans contamination entre lignes.

5.2.2 Préparation de données

Sur le plan méthodologique, il s'agit de données tabulaires et hétérogènes, mêlant variables numériques (quantités, temps, effectifs, capacités) et catégorielles (ligne, modèle, client, type), avec une dimension temporelle. La relation entre ces variables et la quantité produite est non linéaire : un surcroît d'opérateurs ne produit d'effet que dans la limite de la capacité, l'efficacité module l'effet du temps standard, et le comportement d'un modèle dépend de la ligne qui le fabrique. Ces interactions et effets de seuil orientent le choix d'un modèle capable de les capturer, comme présenté dans la section suivante.

La préparation transforme les archives brutes en une base propre et homogène, prête pour l'entraînement. Elle comprend :

- Uniformisation : harmonisation des champs textuels (espaces, casse), conversion des cellules vides en valeurs manquantes explicites, typage des colonnes et décomposition de la date en composantes (jour, semaine, mois, année).
- Retrait des fuites d'information : suppression des variables connues seulement après production (temps perdu, capacité réelle, efficacité et performance mesurées a posteriori), inutilisables au moment de la planification.
- Filtrage des enregistrements : suppression des quantités produites nulles ou absentes, des quantités planifiées nulles et des lignes fermées (closed). Ce point écarte les faux zéros ; lignes sur lesquelles un modèle n'a pas tourné, qui apparaîtraient à tort comme improductives ; difficulté à l'origine de l'échec de notre première approche [38].
- Imputation : valeurs numériques par la médiane, catégorielles par la modalité la plus fréquente.
- Encodage catégoriel : « target encoding » régularisé, qui remplace chaque modalité (ligne, modèle, client...) par la moyenne de production associée, évitant l'explosion de dimension d'un encodage classique [37].
- Attributs dérivés : ratios (opérateurs/capacité, temps/standard, plan/historique), encodage trigonométrique des variables cycliques, variables de mémoire (quantités passées, moyennes glissantes) calculées par couple ligne-modèle [37]. La cible est apprise en échelle logarithmique puis ramenée à son échelle d'origine, ce qui stabilise l'apprentissage et garantit des prédictions positives.

5.3 Modélisation et choix du modèle

La modélisation consiste à formuler le problème comme une tâche d'apprentissage supervisé de type régression, visant à estimer la quantité réellement produite d'un article sur une ligne à partir des variables explicatives. Plusieurs approches ont été envisagées et comparées afin de retenir le modèle le plus adapté à la nature des données et à l'objectif de prédiction.

5.3.1 Première approche : Base simple et les réseaux de neurones

Nous avons d'abord exploré un modèle de base simple, calculant la quantité moyenne produite par modèle, par ligne et par jour. Mais cette moyenne s'écarte de la quantité réelle, car la quantité produite varie fortement : il faut donc comprendre la relation souvent non linéaire entre les variables qui décrivent la capacité réelle de chaque ligne, ce qu'un modèle simple ne peut traduire.

Nous avons ensuite exploré une modélisation par réseau de neurones de type perceptron multicouche, avec trois réseaux distincts ; un par catégorie, codés en python . Chaque réseau comportait une architecture en couches successives 128, 64 puis 32 neurones, activation ReLU, régularisation par dropout [29], une sortie linéaire, et était entraîné par l'optimiseur Adam sur une perte quadratique [28], avec arrêt anticipé et réduction adaptative du taux d'apprentissage. Les variables catégorielles étaient encodées en indicatrices et les variables numériques normalisées. Les données étaient réparties en ensembles d'entraînement, de validation et de test

Cette voie n'a pas donné de résultats satisfaisants. Sur l'ensemble de test, les trois modèles présentaient un coefficient de détermination négatif ou quasi nul et des erreurs élevées au regard des quantités moyennes produites :

Catégorie	MAE	RMSE	R ²
POLO	283,73	357,68	-0,304
BASIC	509,47	611,76	0,044
CASUAL	183,00	232,08	-0,007

Tableau 3.12 : Critère d'évaluation de réseau de neurone

RESULTATS TEST - POLO	RESULTATS TEST - BASIC	RESULTATS TEST - CASUAL
MAE = 283.73	MAE = 509.47	MAE = 183.00
RMSE = 357.68	RMSE = 611.76	RMSE = 232.08
R ² = -0.3040	R ² = 0.0444	R ² = -0.0066

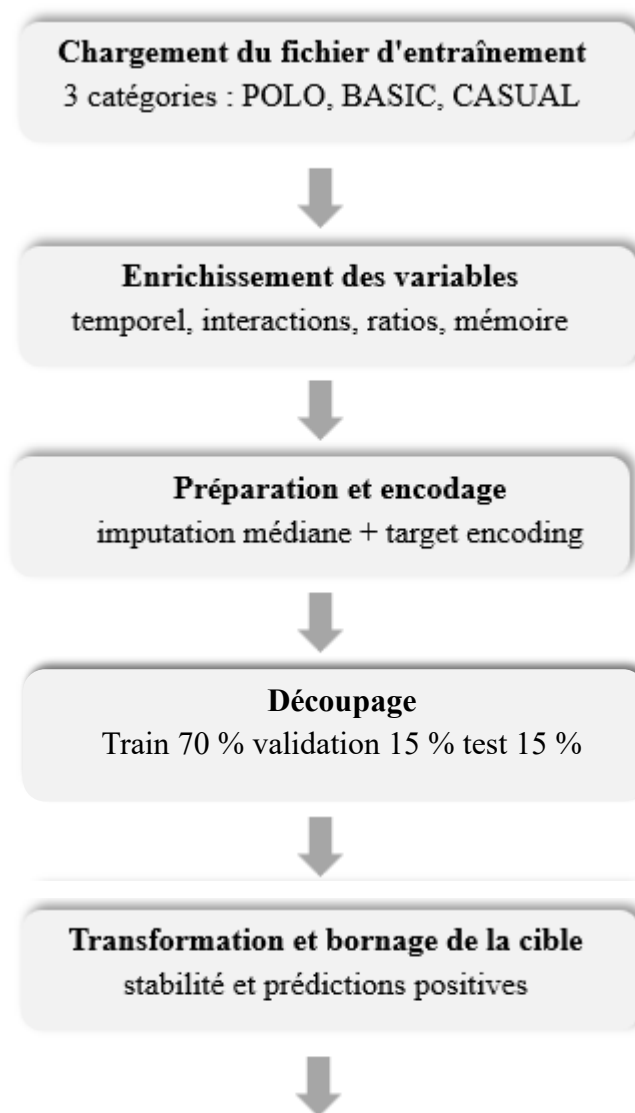
Figure 3.3 : Résultats de test des trois réseaux de neurones (POLO, BASIC, CASUAL)

Un R² négatif signifie que le modèle fait moins bien qu'une simple prédiction par la moyenne : le réseau n'apprenait pas la relation recherchée. Plusieurs facteurs l'expliquent. Le volume de données, de l'ordre de quelques milliers d'observations par catégorie, est modeste pour un réseau de neurones, plus adapté aux grands ensembles peu structurés. L'encodage en indicatrices des nombreuses modalités (ligne, modèle, client) augmentait fortement la dimension de la base sans apport informatif proportionnel. Enfin, la nature tabulaire des données et la dispersion de la cible se prêtaient mal à ce type de modèle. Ces résultats nous ont conduits à écarter cette approche.

5.3.2 Approche retenue : le gradient boosting (XGBoost)

Nous avons réorienté la modélisation vers un algorithme de gradient boosting, reconnu pour ses performances sur les données tabulaires de taille modérée, où les modèles à base d'arbres surpassent régulièrement les réseaux de neurones [9]. Ce choix répond directement aux limites précédentes : il capture les interactions complexes entre variables, ne nécessite aucune mise à l'échelle, gère efficacement les variables catégorielles sans explosion de dimension, et reste interprétable, l'importance de chaque variable étant directement mesurable [48].

La figure suivante présente le pipeline de modélisation appliqué à chaque catégorie de produit, depuis la base de données jusqu'au modèle entraîné :



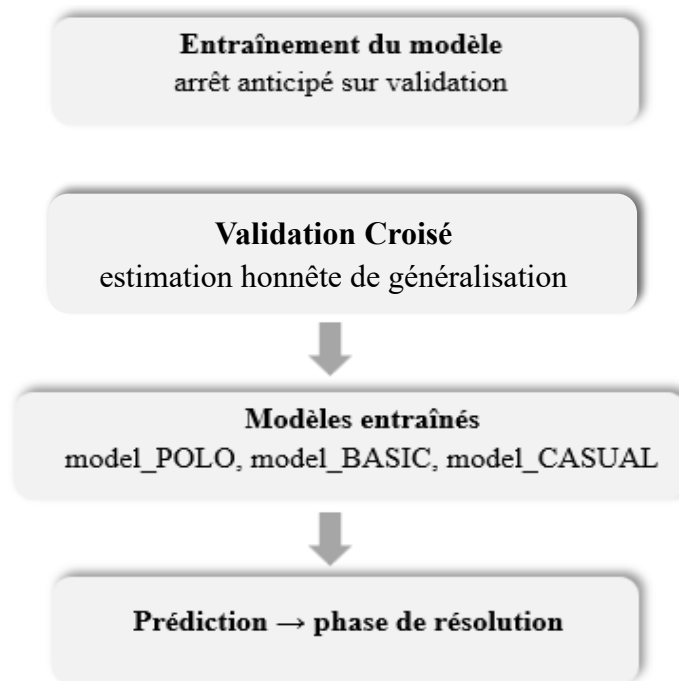


Figure 3.4 : Pipeline de modélisation XGBoost par catégorie

Modélisation du modèle de prédiction

La modélisation consiste à construire, pour chaque famille de produits, un modèle capable d'estimer la quantité réellement produite d'un article sur une ligne à partir des variables décrivant la journée de production. Le problème est posé comme une régression supervisée, la cible étant la quantité produite. Celle-ci est apprise sous une forme logarithmique : le modèle est entraîné sur le logarithme de la quantité, puis la prédiction est ramenée à l'échelle d'origine par la transformation inverse. Ce procédé stabilise l'apprentissage face à une cible dispersée et garantit des prédictions toujours positives.

Trois modèles indépendants sont construits, un par catégorie (POLO, BASIQUE, CASUAL), conformément à l'organisation de l'atelier en lignes spécialisées. Chaque catégorie dispose de sa feuille de variables et de sa feuille cible, et le pipeline les apparie automatiquement avant d'entraîner le modèle correspondant.

Enrichissement des variables

À partir des variables brutes, plusieurs attributs dérivés sont construits afin d'exprimer explicitement des relations utiles : des composantes temporelles (jour, semaine, début et fin de mois) et leur encodage cyclique, des interactions (capacité $\times$ efficacité, effectif rapporté à la capacité, temps de travail rapporté au temps standard, plan rapporté à la capacité), des rapports entre la planification et l'historique, ainsi que des variables de mémoire (productions des journées précédentes, moyennes et écarts-types glissants). Ces variables de mémoire sont calculées par couple ligne-modèle pour les segments polo et basique, et globalement par ligne pour le casual, dont les modèles sont trop rares pour qu'un calcul par groupe soit fiable.

Préparation et encodage

Un préprocesseur unique traite les deux types de variables : les variables numériques sont complétées par leur médiane, et les variables catégorielles, après imputation par la modalité la plus fréquente, sont transformées par encodage par la cible (target encoding) régularisé. Aucune normalisation n'est appliquée, le gradient boosting raisonnant par seuils.

Découpage des données

Les observations sont divisées en trois ensembles successifs : entraînement (70 %), validation (15 %) et test (15 %). Ce découpage temporel, et non aléatoire, est essentiel : il reproduit la situation réelle où le modèle est entraîné sur le passé pour prédire l'avenir, et évite toute fuite d'information du futur vers le passé. Les valeurs extrêmes de la cible sont par ailleurs bornées par catégorie afin qu'elles ne dominent pas l'apprentissage.

Entraînement

Chaque modèle est un gradient boosting XGBoost optimisant une erreur quadratique. L'entraînement s'appuie sur l'ensemble de validation pour un arrêt anticipé : les arbres sont ajoutés tant qu'ils améliorent la performance sur la validation, et la meilleure itération est retenue, ce qui limite le surapprentissage. Les hyperparamètres sont calibrés par famille, selon une stratégie volontairement régularisée ; beaucoup d'arbres, apprentissage lent, arbres peu profonds et termes de régularisation marqués.

Hyperparamètre	POLO	BASIC	CASUAL
Nombre d'arbres	2000	2000	2000
Taux d'apprentissage	0,012	0,015	0,015
Profondeur maximale	5	4	4
Échantillonnage lignes / variables	0,75 / 0,70	0,70 / 0,65	0,65 / 0,65
Régularisation α / λ	0,3 / 1,5	1,0 / 3,0	1,0 / 3,0
Gamma	0,1	0,5	0,5

Tableau 3.13 : Hyperparamètre calibrés par famille

Outre le découpage simple, la robustesse du modèle est appréciée par une validation croisée respectant l'ordre temporel, qui fournit une estimation plus honnête de sa capacité à généraliser.

Une fois entraîné, le modèle restitue, pour toute journée décrite par ces mêmes variables, une estimation de la quantité productible. L'exploitation de ces estimations pour le classement des lignes et l'affectation des ordres relève de la phase de résolution, présentée dans la section suivante.

5.4 Résolution

Le système repose sur trois fichiers Python complémentaires, qui s'enchaînent depuis l'apprentissage jusqu'à la décision.

5.4.1 De l'entraînement aux artefacts

L'entraînement est réalisé par le script `train_9.4`. Pour chaque catégorie, il construit le préprocesseur et le modèle, les entraîne sur la base préparée, puis enregistre le résultat sous forme d'artefacts réutilisables un modèle (`model_POLO.pkl`, `model_BASIC.pkl`, `model_CASUAL.pkl`) et son préprocesseur associé à l'aide de la bibliothèque `joblib`. Cette étape mobilise `pandas` et `numpy` [44] pour les données, `scikit-learn` [44] pour le pipeline de préparation et la validation, `category_encoders` [44] pour l'encodage par la cible, et `xgboost` pour le modèle [9]. L'entraînement a été lancé puis validé au regard des indicateurs d'évaluation, présentés dans la section suivante. Une fois validés, ces artefacts ne sont plus réentraînés : ils sont simplement chargés au moment de l'utilisation, ce qui rend la prédiction instantanée.

5.4.2 Le moteur de prédiction `predict_production.py`

Ce fichier exploite les artefacts produits par l'entraînement. Au démarrage, il charge les trois modèles et leurs préprocesseurs (`joblib.load`) ainsi que l'historique de production. Pour un ordre donné défini par sa catégorie, son modèle et sa quantité planifiée, le moteur parcourt l'ensemble des lignes de la catégorie. Pour chaque ligne, il reconstitue les paramètres opérationnels à partir de l'historique, selon une priorité décroissante : couple ligne-modèle, puis ligne seule, puis modèle seul, puis valeur médiane. Il compose alors une observation représentant la journée à venir, recalcule les variables dérivées, applique le préprocesseur (`preprocessor.transform`) et obtient une estimation par l'appel `model.predict`, ramenée à l'échelle d'origine par la transformation inverse du logarithme (`np.expm1`). Les lignes sont enfin classées par quantité prédite décroissante.

C'est ici que réside la force de l'approche XGBoost : chaque ligne reçoit une véritable prédiction issue du modèle, et non une moyenne ou une règle approximative. La ligne n'étant pas une sortie du modèle mais une variable d'entrée, le moteur peut estimer la production de n'importe quel couple ligne-modèle, y compris pour une combinaison jamais observée, en s'appuyant sur le comportement historique de la ligne. La transformation logarithmique garantit par ailleurs des prédictions toujours positives.

5.4.3 Le moteur d'affectation `prediction_core.py`

Ce fichier constitue l'interface de haut niveau du système : il importe le moteur de prédiction (`from predict_production import ...`) et l'enveloppe dans des fonctions directement exploitables chargement du prédicteur, prédiction d'un ordre ou d'un lot, et surtout le solveur automatique d'affectation. Ce solveur raisonne comme un planificateur. À la date de lancement, il lit le registre des affectations en cours et calcule, pour chaque ligne, les positions libres et les postes disponibles, en retranchant les opérateurs des ordres encore actifs. Il en déduit le nombre de côtés réalisables, sous la double contrainte des deux positions par ligne et du plafond de cinquante postes. Il génère ensuite les configurations possibles selon les quatre modes d'affectation, ne retient que celles qui couvrent l'objectif, et choisit celle dont l'excédent est minimal privilégiant, à excédent égal, le mode le plus simple. L'affectation retenue est enfin

inscrite dans le registre, qui constitue le plan de production. Cette étape s'appuie sur pandas et numpy, et réutilise les constantes de capacité définies dans le moteur de prédiction.

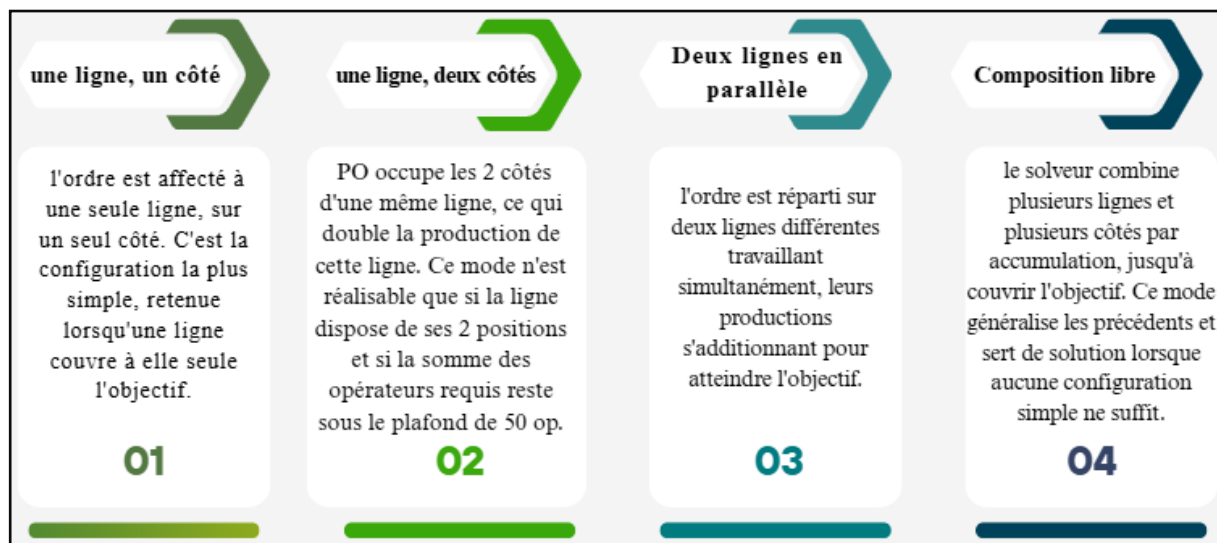


Figure 3.5 : Les quatre modes d'affectations

Le tableau ci-dessous récapitule les bibliothèques mobilisées à chaque étape :

Étape	Fichier	Bibliothèques principales	Instructions clés
Entraînement	train_9.4	xgboost, scikit-learn, category_encoders, pandas, joblib	model.fit, joblib.dump
Prédiction	predict_production.py	joblib, pandas, numpy	joblib.load, preprocessor.transform, model.predict, np.expm1
Affectation	prediction_core.py	pandas, numpy + l'import du moteur de prédiction	auto_affect, lecture/écriture du registre

Tableau 3.14 : Bibliothèques mobilisées au cours de la résolution du système

L'ensemble est rendu accessible aux utilisateurs par une interface web, qui appelle le moteur d'affectation sans que l'utilisateur ait à manipuler le code.

5.5 Résultats, interprétation et évaluation

Cette phase confronte les sorties produites par les modèles aux données réelles afin d'apprécier, d'une part, la qualité des prédictions et, d'autre part, la pertinence opérationnelle de l'affectation. L'analyse poursuit trois objectifs : mesurer la fiabilité des prédictions à l'aide d'indicateurs

adaptés aux problèmes de régression ; situer ce résultat en valeur mesurable par rapport à la première approche fondée sur les réseaux de neurones ; et vérifier que les prédictions, une fois intégrées à la logique d'affectation, conduisent à des plans réalistes. Lorsque les résultats apparaissent insuffisants, certains éléments de la modélisation ou de la préparation sont réajustés dans une logique d'amélioration continue.

L'évaluation est conduite sur le jeu de test; 15 % des observations, postérieur dans le temps aux données d'entraînement, jamais vu par les modèles, pour chacune des trois familles POLO, BASIC et CASUAL. La robustesse est en outre estimée par validation croisée respectant l'ordre chronologique CV R^2 , ce qui prémunit contre toute fuite temporelle[49].

5.5.1 Qualité des prédictions

Chaque critère est d'abord défini, puis interprété au regard du comportement du modèle de gradient boosting XGBoost [48] et comparé, en valeur mesurable, à la première approche par réseaux de neurones perceptron multicouche, ci-après RN, enfin évalué par rapport au seuil d'acceptabilité retenu.

5.5.1.1 Coefficient de détermination (R^2)

Le R^2 mesure la part de la variance de la quantité réellement produite expliquée par le modèle [46]. Il vaut 1 pour une prédiction parfaite et 0 pour un modèle équivalent à la simple moyenne ; une valeur négative signale un modèle pire que la moyenne.

Les deux figures suivantes classent les modèles par R^2 et confrontent R^2 de test et R^2 de validation croisée [46]:

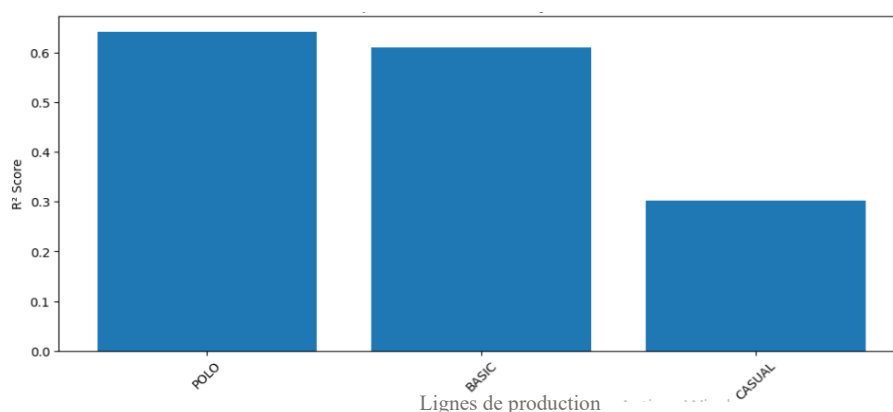


Figure 3.6 : Classement des modèles par R^2 (POLO, BASIC, CASUAL)

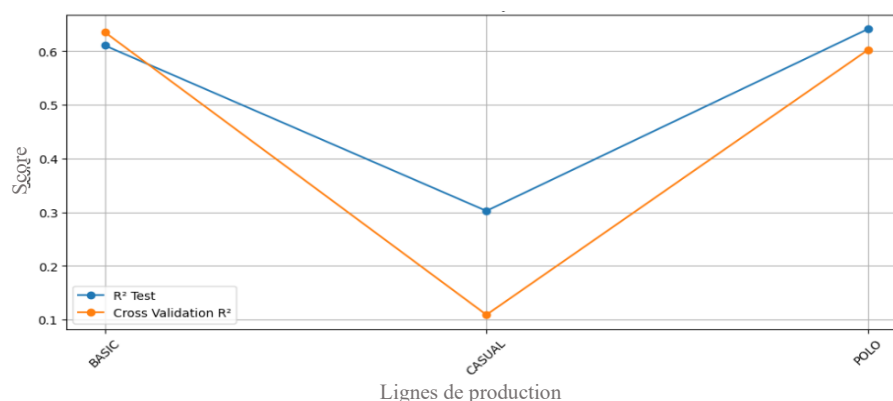


Figure 3.7 : R^2 de test et R^2 de validation croisée par ligne de production

Interprétation

XGBoost explique environ deux tiers de la variance sur POLO 0,656 et BASIC 0,624, et un peu moins d'un tiers sur CASUAL 0,303. La proximité des courbes « R^2 Test » et « Cross Validation R^2 » pour POLO et BASIC traduit une généralisation stable; le décrochage de la courbe de validation croisée sur CASUAL confirme l'instabilité déjà relevée. Le gain par rapport aux RN est décisif: la première approche atteignait $R^2 = -0,304$ pour POLO, 0,044 BASIC et $-0,007$ pour CASUAL, c'est-à-dire un modèle au mieux équivalent à la moyenne et le plus souvent moins bon qu'elle. Le passage à un modèle à base d'arbres, capable de capturer les interactions et les effets de seuil propres aux données tabulaires, fait basculer les trois segments du côté explicatif. Ce résultat valide concrètement le choix méthodologique hypothèses 2 et 4.

Évaluation

Pour des données d'atelier réelles, un R^2 de 0,5 à 0,7 est considéré comme acceptable et un R^2 supérieur à 0,3 comme exploitable. POLO et BASIC sont donc validés sans réserve, CASUAL est validé avec prudence: son R^2 reste supérieur au seuil minimal et son utilité pratique demeure.

5.5.1.2 Erreur absolue moyenne MAE

La MAE est la moyenne des écarts absolus, en pièces, entre quantité réelle et quantité prédite. Exprimée dans l'unité de la cible, elle indique de combien de pièces le modèle se trompe en moyenne, sans pénaliser exagérément les grands écarts [46]. La figure ci-dessous compare MAE et RMSE des trois modèles:

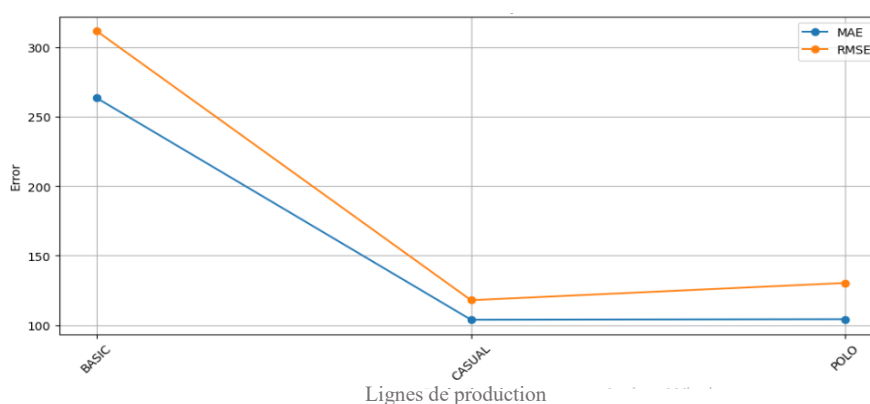


Figure 3.8 : Comparaison des métriques d'erreur (MAE et RMSE) par modèle.

Interprétation

La MAE s'établit à 101 pièces pour POLO, 260 pour BASIC et 104 pour CASUAL. L'écart entre BASIC et les deux autres tient à l'échelle des quantités : les ordres BASIC portent sur des volumes journaliers bien plus élevés, si bien qu'une MAE de 260 pièces y est proportionnellement modérée. Comparée aux RN MAE de 284 pour POLO, 509 pour BASIC et 183 CASUAL la réduction est nette : environ -64% sur POLO, -49% sur BASIC et -43% sur CASUAL. XGBoost commet donc, en moyenne, deux à trois fois moins d'erreur en pièces que le réseau de neurones, ce qui se traduit directement par un plan plus fiable.

Évaluation

La MAE est jugée satisfaisante : ramenée à la quantité moyenne produite, elle correspond à un écart relatif compatible avec une planification opérationnelle, et son recul mesurable par rapport aux RN constitue une amélioration tangible sur les trois segments.

5.5.1.3 Racine de l'erreur quadratique moyenne RMSE

La RMSE est la racine de la moyenne des carrés des erreurs. Comme elle élève les écarts au carré, elle pénalise fortement les grosses erreurs ponctuelles: son écart à la MAE renseigne sur la présence de valeurs aberrantes. Elle se lit sur la Figure 3.8 ci-dessus et sur les tableaux de métriques par segment Figures 3.9 a–c.

Interprétation

La RMSE vaut 128 pour POLO, 307 pour BASIC et 118 pour CASUAL. Le rapport RMSE/MAE reste modéré de l'ordre de 1,1 à 1,3, ce qui indique que les erreurs sont relativement homogènes et qu'aucune poignée d'observations extrêmes ne dégrade le modèle. Par rapport aux RN RMSE de 358 pour POLO, 612 pour BASIC pour et 232 pour CASUAL la baisse atteint environ –64 % pour POLO, –50 % pour BASIC et –49 % pour CASUAL. Le gradient boosting réduit donc non seulement l'erreur moyenne mais aussi l'amplitude des erreurs les plus pénalisantes, ce qui sécurise les engagements de production.

Évaluation

La RMSE confirme le diagnostic de la MAE : modèles robustes sur POLO et BASIC, exploitables sur CASUAL. Le contrôle du rapport RMSE/MAE valide l'absence d'outliers structurants après le bornage des valeurs extrêmes opéré en préparation.

5.5.1.4 Erreur pourcentage absolue moyenne MAPE

Présentation. La MAPE exprime l'erreur en pourcentage de la valeur réelle. Indépendante de l'échelle, elle permet de comparer des segments aux volumes très différents [11]. Les tableaux suivants restituent la RMSE et la MAPE de chaque modèle:

Metrique	Valeur	Interprétation
RMSE	127.70	Racine erreur quadratique moyen
MAPE	16.18%	Errae pourcentage absolue moyen

Tableau 3.15 : Tableau de métriques, segment POLO

Metrique	Valeur	Interprétation
RMSE	306.74	Racine erreur quadratique moyen
MAPE	20.12%	Errae pourcentage absolue moyen

Tableau 3.16 : Tableau de métriques, segment BASIC

Metrique	Valeur	Interprétation
RMSE	117.87	Racine erreur quadratique moyen

MAPE	35.47%	Erreur pourcentage absolue moyen
------	--------	-------------------------------------

Tableau 3.17 : Tableau de métriques, segment CASUAL

Interprétation

La MAPE atteint 16,2 % POLO, 20,1 % BASIC et 35,5 % CASUAL. Sur l'échelle d'interprétation usuelle, une MAPE inférieure à 10 % traduit une prévision excellente, de 10 à 20 % une bonne prévision, de 20 à 50 % une prévision raisonnable, et au-delà de 50 % une prévision faible. POLO se situe dans la catégorie « bonne », BASIC à la limite supérieure de cette catégorie, et CASUAL dans la catégorie « raisonnable ». La MAPE plus élevée de CASUAL s'explique mécaniquement : les quantités y sont plus faibles et plus dispersées, et une même erreur absolue $MAE \approx 104$, proche de POLO pèse proportionnellement davantage. C'est cohérent avec une famille très personnalisée, où la quantité varie fortement selon le client et la commande.

Évaluation

Les MAPE de POLO et BASIC valident une qualité de prévision bonne à raisonnable. Celle de CASUAL, bien que supérieure, reste dans la plage raisonnable et demeure exploitable dès lors que la prédiction est utilisée comme estimation de capacité et non comme valeur exacte interprétation qui est précisément celle de l'outil d'affectation .

5.5.1.5 Biais moyen

Le biais moyen est la moyenne des erreurs signées (erreur = réel – prédit). Un biais nul indique des erreurs symétriques ; un biais positif signale une tendance à sous-estimer la production, un biais négatif à la surestimer [46]. La distribution des erreurs du segment POLO illustre ce diagnostic:

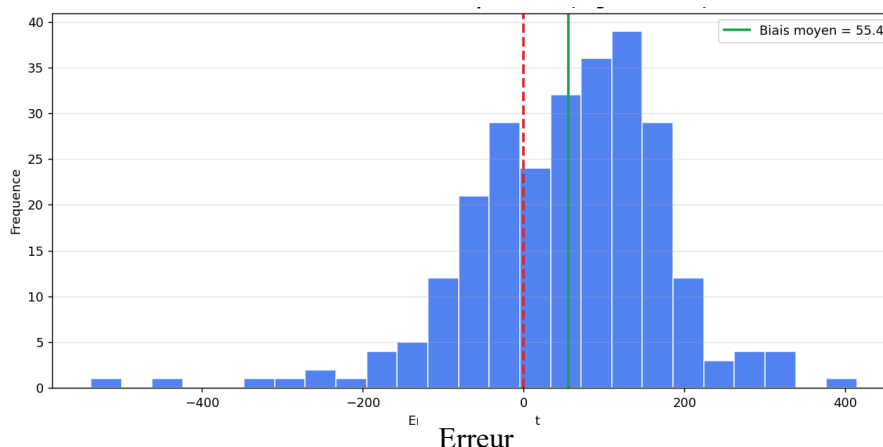


Figure 3.9 : Distribution des erreurs pour le segment POLO

Interprétation

Le biais moyen vaut environ +55 pièces sur POLO et +188 sur BASIC et il est quasi nul sur CASUAL ($\approx +1$). Cette légère sous-estimation sur POLO et BASIC n'est pas un défaut au regard de l'usage visé : en matière de planification, mieux vaut un modèle prudent, qui annonce une capacité atteignable, qu'un modèle optimiste qui promettrait des quantités non tenues. Ainsi, ces valeurs restent très cohérentes avec les grandes quantités produites chez TAYAL. Le biais quasi nul de CASUAL montre que, malgré sa dispersion plus forte, le modèle ne penche pas systématiquement d'un côté.

Évaluation

Le biais est faible et orienté du « bon côté » pour la planification. Il est jugé acceptable et conforte la lisibilité opérationnelle des prédictions, la sous-estimation modérée se traduisant naturellement par des affectations sécurisées.

5.5.2 Diagnostics de validité du modèle

Au-delà des indicateurs chiffrés, plusieurs représentations graphiques permettent de vérifier la santé des modèles: absence de sur-apprentissage, structure des erreurs, normalité des résidus, fidélité du suivi temporel et cohérence des variables explicatives:

5.5.2.1 Courbes d'apprentissage

Les courbes d'apprentissage tracent l'erreur sur l'ensemble d'entraînement et sur l'ensemble de validation au fil des itérations de boosting. Leur comparaison révèle un éventuel sur-apprentissage et justifie l'arrêt anticipé à la « meilleure itération » [46].

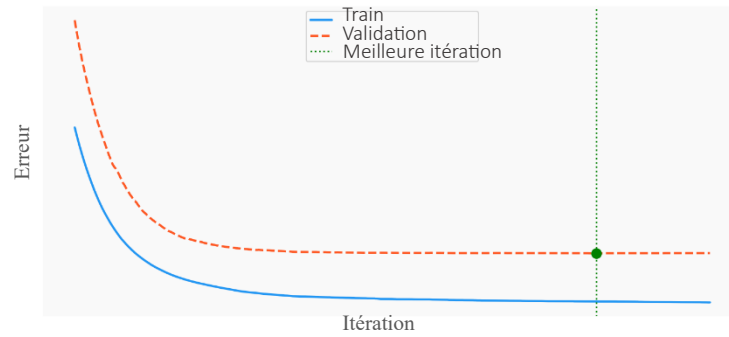


Figure 3.10 : Courbes d'apprentissage, POLO

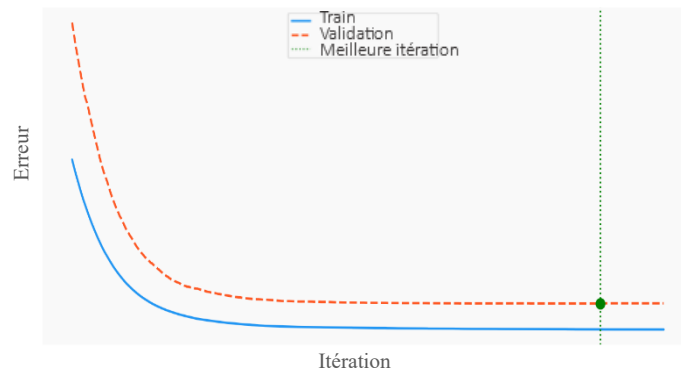


Figure 3.11 : Courbes d'apprentissage, BASIC

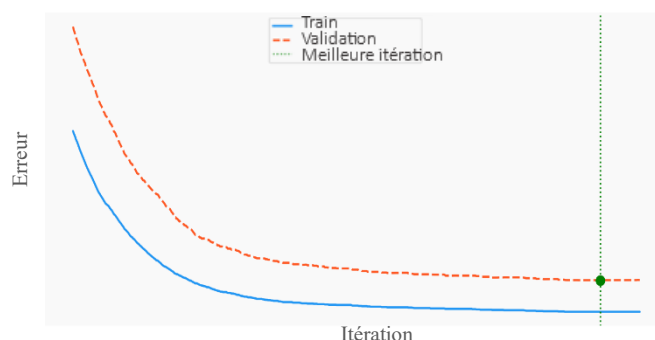


Figure 3.12 : Courbes d'apprentissage, CASUAL

Interprétation

Sur les trois segments, les deux courbes décroissent puis se stabilisent en restant proches l'une de l'autre, sans divergence de la courbe de validation. C'est la signature d'un modèle sain : il apprend la relation sans mémoriser le bruit. L'arrêt anticipé fige le modèle à la meilleure itération (917 pour POLO, 664 pour BASIC, 668 pour CASUAL), évitant les

arbres surnuméraires. Ce comportement contraste avec la première approche, où le réseau de neurones pénalisé par l'explosion de dimension liée à l'encodage en indicatrices et par un volume de données modeste ne parvenait pas à généraliser.

Évaluation

Le diagnostic « Risque d'overfitting: Faible (Sain) » de l'audit est confirmé visuellement pour les trois modèles. La validité statistique des résultats du jeu de test est donc établie.

5.5.2.2 Nuage de points

Ce nuage place chaque observation selon sa quantité réelle (abscisse) et sa quantité prédite (ordonnée). Une prédiction parfaite aligne les points sur la diagonale ; la dispersion autour de cette droite mesure visuellement l'erreur [46]:

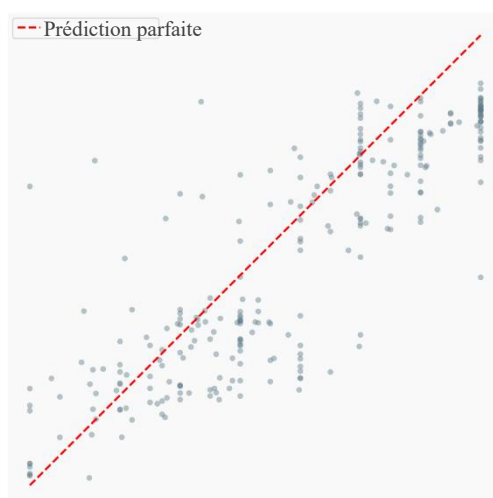


Figure 3.13 : Prédit vs réel, segment POLO

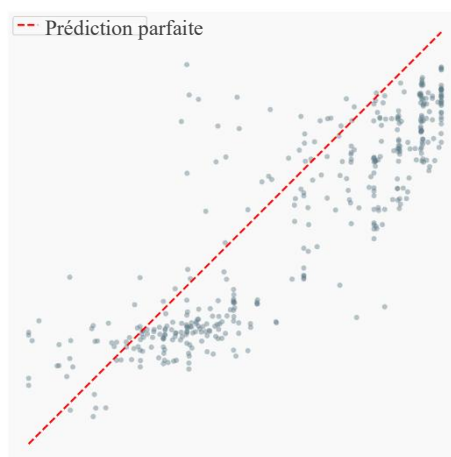


Figure 3.14 : Prédit vs réel, segment BASIC

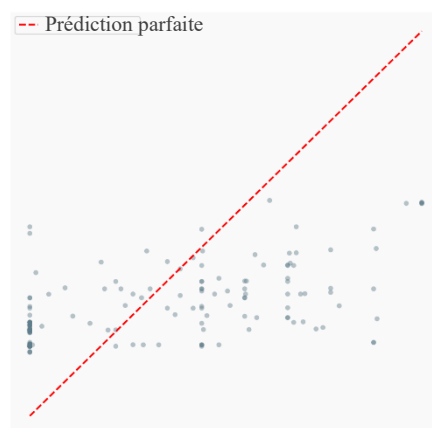


Figure 3.15 : Prédit vs réel, segment CASUAL

Interprétation

Pour POLO et BASIC, le nuage s'organise nettement autour de la diagonale, avec un léger tassement sous la droite aux fortes quantités, ce qui rejoint le biais positif mesuré en. Les alignements verticaux visibles correspondent à des quantités réelles fréquentes et arrondies (250, 500, 800, 1000...). Pour CASUAL, la dispersion est plus large et le nuage moins resserré, image fidèle d'un R^2 plus faible et d'une catégorie plus hétérogène. À titre de repère, la figure POLO annote un instantané d'évaluation à $R^2 \approx 0,62$; l'audit consolidé retient 0,656, l'écart provenant du sous-ensemble d'évaluation et ne modifiant pas la conclusion.

Évaluation

La structure des nuages confirme les R^2 : forte cohérence prédit/réel sur POLO et BASIC, cohérence plus lâche mais réelle sur CASUAL. Aucun comportement pathologique (nuage horizontal, prédictions constantes) n'apparaît, ce qui n'était pas le cas des RN.

5.5.2.3 Distribution des résidus et résidus en fonction du prédit

Ces doubles graphiques associent l'histogramme des erreurs (à gauche) et le tracé des résidus en fonction de la quantité prédite (à droite). Le second permet de détecter une éventuelle hétéroscédasticité (erreurs croissant avec le niveau de production) [46]:

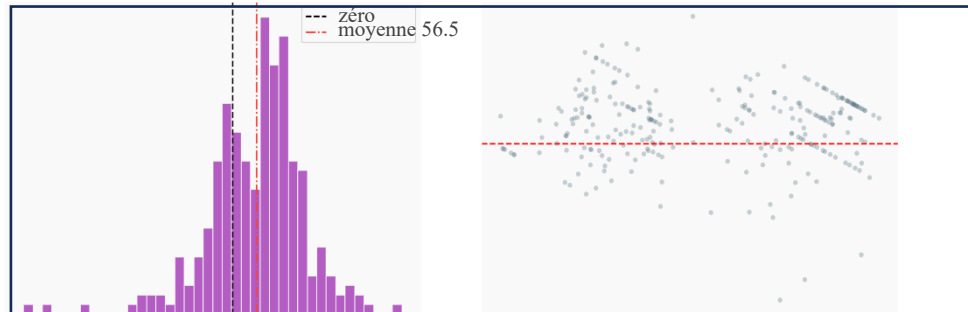


Figure 3.16 : Analyse des résidus, POLO

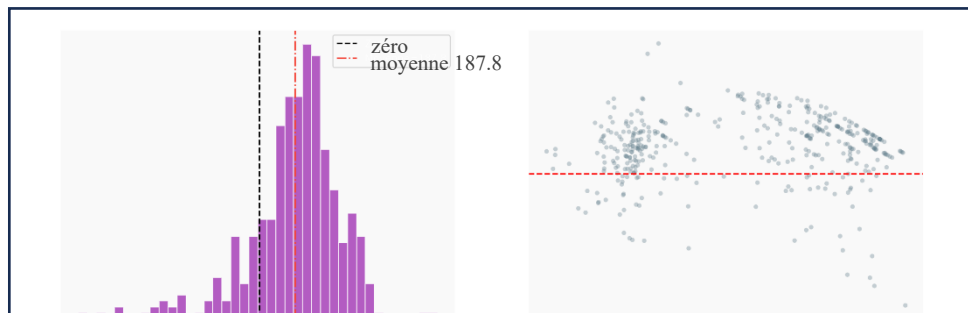


Figure 3.17 : Analyse des résidus, BASIC

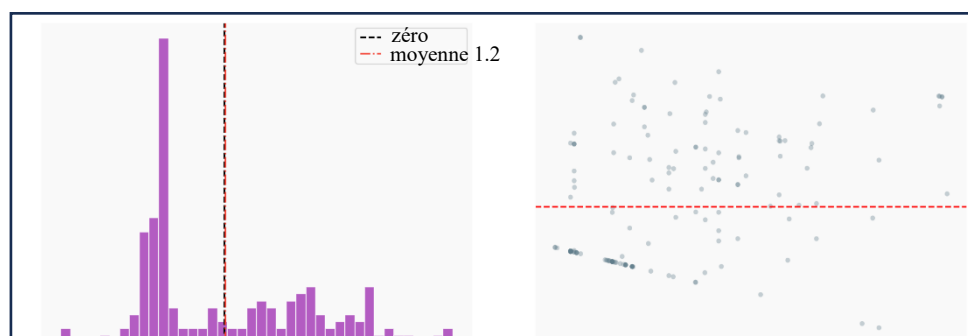


Figure 3.18 : Analyse des résidus, CASUAL

Interprétation

Le panneau de gauche reprend la distribution des erreurs déjà commentée au (centrage et biais) l'analyse se concentre ici sur le panneau de droite. Les résidus se répartissent de part et d'autre de la ligne zéro sans tendance marquée, ce qui indique une erreur globalement non structurée. On observe néanmoins une dispersion qui s'élargit légèrement vers les fortes quantités prédites (hétéroscédasticité modérée): le modèle est un peu moins précis sur les très gros volumes, comportement attendu et sans gravité pour l'usage. Les bandes obliques résultent du caractère discret et arrondi des quantités cibles.

Évaluation

La structure des résidus est saine: pas de biais dépendant du niveau, pas d'erreur explosant sur une plage particulière. L'hétéroscédasticité modérée constatée est documentée et compatible avec une exploitation opérationnelle.

5.5.2.4 Droite de Henry (QQ-plot des résidus)

Le QQ-plot compare les quantiles des résidus à ceux d'une loi normale. Un alignement sur la droite indique des résidus approximativement gaussiens, gage de fiabilité des intervalles d'erreur.

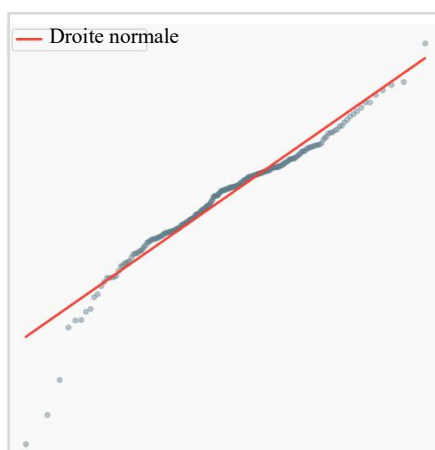


Figure 3.19 : QQ-plot des résidus, POLO

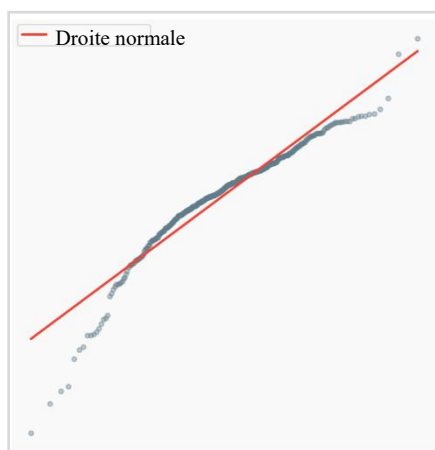


Figure 3.20 : QQ-plot des résidus, BASIC

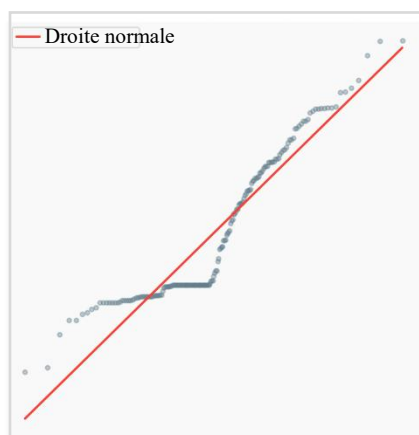


Figure 3.21 : QQ-plot des résidus, CASUAL

Interprétation

Sur les trois segments, les points suivent globalement la droite normale dans la partie centrale, avec des écarts aux extrémités, typiques de données industrielles comportant quelques journées atypiques. BASIC présente l'alignement le plus net, CASUAL le plus irrégulier, en cohérence avec son R^2 plus faible.

Évaluation

La quasi-normalité des résidus au centre de la distribution valide l'hypothèse d'erreurs raisonnablement régulières et conforte l'usage des indicateurs moyens. Les écarts en queue, limités, sont jugés acceptables.

5.5.2.5 Suivi temporel : quantité réelle vs prédite

Ces courbes superposent, dans l'ordre chronologique du jeu de test, la quantité réellement produite et la quantité prédite. Elles montrent la capacité du modèle à suivre les variations jour après jour.

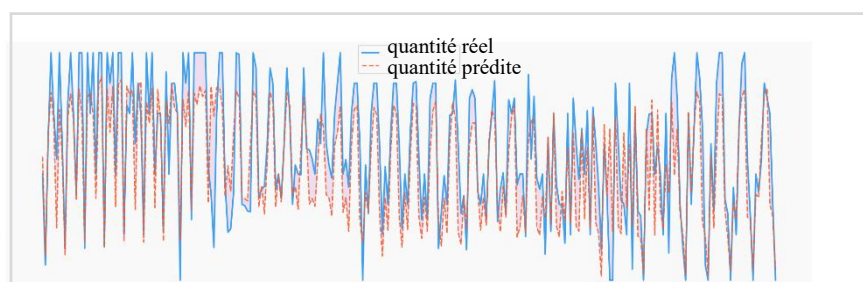


Figure 3.22 : Réel vs prédit (série), POLO

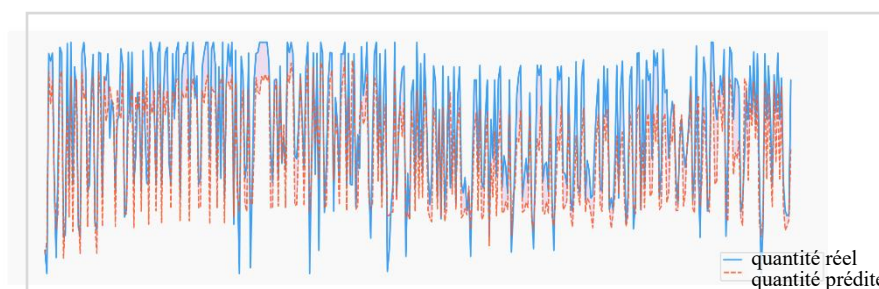


Figure 3.23 : Réel vs prédit (série), BASIC



Figure 3.24 : Réel vs prédit (série), CASUAL

Interprétation

La courbe prédite épouse les montées et descentes de la courbe réelle, y compris les fortes oscillations propres à un atelier de confection. Le modèle reproduit la dynamique (alternance de hautes et basses journées) tout en lissant légèrement les pics extrêmes, ce qui explique l'aire résiduelle teintée entre les deux courbes et confirme la sous-estimation modérée des sommets. POLO et BASIC suivent fidèlement la tendance ; CASUAL reste corrélé mais avec des écarts plus francs sur certaines pointes.

Évaluation

La concordance est excellente sur POLO et BASIC, et satisfaisante sur CASUAL où subsistent quelques inversions ponctuelles.

Le suivi temporel est jugé fidèle : le modèle capte la structure de la production réelle et non un simple niveau moyen, ce qui le distingue radicalement des RN dont le R^2 négatif traduisait l'incapacité à suivre ces variations.

À l'échelle agrégée pertinente pour la décision, les prédictions sont fidèles. Cette vue conforte la valeur opérationnelle du modèle : il restitue correctement la capacité relative des lignes et des modèles.

5.5.2.6 Importance des variables et valeurs SHAP (interprétabilité)

L'importance par gain mesure la contribution de chaque variable à la réduction d'erreur des arbres[40]. Les graphiques SHAP vont plus loin : le diagramme en barres ordonne les variables par contribution moyenne, et le diagramme « beeswarm » montre, pour chaque observation, le sens de l'effet (bleu = valeur faible, rouge = valeur élevée) sur la prediction:

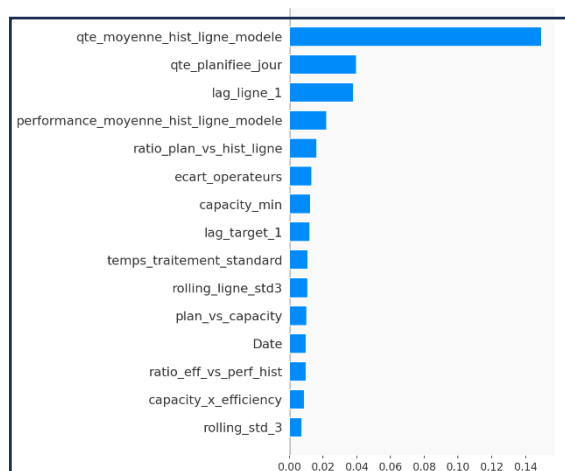


Figure 3.25 : Importance SHAP moyenne, POLO

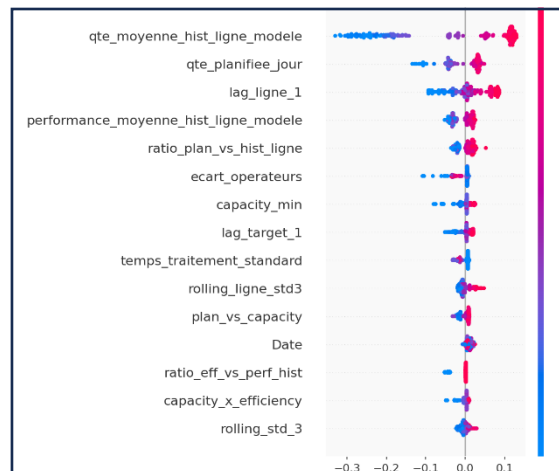


Figure 3.26 : Diagramme SHAP beeswarm, POLO

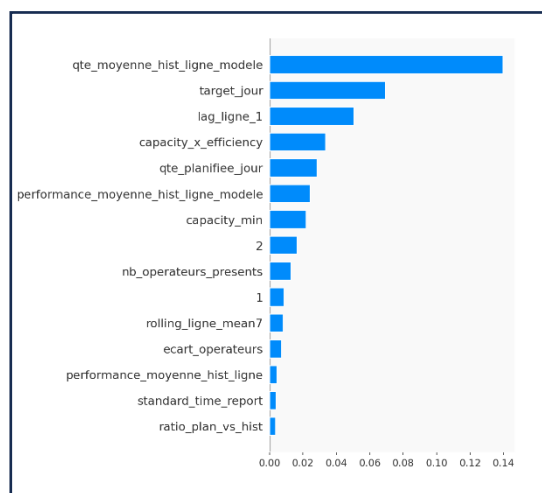


Figure 3.27 : Importance SHAP moyenne, BASIC

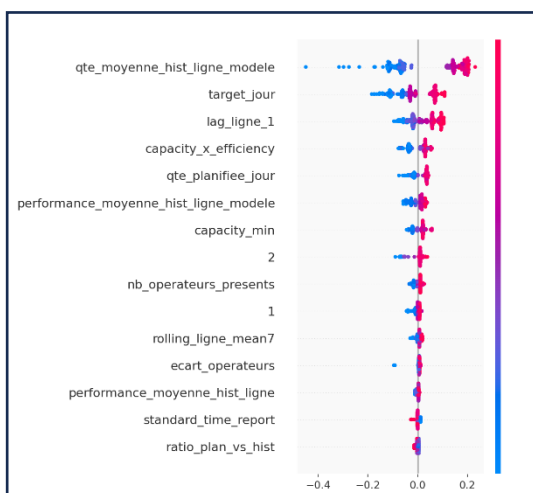


Figure 3.28 : Diagramme SHAP beeswarm, BASIC

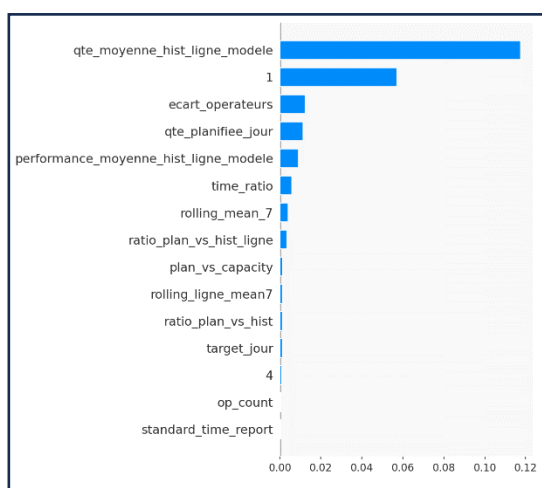


Figure 3.29 : Importance SHAP moyenne, CASU

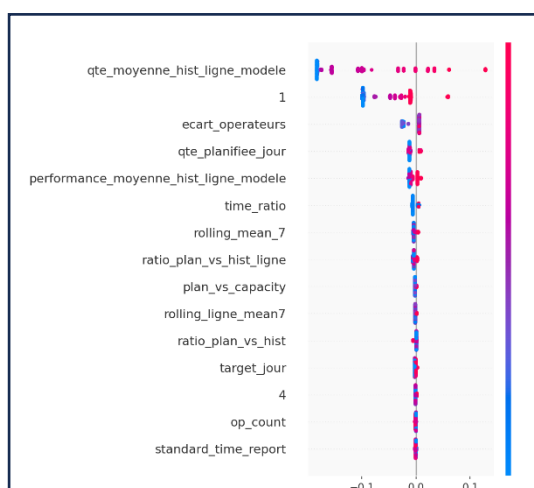


Figure 3.30 : Diagramme SHAP beeswarm, CASUAL

Interprétation

Sur les trois segments, la variable `qte_moyenne_hist_ligne_modele` domine très largement : le comportement historique du couple ligne-moèle est le déterminant majeur de la production, exactement comme le laissait attendre l'analyse métier. Viennent ensuite l'objectif du jour `qte_planifiee_jour` / `target_jour`, la mémoire récente de la ligne `lag_ligne_1`, `rolling_ligne_mean7`, la performance historique et des variables de capacité et d'effectif `capacity_x_efficiency`, `capacity_min`, `ecart_operateurs`. Le beeswarm confirme le sens attendu des effets : une moyenne historique élevée (points rouges) pousse la prédiction vers le haut, une moyenne faible (points bleus) vers le bas. Surtout, aucune variable connue seulement après la production n'apparaît : la suppression des fuites d'information opérée en préparation est validée.

Évaluation

Le modèle est interprétable et ses leviers sont cohérents avec le métier, ce qui renforce la confiance du planificateur et la transmissibilité de la connaissance (hypothèse 5). La hiérarchie stable des variables d'une catégorie à l'autre atteste de la robustesse de la modélisation.

5.5.3 Synthèse comparative : XGBoost vs réseaux de neurones

Le tableau suivant met en regard, en valeur mesurable, la première approche : réseaux de neurones et l'approche retenue XGBoost, pour les trois indicateurs principaux:

Segment	Métrique	RN	XGBoost	Gain
POLO	R ²	-0.304	0.656	Passage explicatif
POLO	MAE	284	101	-64 %
POLO	RMSE	358	128	-64 %
BASIC	R ²	0.044	0.624	* 14
BASIC	MAE	509	260	-49 %
BASIC	RMSE	612	307	-50 %
CASUAL	R ²	-0.007	0.303	Passage explicatif
CASUAL	MAE	183	104	-43 %
CASUAL	RMSE	232	118	-49 %

Tableau 3.18 : Comparaison mesurable des deux approches

Interprétation

Le changement de famille de modèle produit un saut de performance sur tous les segments et tous les indicateurs. Les réseaux de neurones, inadaptés à un volume de données modéré et à des variables catégorielles à forte cardinalité encodées en indicatrices, n'apprenaient pas la relation; R² négatif ou quasi nul. Le gradient boosting, qui raisonne par seuils et capte naturellement les interactions et effets de seuil des données tabulaires, divise par deux à trois l'erreur et fait passer les R² du côté explicative [42].

Évaluation

La supériorité de l'approche par arbres est démontrée en valeur mesurable, ce qui valide le choix méthodologique (hypothèses 3 et 4) et justifie a posteriori la réorientation décidée après l'échec de la première approche.

5.5.4 Pertinence de la prédiction et de l'affectation

5.5.4.1 Lecture des sorties de prédiction

Une fois la qualité des prédictions établie, le moteur de prédiction classe les lignes de la catégorie par quantité prédite décroissante et restitue, pour un ordre donné, la capacité estimée de chaque ligne. Ces sorties sont présentées au planificateur sous forme de tableaux exploitables.

Dans les tableaux de bord, l'objectif correspond à la quantité planifiée du PO par jour. Lorsqu'une ligne ne « monte » qu'à une valeur inférieure à cet objectif alors que le PO en vise davantage, ce n'est pas une erreur du modèle : c'est la prédiction réaliste de la capacité de cette ligne sur la journée. Le solveur peut alors affecter le PO sur plusieurs côtés d'une même ligne ou sur plusieurs lignes, et choisit automatiquement le mode d'affectation en tenant compte de l'état actuel des lignes de confection; positions libres, postes disponibles, ordres encore actifs et en retenant la configuration dont l'excédent est minimal.

5.5.4.2 Écart planifié vs prédit et atteinte des objectifs

Le graphique d'avancement cumulé confronte, jour ouvré après jour ouvré, le cumul de production prédite à l'objectif planifié du PO. L'aire colorée matérialise l'avance (vert) ou le retard (rouge) prévisionnel, c'est-à-dire l'écart entre ce qui est planifié et ce que les lignes peuvent réellement soutenir :

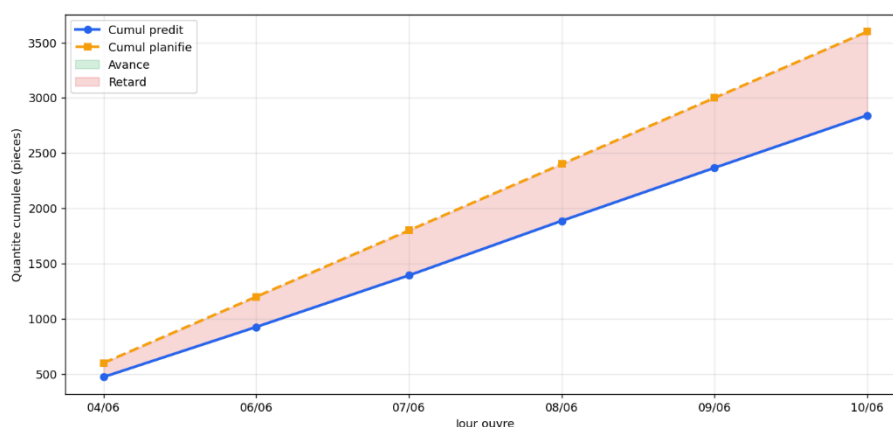


Figure 3.31 : Avancement cumulé du PO : production prédite vs objectif planifié

Interprétation

Le cumul prédit progresse parallèlement au cumul planifié mais reste légèrement en deçà, créant une zone de retard prévisionnel qui s'élargit au fil des jours. Loin d'être un défaut, cette lecture est l'apport central de l'outil : elle révèle à l'avance qu'avec la configuration courante l'objectif ne sera pas tenu, et de combien. Le planificateur dispose ainsi, dès le lancement, d'une alerte chiffrée lui permettant de réagir : ajouter un côté, mobiliser une seconde ligne, revoir la quantité journalière au lieu de constater l'écart en fin de PO comme dans la pratique empirique actuelle.

Évaluation

La pertinence de l'affectation se mesure ici à sa capacité à rendre l'écart planifié/prédit visible et actionnable avant production. L'objectif d'aide à la décision est atteint : l'outil transforme une estimation tacite, dépendante de l'expérience individuelle, en une trajectoire prévisionnelle objective et en un plan d'affectation optimisé sous contraintes d'atelier.

5.5.4.3 Évaluation opérationnelle d'ensemble

Sur le plan opérationnel, la chaîne complète : prédiction de quantité produite par jour par ligne, calcul de l'écart à l'objectif, choix automatique du mode d'affectation répond aux limites identifiées dans la problématique : elle réduit l'incohérence entre quantités planifiées et produites, quantifie objectivement l'efficacité des lignes et rend cette information accessible aussi bien aux planificateurs expérimentés qu'aux nouveaux ingénieurs (hypothèses 1 et 5). La qualité de prédiction validée garantit que ces décisions reposent sur des estimations fiables et reproductibles.

5.5.5 Logique d'amélioration continue

Lorsque les résultats apparaissent insuffisants, certains éléments de la modélisation ou de la préparation sont réajustés. Le cas du segment CASUAL illustre cette démarche.

Avec un R^2 de test de 0,303, une validation croisée plus basse et instable ($0,111 \pm 0,232$) et une MAPE de 35,5 %, CASUAL est le segment le plus fragile. Ce résultat s'explique par un volume de données plus restreint; 1 199 observations, soit 19,6 % du corpus et par une famille intrinsèquement plus personnalisée et complexe, dont les quantités varient fortement selon le client et la commande. Il demeure toutefois exploitable : son R^2 dépasse le seuil minimal de validité; 0,30, son MAE absolu; 104 pièces est comparable à celui de POLO, et son biais est quasi nul. Comparé aux modèles POLO et BASIC, plus stables, CASUAL est donc à interpréter comme une estimation prudente de capacité plutôt que comme une prévision exacte usage parfaitement compatible avec la logique d'affectation, qui n'exige pas une valeur ponctuelle mais une capacité réaliste à composer.

5.5.6 Conclusion de l'évaluation

L'évaluation valide la démarche en valeur mesurable. Les indicateurs de régression confirment des modèles fiables sur POLO et BASIC et exploitables sur CASUAL, avec un gain décisif et chiffré par rapport à la première approche par réseaux de neurones. Les diagnostics complémentaires courbes d'apprentissage saines, résidus non structurés et quasi normaux, suivi temporel fidèle, importance des variables cohérente avec le métier et exempte de fuite attestent de la validité statistique et de l'interprétabilité des modèles.

Enfin, l'exploitation des prédictions au sein de la logique d'affectation rend l'écart planifié/prédit visible et actionnable avant production et optimise le choix du mode d'affectation sous contraintes d'atelier. L'ensemble confirme les hypothèses H1 à H5 et établit que les modèles, intégrés à l'outil d'aide à la décision, appuient une planification réaliste, supportable par les lignes et conforme aux objectifs fixés.

5.6 Déploiement

La dernière phase de la méthodologie rend le système opérationnel. Les modèles entraînés et la logique d'affectation décrits aux sections précédentes sont encapsulés dans une application web, a été développée avec le framework Flask (Python) [47] et destinée au service planification de l'unité 23.

L'objectif de cette interface est de mettre la capacité de prédiction et le solveur d'affectation à portée du planificateur, sans qu'il ait à manipuler le code ni les fichiers de données : il saisit un ordre de production, l'outil estime la capacité de chaque ligne, propose un plan d'affectation optimisé et restitue des tableaux de bord de suivi. Cette section présente l'application écran par écran et explicite, pour chaque élément, son rôle et son apport à la décision, ainsi que l'enchaînement qui garantit la validité et la cohérence des opérations.

5.6.1 Page d'accueil et entrée dans l'application

La page d'accueil constitue le point d'entrée. Elle affiche un bandeau d'accroche « Optimiser votre chaîne de production », une rangée de quatre vignettes illustrant les fonctions clés, et un bouton central « Commencer » qui déclenche l'ouverture de la session de travail.

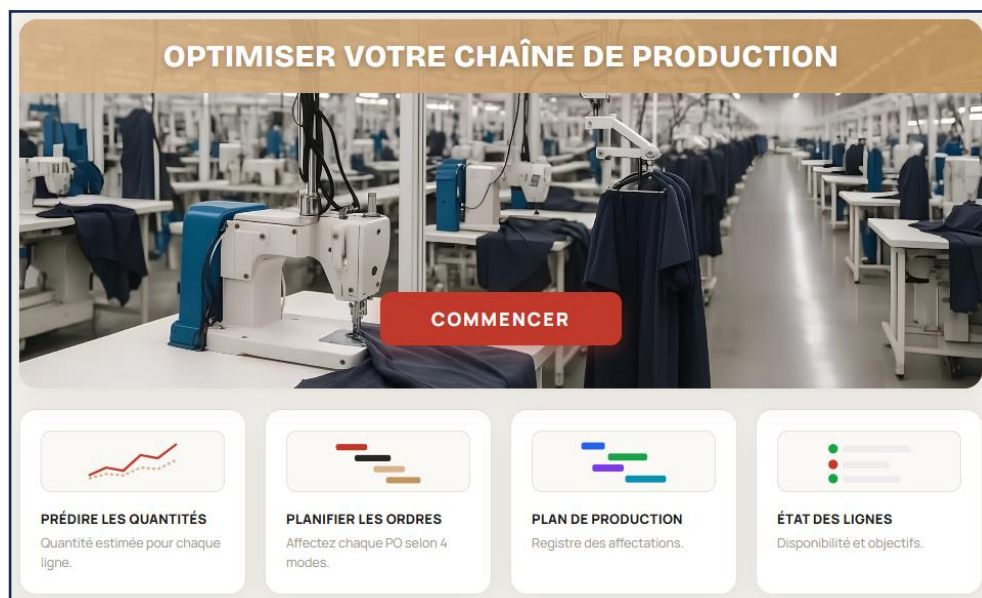


Figure 3.32 : Page d'accueil de l'application

Rôle des éléments

Le bandeau « Optimiser votre chaîne de production » annonce la finalité de l'outil et le situe immédiatement dans son contexte industriel. Les quatre vignettes donnent au visiteur une vue d'ensemble du concept de l'application avant même de l'utiliser ; elles résument la chaîne de valeur de l'outil :

- Prédire les quantités : estimation de la quantité supportable par chaque ligne.
- Planifier les ordres : affectation de chaque PO selon les quatre modes du solveur.
- Plan de production : registre consolidé des affectations retenues.
- État des lignes : disponibilité des lignes et suivi de l'atteinte des objectifs.

Le bouton « Commencer », placé au centre, lance l'opération. Cette page joue donc un double rôle : présenter le périmètre fonctionnel et orienter l'utilisateur vers l'action.

5.6.2 Contrôle d'accès

Un clic sur « Commencer » ouvre une boîte de dialogue « Accès protégé » demandant un mot de passe avant tout usage de l'outil.

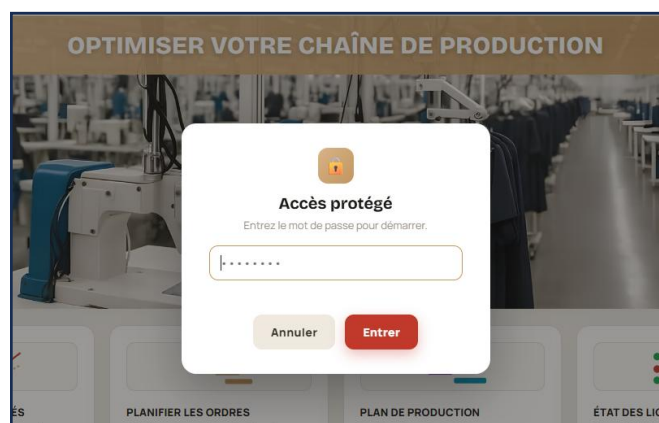


Figure 3.33 : Boîte de dialogue d'authentification par mot de passe

Importance

Ce verrou répond à une exigence de sécurité : les données de production (quantités, modèles, clients, performances des lignes) sont des informations confidentielles de l'entreprise. La protection par mot de passe restreint l'accès au personnel autorisé du service planification et garantit que ni les données historiques ni le plan de production ne soient consultés ou modifiés par un tiers. Après saisie correcte et validation par « Entrer », la fenêtre de travail s'ouvre ; le bouton « Annuler » referme la demande sans ouvrir la session.

5.6.3 Barre de navigation

Une fois la session ouverte, une barre d'icônes en haut de l'écran structure l'application en six espaces, accessibles à tout moment.

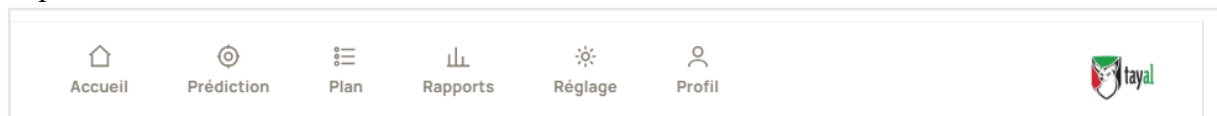


Figure 3.34 : Barre de navigation de l'application

Chaque onglet correspond à une étape de la chaîne de décision :

- Accueil : retour à la page d'accueil.
- Prédiction : saisie d'un ordre de production et lancer la prédiction.
- Plan : consultation du plan de production issu de la prédiction et de l'affectation.
- Rapports : suivi de l'état des lignes et de l'atteinte des objectifs.
- Réglage : choix de la langue et accès au guide d'utilisation.
- Profil : identification du planificateur.
- Logo de l'entreprise TAYAL à droite de la barre.

Cette organisation reproduit le déroulement logique du travail de planification : prédire, puis affecter et planifier, puis suivre et contrôler. L'enchaînement des onglets de gauche à droite traduit ainsi le cycle opérationnel complet.

5.6.4 Onglet « Prédiction »

Cet onglet est le cœur fonctionnel de l'outil. Il propose deux modes de saisie de l'ordre de production « Saisir un PO » manuellement, ou « Importer un Excel » pour traiter un lot d'ordres puis un formulaire de caractérisation de l'ordre et un bouton « Prédire ».

The screenshot shows the 'Prédiction' tab interface. At the top is a navigation bar with icons for Accueil, Prédiction (active), Plan, Rapports, Réglage, and Profil. The main content area is titled '1 - SAISIE DE L'ORDRE DE PRODUCTION'. It contains two buttons: 'Saisir un PO' (highlighted with a red border) and 'Importer un Excel'. Below these are six input fields arranged in two rows of three. The first row contains 'ORDRE DE PRODUCTION (PO)', 'CATÉGORIE' (a dropdown menu showing 'POLO'), and 'STYLE CODE'. The second row contains 'TEMPS DE TRAITEMENT (SDT) - OPTIONNEL', 'NOMBRE D'OPÉRATEURS - OPTIONNEL', and 'QUANTITÉ JOURNALIÈRE PLANIFIÉE' (with the value '600' entered). At the bottom of this section are two buttons: 'PRÉDIRE' (red) and 'RÉINITIALISER' (grey). Below this section is a header for '2 - RÉSULTATS DE PRÉDICTION'.

Figure 3.35 : Onglet Prédiction 1^{er} champs

Le 1^{er} champ du formulaire correspond aux informations dont le modèle a besoin, en distinguant ce qui est connu au moment de la planification de ce qui est optionnel :

- Ordre de production (PO) : identifiant de l'ordre à planifier.
- Catégorie : famille du produit (POLO, BASIC ou CASUAL).
- Style code : référence du modèle d'article.
- Temps de traitement (SDT) : temps standard du modèle.
- Nombre d'opérateurs : effectif envisagé.
- Quantité journalière planifiée : l'objectif fixé pour le PO par jour.

The screenshot shows the '2 - RÉSULTATS DE PRÉDICTION' section. It contains a large white box with the text: 'Aucune prédiction. Saisissez un PO puis cliquez PRÉDIRE.' At the bottom of the interface, there is a footer with the logo of L'Algérienne des Industries Textiles, the text 'L'Algérienne des Industries Textiles · TAYAL SPA · 2026', and a small red icon.

Figure 3.36 : Onglet Prédiction 2^{ème} champs

Au clic sur « Prédire », l'application charge le modèle de la catégorie, reconstitue les paramètres manquants à partir de l'historique (couple ligne-modèle, ligne, modèle, médiane), applique le modèle et ramène la prédiction à l'échelle d'origine. La seconde zone, « Résultats de

prédiction», affiche alors la quantité prédite par ligne, classée par ordre décroissant. Le bouton « Réinitialiser » vide le formulaire.

Cet onglet transforme une estimation auparavant empirique en une prédiction objective et reproductible. Le champ « Quantité journalière planifiée » fixe l'objectif, tandis que la prédiction indique la capacité réaliste de chaque ligne : la confrontation des deux est exactement ce qui alimente l'affectation et le suivi des écarts.

Ce 2^{ème} champ restitue, sous forme de tableau, la quantité prédite de chaque ligne pour l'ordre saisi et met en avant la ligne la plus performante. L'exemple ci-dessous correspond au PO G23016948, POLO, modèle MP135, objectif 600 pièces/jour :

LIGNE	QTÉ PRÉDITE	%	OBJECTIF	SOURCE	OP.
L4	766	127.7%	OUI	ligne+modele	25
L5	740.6	123.4%	OUI	ligne+modele	26
L6	712.1	118.7%	OUI	ligne+modele	26
L2	674.1	112.4%	OUI	ligne+modele	25
L3	660.5	110.1%	OUI	ligne+modele	25
L1	621.9	103.6%	OUI	ligne+modele	23
L17	473.4	78.9%	NON	ligne (modele jamais fait ici)	26

Figure 3.37 : Résultats de prédiction

Chaque ligne indique la quantité prédite (QTÉ PRÉDITE), le taux de couverture de l'objectif (%), le verdict d'atteinte (OBJECTIF : Oui/Non), la SOURCE de la prédiction et l'effectif (OP.). Le bandeau « Meilleure ligne » recommande directement la plus adaptée ici L4, avec 766 pièces prédites, soit 127,7 % de l'objectif. Les lignes sont classées par capacité décroissante : L4, L5, L6, L2, L3, L1 couvrent largement les 600 pièces (Oui), tandis que d'autres lignes restent en deçà (par exemple L17 à 78,9 %,marquées Non).

Le troisième champs, « Planification du PO », transforme la prédiction en plan. Le planificateur choisit l'ordre à planifier, fixe la date de lancement et la date de fin, puis lance le solveur : « Planifier auto » pour une affectation automatique, ou « Planifier manuel » pour composer lui-même la configuration :

3 - PLANIFICATION DU PO

PO À PLANIFIER: G23016948

DATE DE LANCEMENT: 10/06/2026

DATE DE FIN: 11/06/2026

PLANIFIER AUTO PLANIFIER MANUEL

Affectation proposée par le solveur – mode 1 : lignes libres : L4 (2 côtés, 50 postes), L5 (1 côté, 50 postes), L6 (1 côté, 50 postes), L2 (2 côtés, 50 postes), L3 (2 côtés, 50 postes), L1 (2 côtés, 50 postes), L19 (1 côté, 50 postes), L20 (2 côtés, 50 postes)

PO_ID	CATÉGORIE	MODÈLE	LIGNE_ID	CÔTÉ	QTÉ AFFECTÉ/JOUR	DATE DE LANCEMENT	DATE DE FIN
G23016948	POLO	MP135	L1	A	621.9	2026-06-10	2026-06-11

L1 - 1 cote = 621.9 pcs/j - couverture 103.6% - objectif atteint ✓

Lignes occupées : L17 (-), L7 (2026-06-11)

CONFIRMER ET ENREGISTRER

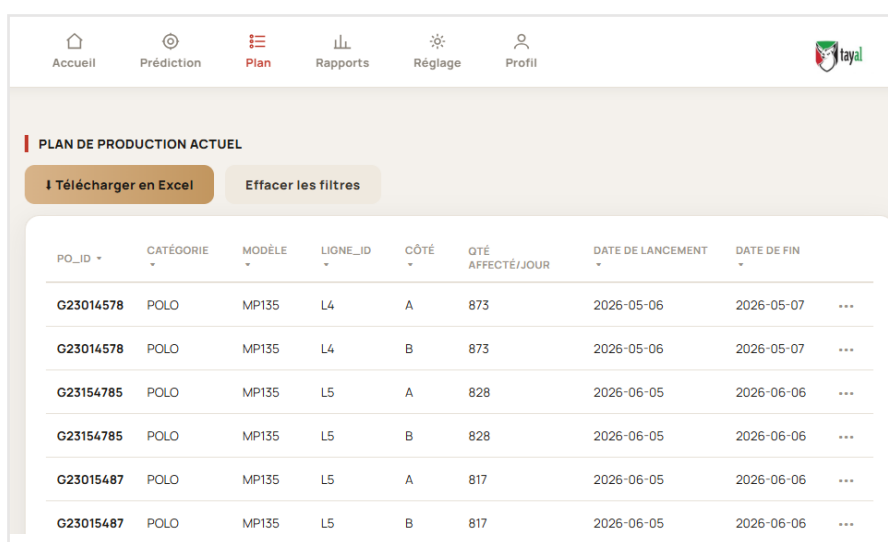
Figure 3.38 : Résultats de prédiction

« objectif atteint » confirme que la configuration couvre l'objectif. L'en-tête rappelle les lignes libres mobilisables et leur capacité en postes, ce qui rend la décision entièrement transparente.

Le bouton « Confirmer et enregistrer » inscrit l'affectation retenue dans le registre du plan de production. Le PO bascule alors dans l'onglet « Plan », où il devient consultable, modifiable et exportable.

5.6.5 Onglet « Plan »

L'onglet « Plan » affiche le plan de production courant, c'est-à-dire le registre des affectations retenues par le solveur. Le plan présenté ici a été établi automatiquement, à partir d'une prédiction puis d'une affectation, sur un échantillon de données réelles de l'entreprise, afin de valider et d'illustrer le fonctionnement de l'application de bout en bout :



The screenshot shows a web application interface with a navigation bar at the top containing icons for Accueil, Prédiction, Plan (highlighted), Rapports, Réglage, and Profil. The main content area is titled 'PLAN DE PRODUCTION ACTUEL' and includes two buttons: 'Télécharger en Excel' and 'Effacer les filtres'. Below this is a table with the following data:

PO_ID	CATÉGORIE	MODÈLE	LIGNE_ID	CÔTÉ	QTÉ AFFECTÉ/JOUR	DATE DE LANCEMENT	DATE DE FIN	
G23014578	POLO	MP135	L4	A	873	2026-05-06	2026-05-07	...
G23014578	POLO	MP135	L4	B	873	2026-05-06	2026-05-07	...
G23154785	POLO	MP135	L5	A	828	2026-06-05	2026-06-06	...
G23154785	POLO	MP135	L5	B	828	2026-06-05	2026-06-06	...
G23015487	POLO	MP135	L5	A	817	2026-06-05	2026-06-06	...
G23015487	POLO	MP135	L5	B	817	2026-06-05	2026-06-06	...

Figure 3.39 : registre des affectations de production

Chaque ligne du tableau décrit une affectation élémentaire :

- PO_ID : l'ordre de production concerné.
- Catégorie et Modèle : la famille et la référence de l'article.
- Ligne_ID et Côté : la ligne et le côté affectés ; un même PO peut apparaître sur plusieurs côtés ou plusieurs lignes, ce qui matérialise les modes d'affectation du solveur.
- Qté affectée/jour : la quantité confiée à cette ligne-côté par jour, issue de la capacité prédite.
- Date de lancement et Date de fin : la fenêtre d'exécution de l'affectation

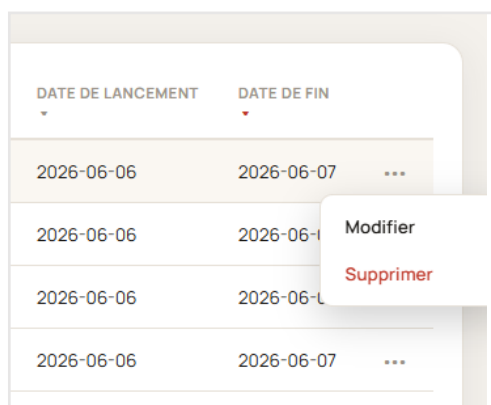


Figure 3.40 : Actions disponibles sur une affectation

Le menu contextuel de chaque ligne « Modifier », « Supprimer » laisse au planificateur la maîtrise finale : l'outil propose une affectation optimisée, mais l'expert conserve la possibilité d'ajuster le plan en fonction de contraintes non modélisées. Le plan, alimenté automatiquement par la prédiction et l'affectation, constitue le livrable opérationnel central de l'application.

5.6.6 Onglet « Rapports »

L'onglet « Rapports » referme la boucle de pilotage : il vérifie que les affectations sont réalisables ; disponibilité des lignes et mesure l'atteinte des objectifs. Il se compose de deux sous-onglets.

État des lignes

Ce sous-onglet liste, par catégorie, la disponibilité de chaque ligne de confection.

LIGNE	STATUT	LIBRE À PARTIR DU
L1	occupee	2026-06-11
L2	occupee	2026-06-06
L3	occupee	2026-06-07
L4	occupee	2026-05-07
L5	occupee	2026-06-07
L6	libre	-
L7	occupee	2026-06-11

Figure 3.41 : Rapport « État des lignes »

Chaque ligne porte un statut « libre » ou « occupée », et, lorsqu'elle est occupée, une date de disponibilité (« Libre à partir du »). Cette information est essentielle au solveur : c'est elle qui détermine les positions et postes réellement disponibles à la date de lancement, et donc les configurations d'affectation possibles.

Objectifs

Ce sous-onglet évalue dans quelle mesure les objectifs des ordres sont couverts. Il associe un graphique de synthèse par catégorie et un tableau détaillé par PO.

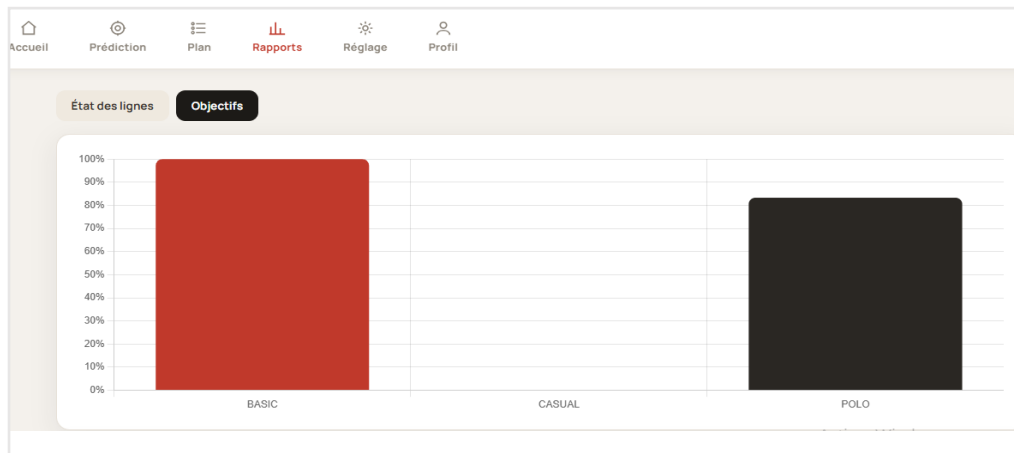


Figure 3.42 : Taux d'atteinte des objectifs par catégorie

PO	CATÉGORIE	OBJECTIF/J	AFFECTÉ/J	COUVERTURE	ATTEINT
G23014578	POLO	1600	1746	109.1%	OUI
G23014587	BASIC	1500	1566	104.4%	OUI
G23015487	POLO	1600	2259	141.2%	OUI
G23016911	POLO	1500	1514	100.9%	OUI
G23016948	POLO	600	621.9	103.6%	OUI

Figure 3.43 : Détail par ordre de production sur l'objectif

Le tableau confronte, pour chaque PO, l'objectif journalier « Objectif/J », la quantité affectée par le plan « Affecté/J », le taux de couverture qui en résulte et la conclusion « Atteint » Oui/Non. Un ordre est jugé atteint lorsque la quantité affectée couvre au moins l'objectif couverture $\geq 100\%$.

Ce rapport est l'indicateur de pertinence opérationnelle de l'outil. Il rend l'écart entre planifié et réalisable lisible et chiffré avant production, et signale immédiatement les ordres à risque, sur lesquels le planificateur peut agir.

5.6.7 Onglet « Réglage »

Cet onglet regroupe les paramètres généraux de l'application :

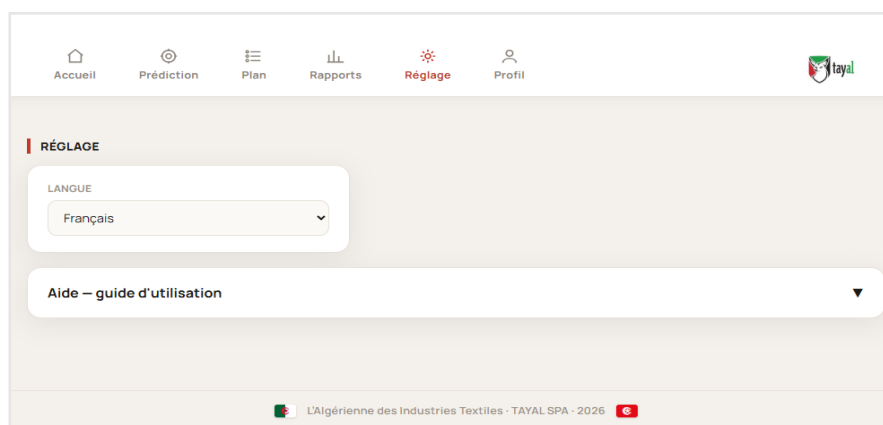


Figure 3.44 : Onglet Réglage

Le sélecteur de « Langue » adapte l'interface à l'utilisateur, ce qui favorise l'adoption de l'outil par l'ensemble du service. La section « Aide - guide d'utilisation » fournit une documentation intégrée : elle participe à la transmissibilité de la connaissance, en rendant l'outil utilisable aussi bien par les planificateurs expérimentés que par les nouveaux ingénieurs

5.6.8 Onglet « Profil »

L'onglet « Profil » gère l'identité du planificateur à travers un « Compte planificateur » :

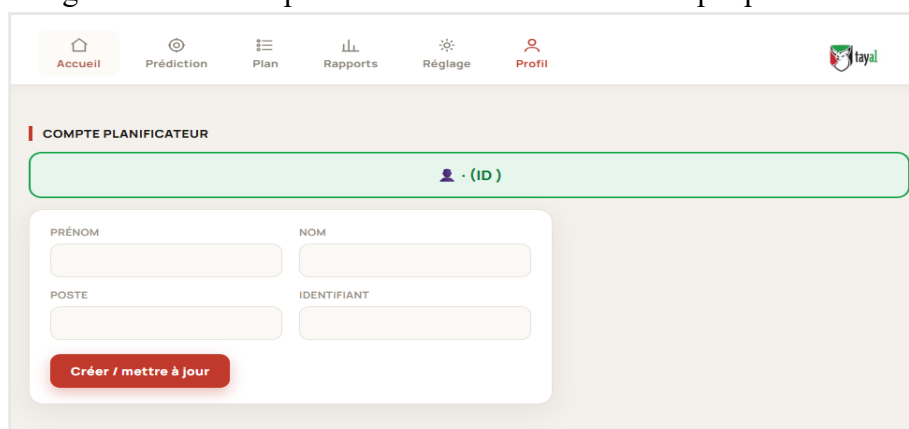


Figure 3.45 : Onglet Profil

Réunissant en un seul outil la prédiction, l'affectation et le suivi, l'application vise à faciliter la prise de décision et à améliorer la planification de l'unité 23. Pour les ingénieurs de planification et de production de TAYAL, sa valeur ajoutée est concrète : estimer la production réelle de chaque ligne pour chaque modèle, proposer une affectation optimisée sous contraintes d'atelier et rendre l'écart entre planifié et réalisable visible avant production. Elle substitue ainsi à l'estimation empirique une démarche objective, rapide et transmissible, qui fiabilise les engagements et rend l'efficacité des lignes lisible, un gain de temps et de maîtrise directement exploitable au quotidien.

6. Conclusion

Ce chapitre a déroulé une démarche complète, du problème posé jusqu'à un outil opérationnel. Face à une planification fondée sur l'expérience individuelle, source d'écarts récurrents entre quantités planifiées et produites et d'une efficience mal maîtrisée, le travail a visé à prédire de façon fiable la production par ligne pour appuyer une planification réaliste, selon les hypothèses 1 à 5.

Après une préparation des données éliminant les fuites d'information, la modélisation a montré l'inadéquation des réseaux de neurones, avant de se réorienter vers le gradient boosting XGBoost, tournant décisif de la démarche. L'évaluation a validé ce choix en valeur mesurable: modèles fiables sur POLO et BASIC et exploitable sur CASUAL, avec une réduction de 43 à 64 % de la MAE et de la RMSE par rapport aux réseaux de neurones. Les diagnostics complémentaires ont confirmé la validité statistique et l'interprétabilité des modèles, la fragilité de CASUAL étant expliquée et inscrite dans une logique d'amélioration continue.

Enfin, la prédiction a été intégrée à un solveur d'affectation et à une application web Flask sécurisée, qui reproduit le cycle de décision du planificateur et rend l'écart planifié/réalisable visible avant production. L'ensemble confirme les hypothèses posées et répond à la problématique : une planification réaliste, supportable par les lignes et conforme aux objectifs, fondée non plus sur l'intuition mais sur des estimations fiables, reproductibles et transmissibles.

Conclusion générale

Au terme de ce travail, l'objectif initial est atteint : démontrer que l'intelligence artificielle peut devenir un acteur concret de la planification de la production et apporter une valeur tangible à l'entreprise. Notre démarche a consisté à transformer une décision longtemps fondée sur l'expérience en une décision objective, appuyée sur l'exploitation des données historiques de production. Le raisonnement suivi a été progressif : comprendre le besoin, préparer rigoureusement les données, comparer plusieurs approches de modélisation, puis retenir celle qui correspond le mieux à la nature du problème, avant de l'intégrer dans un outil exploitable.

Les résultats obtenus confirment la pertinence de cette démarche. Le modèle retenu fournit une estimation fiable et très proche de la quantité réellement produite. Il ne prétend pas à une précision parfaite, mais l'écart qui subsiste constitue lui-même une information utile : il offre au décideur, dès le lancement, une vision claire de la quantité réellement soutenable par les lignes. De plus, comme les ordres portent sur des quantités élevées, cet écart reste proportionnellement faible, voire négligeable à l'échelle d'un ordre de production.

Ces résultats valident l'hypothèse centrale de ce travail : un modèle d'apprentissage automatique, correctement entraîné sur les données historiques et capable de capturer la relation entre les variables explicatives et la quantité réellement produite, fournit des prédictions suffisamment fiables et satisfaisantes pour servir de base à la planification quotidienne des lignes de confection. L'hypothèse est donc confirmée.

Ce travail ouvre enfin des perspectives concrètes. À court terme, l'enrichissement de l'historique et l'ajout de nouvelles variables. À moyen terme, l'intégration des départements méthodologie, ainsi qu'un suivi aval des quantités réellement produites pour évaluer et valider le système en continu. Ce suivi alimente une véritable boucle d'amélioration continue : au fil du temps, les quantités planifiées s'alignent sur la performance prédite, les données réinjectées affinent les modèles, et l'efficacité comme la performance des lignes progressent. L'entreprise en tire un gain durable et une valeur ajoutée croissante. Étendue à d'autres unités, cette approche illustre l'impact significatif de l'intelligence artificielle sur la performance opérationnelle et la compétitivité des entreprises industrielles.

Bibliographie

- [1] S. Mohsen, A. Elkaseer, and S. G. Scholz, “Industry 4.0-oriented deep learning models for human activity recognition,” *IEEE Access*, vol. 9, pp. 150508–150521, 2021.
- [2] R. R. Lummus and R. J. Vokurka, “Defining supply chain management: a historical perspective and practical guidelines,” *Industrial Management & Data Systems*, vol. 99, no. 1, pp. 11–17, 1999.
- [3] X.-M. Yuan and A. Xue, “Supply chain 4.0: New generation of supply chain management,” *Logistics*, vol. 7, p. 9, 2023.
- [4] Association Canadienne de Gestion des Produits et des Stocks (ACGPS), *Définition et principes de la gestion des stocks*. Canada : ACGPS, 1993.
- [5] S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. Upper Saddle River, NJ: Pearson, 2009.
- [6] G. Shmueli, “To explain or to predict?,” *Statistical Science*, vol. 25, no. 3, pp. 289–310, 2010.
- [8] W. J. Stevenson, *Operations Management*, 13th ed. McGraw-Hill Education, 2018.
- [11] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The M4 Competition: 100,000 time series and 61 forecasting methods,” *International Journal of Forecasting*, vol. 36, no. 1, pp. 54–74, 2020.
- [12] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*, 5th ed. Hoboken, NJ: Wiley, 2015.
- [13] S. Thomassey, “Sales forecasting in clothing industry: The key success factor of the supply chain management,” *International Journal of Production Economics*, vol. 128, no. 2, pp. 470–483, 2010.
- [14] H. Kagermann, W. Wahlster, and J. Helbig, *Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0*. Frankfurt: Acatech, 2013.
- [15] H. Lasi, P. Fettke, H. G. Kemper, T. Feld, and M. Hoffmann, “Industry 4.0,” *Business & Information Systems Engineering*, vol. 6, no. 4, pp. 239–242, 2014.
- [16] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson, 2021.
- [19] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.
- [20] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. New York: Springer, 2009.
- [21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2016.
- [22] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [23] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

- [24] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [25] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [26] K. Cho et al., “Learning phrase representations using RNN encoder–decoder for statistical machine translation,” in *Proc. EMNLP*, 2014, pp. 1724–1734.
- [27] A. Vaswani et al., “Attention is all you need,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
- [28] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. Int. Conf. Learning Representations (ICLR)*, 2015.
- [29] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [30] L. Breiman, “Bagging predictors,” *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
- [31] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [32] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.
- [33] J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
- [34] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
- [35] G. Ke et al., “LightGBM: A highly efficient gradient boosting decision tree,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
- [36] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, “CatBoost: Unbiased boosting with categorical features,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.
- [37] D. Micci-Barreca, “A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems,” *ACM SIGKDD Explorations Newsletter*, vol. 3, no. 1, pp. 27–32, 2001.
- [38] S. Kaufman, S. Rosset, C. Perlich, and O. Stitelman, “Leakage in data mining: Formulation, detection, and avoidance,” *ACM Transactions on Knowledge Discovery from Data*, vol. 6, no. 4, pp. 1–21, 2012.
- [39] S. García, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining*. Cham: Springer, 2015.
- [40] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
- [41] L. Grinsztajn, E. Oyallon, and G. Varoquaux, “Why do tree-based models still outperform deep learning on tabular data?,” in *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2022.

- [42] R. Shwartz-Ziv and A. Armon, "Tabular data: Deep learning is not all you need," *Information Fusion*, vol. 81, pp. 84–90, 2022.
- [43] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [44] W. McKinney, "Data structures for statistical computing in Python," in *Proc. 9th Python in Science Conf. (SciPy)*, 2010, pp. 51–56.
- [45] C. R. Harris et al., "Array programming with NumPy," *Nature*, vol. 585, pp. 357–362, 2020.
- [46] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.
- [47] Pallets, *Flask Documentation*. [En ligne]. Disponible : <https://flask.palletsprojects.com/en/stable/> (consulté le 12 février 2026).
- [48] XGBoost Developers, *XGBoost Documentation*. [En ligne]. Disponible : https://xgboost.readthedocs.io/en/release_3.2.0/ (consulté le 3 mars 2026).
(consulté le 3 mars 2026).
- [49] Scikit-learn Developers, "Cross-validation: evaluating estimator performance." [En ligne]. Disponible : <https://www.jmlr.org/tmlr/> (consulté le 19 mai 2026).
- [50] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943.
- [51] TAYAL S.P.A. (2024). Site officiel de l'entreprise TAYAL. Disponible sur : <https://www.tayal.dz>. [Consulté le 10 février 2025]
- [7] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 3rd ed. OTexts, 2021. [En ligne]. Disponible : <https://otexts.com/fpp3/> (consulté le 18 mars 2025).
- [9] XGBoost Developers, *XGBoost Documentation*. [En ligne]. Disponible : <https://xgboost.readthedocs.io/> (consulté le 3 mars 2026).
- [10] Scikit-learn Developers, "Cross-validation: evaluating." [En ligne]. Disponible : https://scikit-learn.org/stable/modules/cross_validation.html (consulté le 11 Juin 2026).
- [17] OECD, *Artificial Intelligence in Society*. Paris: OECD Publishing, 2019. [En ligne]. Disponible <https://doi.org/10.1787/eedfee77-en> (consulté le 5 avril 2025).
- [18] National Institute of Standards and Technology, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. Gaithersburg, MD: NIST, 2023. [En ligne]. Disponible : <https://www.nist.gov/itl/ai-risk-management-framework> (consulté le 22 avril 2025).