الجمهورية الجزائرية الديمقراطية الشعبية

Republique Algerienne Democratique et Populaire

وزارة التعليم العالي و البحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة العربي التبسي -تبسة-

Université Larbi Tebessi – Tébessa –

Faculté des sciences et technologies

Département de Génie Civil

MÉMOIRE DE MASTER

Présenté en vue de l'obtention du **Diplôme de Master Académique**

Domaine : Génie Civil

Spécialité : Structures

Par : AIMEN Amira

Sujet :

Modèle hybride pour la prédiction de la résistance au cisaillement des connecteurs en U dans du béton ordinaire et à haute résistance

Présenté et soutenu publiquement le **08/06/2026**, devant le jury composé de :

Mr. BOUTAGOUGA Djamel	Professeur	Président
Mr. BOURSAS Farid	MCA	Rapporteur
Mr. LABED Abdrrahim	MCA	Examinateur 1

Année universitaire : 2025/2026



بسم الله الرحمن الرحيم

Remerciements

Au nom de Dieu, le Clément, le Miséricordieux, et louange à Dieu, Seigneur de l'univers, dont les bienfaits permettent l'accomplissement de toutes les œuvres, et qui m'a accordé la réussite et l'assistance nécessaires pour mener à bien ce travail dans des conditions scientifiques et méthodologiques appropriées.

Je tiens à exprimer mes sincères remerciements à Monsieur **Boursas Farid** pour son encadrement rigoureux, son suivi constant ainsi que pour ses orientations méthodologiques pertinentes. Ses conseils et ses remarques ont contribué de manière significative à l'amélioration de ce travail. J'ai également apprécié son sérieux académique et son engagement tout au long de ce mémoire.

Je souhaite également adresser mes remerciements à l'ensemble des enseignants du département de génie civil, pour les efforts scientifiques et pédagogiques fournis durant tout le parcours universitaire, ainsi que pour la qualité de la formation dispensée, qui a contribué au développement de mes connaissances et au renforcement de mes compétences en recherche.

Je ne saurais, en outre, exprimer toute ma reconnaissance envers ma famille, qui n'a ménagé aucun effort pour me soutenir et m'accompagner tout au long de mon parcours académique. Elle a constitué un véritable pilier de soutien, notamment à travers ses encouragements constants et ses nombreux sacrifices.

Enfin, j'adresse mes remerciements à toutes les personnes ayant contribué, de près ou de loin, à la réalisation de ce travail, que ce soit par un conseil, un soutien ou une aide, et dont l'apport a été précieux dans son aboutissement.

Dédicace

Avec l'expression de mon profond respect et de ma sincère reconnaissance, je dédie ce travail de fin d'études et ma joie de réussite :

À la fatigue et aux efforts qui m'ont appris que la réussite ne s'obtient qu'avec la patience, la persévérance et la détermination.

À la lumière qui a éclairé mon chemin et à la lampe dont la clarté ne s'éteint jamais.

À celui qui m'a offert le plus précieux des dons et dont j'ai tiré ma force et mon estime de soi :

Mon cher père AHMED

À celle qui a fait du paradis un endroit sous ses pieds et qui a rendu les difficultés plus faciles grâce à ses prières, la grande femme qui a toujours souhaité voir le bonheur dans mes yeux :

Ma merveilleuse mère HAFIDA

À celle qui a renforcé mon bras et a été une source inépuisable de soutien et de tendresse, le confort de mes yeux :

Ma douce sœur AMAL

À mes piliers et mes soutiens, qui m'ont apporté amour et force :

Mes chers frères HASSAN & HAMDEN

À la raison de mon bonheur et au sucre de ma vie.

Ce n'est pas la fin de mon parcours, mais seulement le début de mes réussites.

Louange à Allah pour tout ce qu'Il m'a accordé.
Merci, amour et gratitude au commencement et à la fin.
Et notre dernière invocation est : Louange à Allah, Seigneur de l'univers.

Eng. Amira

Résumé

Les structures mixtes acier-béton sont largement utilisées en génie civil en raison de leurs performances mécaniques optimisées. La résistance au cisaillement des connecteurs en U (Channel Connectors), qui assurent la liaison entre la poutre en acier et la dalle en béton, constitue un paramètre clé dans le dimensionnement de ces structures. Les modèles analytiques traditionnels, notamment ceux proposés par l'Eurocode 4, présentent des limites en termes de précision et de capacité de généralisation. Ce travail propose une approche hybride d'apprentissage automatique pour la prédiction de la résistance au cisaillement des connecteurs en U ancrés dans du béton ordinaire et à haute résistance. Une base de données de 618 essais expérimentaux de cisaillement par poussée, compilés à partir de la littérature scientifique, a servi de support à l'élaboration des modèles. Cinq algorithmes de régression ont été sélectionnés et optimisés via GridSearchCV avec une validation croisée à 5 plis : forêt aléatoire (Random Forest), XGBoost, régression par vecteurs de support (SVR), perceptron multicouche (MLP) et k-plus proches voisins (k-NN). Ces modèles ont ensuite été combinés au sein d'un méta-modèle de stacking basé sur la régression Ridge, le tout implémenté en Python via Jupyter Notebook. Le modèle de stacking a obtenu les meilleures performances avec $R^2 = 0.8974$, $MAE = 20.44\,\text{kN}$ et $RMSE = 50.55\,\text{kN}$ sur le jeu de test. La validation croisée et l'analyse des résidus ont confirmé la stabilité et la fiabilité des modèles développés. Ce travail démontre l'efficacité des techniques de Machine Learning comme alternative rapide et précise aux formules empiriques pour le dimensionnement des structures mixtes.

Mots-clés : connecteurs de cisaillement, canal en U, apprentissage automatique, stacking, RF, XGBoost, structures mixtes, prédiction de résistance au cisaillement.

Abstract

Steel-concrete composite structures are widely used in civil engineering due to their optimized mechanical performance. The shear resistance of U-connectors (Channel Connectors), which ensure the connection between the steel beam and the concrete slab, constitutes a key parameter in the design of these structures. Traditional analytical models, particularly those proposed by Eurocode 4, have limitations in terms of accuracy and generalization capacity. This work proposes a hybrid machine learning approach for predicting the shear strength of U-shaped connectors anchored in ordinary and high-strength concrete. A database of 618 experimental push-out tests, compiled from the scientific literature, served as the basis for developing the models. Five regression algorithms were selected and optimized using GridSearchCV with 5-fold cross-validation : Random Forest, XGBoost, Support Vector Regression (SVR), Multi-Layer Perceptron (MLP), and k-Nearest Neighbors (k-NN). These models were then combined into a stacking meta-model based on Ridge regression, all implemented in Python via Jupyter Notebook. The stacking model achieved the best performance with $R^2 = 0.8974$, $MAE = 20.44$ kN, and $RMSE = 50.55$ kN on the test set. Cross-validation and residual analysis confirmed the stability and reliability of the developed models. This work demonstrates the effectiveness of Machine Learning techniques as a fast and accurate alternative to empirical formulas for the design of composite structures.

Keywords : shear connectors, U-channel, machine learning, stacking, Random Forest, XGBoost, composite structures, shear strength prediction.

ملخص

تُستخدم المنشآت المركبة من الفولاذ والخرسانة على نطاق واسع في الهندسة المدنية بفضل خصائصها الميكانيكية المحسّنة. وتُعد مقاومة القص للموصلات على شكل U(connecteurs de canal) التي تضمن الربط بين الجائز الفولاذي والبلاطة الخرسانية، عاملاً أساسياً في تصميم هذه المنشآت. غير أن النماذج التحليلية التقليدية، وخاصة تلك المقترحة في معيار Eurocode 4 , تُظهر بعض القيود من حيث الدقة والقدرة على التعميم. يقترح هذا العمل منهجية هجينة تعتمد على التعلم الآلي للتنبؤ بمقاومة القص لموصلات U المثبتة في الخرسانة العادية وعالية المقاومة. وقد تم الاعتماد على قاعدة بيانات تضم 618 اختبار دفع تجريبياً (tests de poussée) جُمعت من الأدبيات العلمية لتطوير النماذج. تم اختيار 5 خوارزميات انحدار وتحسينها باستخدام تقنية GridSearchCV مع التحقق المتقاطع بخمسة طيات (Validation croisée à 5 plis) وهي : الغابة العشوائية (Forêt Aléatoire) , وتعزيز التدرج الشديد (XGBoost), وانحدار متجهات الدعم (SVR), والشبكة العصبية متعددة الطبقات (MLP), وخوارزمية أقرب الجيران (k-NN). ثم تم دمج هذه النماذج ضمن نموذج تجميعي (Stacking) يعتمد على نموذج Ridge كنموذج فوقي (Meta-model), وتم تنفيذ ذلك باستخدام لغة Python عبر بيئة Jupyter Notebook. حقق نموذج التجميع (Stacking) أفضل النتائج، حيث سجل معامل التحديد $R^2 = 0.8974$, ومتوسط الخطأ المطلق $MAE = 20.44, \text{kN}$، وجذر متوسط مربع الخطأ $RMSE = 50.55, \text{kN}$ على مجموعة الاختبار. كما أكدت نتائج التحقق المتقاطع وتحليل البواقي استقرار وموثوقية النماذج المطورة. وتبرز هذه الدراسة فعالية تقنيات التعلم الآلي كبديل سريع ودقيق للصيغ التجريبية التقليدية في تصميم المنشآت المركبة.

الكلمات المفتاحية : موصلات القص, قناة على شكل U, التعلم الآلي, التجميع, الغابة العشوائية, تعزيز التدرج الشديد, المنشآت المركبة, التنبؤ بمقاومة القص.

Table des matières

Remerciements **I**

Dédicace **II**

Résumé **III**

Abstract **IV**

ملخص **V**

Liste des figures **XII**

Liste des tableaux **XIII**

Liste des abréviations **XIV**

Introduction Générale **XVII**

Objectifs de l'étude **XVIII**

Structure du mémoire **XVIII**

Chapitre 01 **1**

1 Aperçu Général de L'apprentissage Automatique **2**

1.1 Introduction 2

1.2 Definition 3

1.3 Types d'Apprentissage Automatique 3

1.3.1 Apprentissage supervisé 4

1.3.2 Apprentissage non-supervisé 6

1.3.3 Apprentissage par renforcement 7

1.4 Cycle de vie d'une tâche d'apprentissage automatique 8
1.5 Les algorithmes d'apprentissage automatique 9
1.5.1 Algorithmes d'apprentissage supervisé 9
1.5.2 Algorithmes d'Apprentissage non-supervisé 18
1.5.3 Algorithmes d'Apprentissage par renforcement 19
1.6 Conclusion 20

Chapitre 02 **22**

2 Revue de littérature et élaboration du dataset **23**
2.1 Introduction aux structures mixtes acier–béton 23
2.1.1 Principe de fonctionnement 23
2.1.2 Rôle des connecteurs de cisaillement 23
2.2 Les connecteurs de cisaillement en U (Channel Connectors) . . . 24
2.2.1 Géométrie et caractéristiques 24
2.2.2 Mécanisme de transfert de cisaillement 25
2.3 Revue des travaux de recherche 26
2.3.1 Études expérimentales 26
2.3.2 Études numériques(FEM) 37
2.3.3 Modèles analytiques existants 41
2.4 Paramètres influençant la résistance au cisaillement 42
2.5 Limites des modèles existants 43
2.6 Construction du dataset 43
2.6.1 Sources des données 44
2.6.2 Description des variables 45
2.6.3 Organisation du dataset 46
2.6.4 Critères d'inclusion et d'exclusion 46
2.7 Conclusion 47

Chapitre 03 **48**

3 Programmation du ML en python **49**
3.1 Introduction au notebook et environnement de travail 49

3.2 Chargement et exploration des données 50
3.2.1 Importation du dataset 50
3.2.2 Vérification de la structure des données 51
3.2.3 Analyse des dimensions et variables 51
3.2.4 Vérification des types de données 52
3.2.5 Détection des valeurs manquantes 52
3.2.6 Analyse statistique descriptive initiale 52
3.2.7 Vérification de la cohérence physique des variables 52
3.3 Analyse exploratoire des données (EDA) — 4 visualisations principales 53
3.3.1 Matrice de Corrélation (Heatmap) 53
3.3.2 Distribution Statistique des Variables (Histogrammes) . . . 53
3.3.3 Analyse des Relations Variables–Résistance (Scatter plots) 53
3.3.4 Détection des Valeurs Aberrantes (Boxplots) 54
3.3.5 Apport de l'EDA dans la Compréhension Physique du Matériau 54
3.4 Augmentation des Données et Séparation Train/Test 54
3.4.1 Justification Scientifique de l'Augmentation des Données . 54
3.4.2 Méthodes Utilisées pour Enrichir le Dataset 55
3.4.3 Prétraitement et Normalisation 55
3.4.4 Définition des Variables d'Entrée (X) et de Sortie (y) . . . 56
3.4.5 Division Train/Test 56
3.4.6 Importance du Split pour la Généralisation 56
3.4.7 Métriques d'évaluation des modèles (R^2, MAE, RMSE, MSE) 57
3.5 Entraînement des modèles et optimisation des hyperparamètres (GridSearchCV) 58
3.5.1 Présentation des Modèles Sélectionnés 58
3.5.2 Régression par Forêt Aléatoire 58
3.5.3 Régresseur XGBoost 59
3.5.4 Régression par vecteurs de support (SVR) 59

3.5.5 Perceptron Multi-Couches (MLP) 60
3.5.6 Régression par K-Plus Proches Voisins (k-NN) 60
3.5.7 Principe de GridSearchCV 61
3.5.8 Recherche des Hyperparamètres Optimaux 62
3.5.9 Validation Croisée Intégrée 62
3.6 Construction du Modèle hybride 63
3.6.1 Concept d'Apprentissage en Ensemble 63
3.6.2 Sélection des Modèles de Base 63
3.6.3 Principe du Stacking Regressor 64
3.6.4 Choix de Ridge comme Méta-Apprenant 65
3.6.5 Architecture Finale du Modèle Hybride 65
3.6.6 Avantages en Robustesse et Précision 66
3.7 Conclusion 66

Chapitre 04 **67**

4 Résultats et discussion **68**
4.1 Introduction 68
4.1.1 Interprétation de la matrice de corrélation 68
4.1.2 Analyse des distributions statistiques et courbes KDE (Kernel Density Estimation) 70
4.1.3 Relations entre paramètres d'entrée et résistance 72
4.1.4 Détection et impact des valeurs aberrantes 75
4.2 Performances des modèles — métriques et comparaison 76
4.2.1 Critères d'évaluation (R^2, RMSE, MAE, MSE) 77
4.2.2 Résultats individuels des modèles de base 77
4.2.3 Performance du modèle Stacking 77
4.2.4 Comparaison quantitative 77
4.2.5 Classement des modèles 80
4.2.6 Discussion scientifique des écarts de performance 81
4.3 Analyse des résidus et erreurs 82
4.3.1 Distribution des résidus 84

4.3.2 Valeurs prédites vs valeurs réelles 84
4.3.3 Erreurs absolues et relatives 86
4.3.4 Détection des biais 87
4.3.5 Robustesse du modèle 87
4.4 Validation croisée et détection d'overfitting 88
4.4.1 Résultats de la validation croisée 88
4.4.2 Comparaison entraînement/test 89
4.4.3 Détection du surapprentissage 90
4.4.4 Analyse de la généralisation 91
4.4.5 Fiabilité pour applications réelles 91
4.5 Importance des variables d'entrée (Feature Importance) 91
4.6 Interprétation du méta-modèle Stacking 94
4.6.1 Contribution des modèles de base 95
4.6.2 Poids du méta-modèle Ridge 95
4.6.3 Synergie algorithmique 96
4.6.4 Justification de la supériorité prédictive 96
4.6.5 Apport méthodologique à la recherche 96
4.7 Synthèse générale et recommandations 97
4.7.1 Résumé des résultats majeurs 97
4.7.2 Implications pratiques pour la conception des structures mixtes acier-béton 97
4.7.3 Recommandations méthodologiques 98
4.7.4 Limites de l'étude 98
4.7.5 Perspectives futures (Deep Learning, optimisation multi-objectifs, nouveaux datasets) 99
4.8 Développement d'une Application de Prédiction de la Résistance au Cisaillement 100
4.9 Conclusion 102

Conclusion Générale **103**

Bibliographie **105**

Liste des figures

1.1 Les grandes classes d'apprentissage automatique 3
1.2 Ensemble de données étiquetées Chaque image d'entrée est associée avec la prédiction voulue 4
1.3 Ensemble de données non-étiquetées Les images d'entrées, extraites de la base de données MNIST Fashion, ne sont pas associées à une sortie cible 6
1.4 Exemple d'apprentissage non supervisé 7
1.5 Schéma descriptive de l'apprentissage par renforcement 7
1.6 Flux d'apprentissage automatique 8
1.7 Étapes pour l'algorithme KNN 10
1.8 Fonctionnement de la machine à vecteurs de support 11
1.9 Pseudo-code pour la machine à vecteurs de support 12
1.10 ε-insensitive regression band. The solid blue line represents the estimated regression curve $f(x)$. 13
1.11 Régions de décision des perceptrons multicouches 14
1.12 Exemple d'arbre de décision 15
1.13 Schéma du diagramme de l'algorithme de forêt aléatoire 16
1.14 Étapes de l'algorithme K-moyennes 18

2.1 Échantillon d'essai de poussée sur des connecteurs de cisaillement en U 25
2.2 essai de poussée (échantillon) 26
2.3 Détails de l'échantillon 27
2.4 Détails de l'échantillon et configuration du test 27
2.5 Détails de specimens de tests 28
2.6 Échantillon de poussée avec connecteur en forme de I 30

2.7 Dispositif d'essai de poussée 31
2.8 Détails de l'échantillon de test de poussée 32
2.9 Présentation schématique des connecteurs à angles inclinés avec un degré d'inclinaison de : a 122,5° ; b 135° 33
2.10 Échantillons de poussée 34
2.11 Géométrie et détails des échantillons de test 35
2.12 Dimensions géométriques des échantillons, en mm 36
2.13 Configuration structurelle et détails de chargement des échantillons (mm) 37
2.14 Modèle géométrique de test de poussée 38

4.1 Heatmap corrélation Pearson 69
4.2 Distributions des variables(a) 71
4.3 Distributions des variables(b) 72
4.4 Scatter plots paramètres vs résistance(a) 73
4.5 Scatter plots paramètres vs résistance(b) 74
4.6 Box plots standardisés(a) 75
4.7 Box plots standardisés(b) 76
4.8 Comparaison R^2, MAE en barres horizontales 79
4.9 Comparaison MSE,RMSE en barres horizontales 79
4.10 Diagramme de Taylor 81
4.11 Graphes de résidus(a) 82
4.12 Graphes de résidus(b) 83
4.13 Valeurs réelles vs prédites 85
4.14 Distribution des erreurs en violin plots 87
4.15 Box plots CV 5-Fold R^2 89
4.16 R^2 Train vs Test par modèle 89
4.17 Importance des variables selon le modèle RF 92
4.18 Importance des variables selon les modèles RF et XGBoost 92
4.19 Coefficients Ridge du méta-modèle 94
4.20 Interface de l'application avant prédiction 101
4.21 Résultat de la prédiction 102

Liste des tableaux

1.1 Exemple d'un ensemble de données d'apprentissage supervisé (Régression) 5
1.2 Exemple d'un ensemble de données d'apprentissage supervisé (Classification) 5

2.1 Propriétés géométriques des connecteurs en U 24
2.2 Résistance des connecteurs sous chargement monotone et cyclique et taux de dégradation associé 29
2.3 Propriétés géométriques des sections en acier utilisées dans l'étude 34
2.4 Récapitulatif des 23 sources du dataset (618 observations) 44
2.5 Description et statistiques descriptives des 13 variables du dataset 45
2.6 Critères d'inclusion et d'exclusion appliqués à la construction du dataset 46

3.1 Description des variables du dataset 51
3.2 Classification des variables du jeu de données 51
3.3 Types des variables 52
3.4 Statistiques descriptives des variables du jeu de données 52
3.5 Architecture finale du modèle hybride par stacking 65

4.1 Corrélation de Pearson entre les paramètres d'entrée et la résistance 69
4.2 Performances comparatives de tous les modèles sur les jeux d'entraînement et de test 77
4.3 Résultats de la validation croisée 5-Fold 88
4.4 Importance des variables d'entrée selon le modèle XGBoost . . . 93
4.5 Coefficients Ridge du méta-modèle Stacking 96

Liste des abréviations

- AA : Apprentissage automatique (Machine Learning)
- AB : Renforcement adaptatif (Adaptive Boosting)
- AISC : Institut Américain de la Construction en Acier (American Institute of Steel Construction)
- ANFIS : Système d'inférence neuro-flou adaptatif (Adaptive Neuro-Fuzzy Inference System)
- ANSYS : Logiciel d'analyse numérique (ANSYS)
- CART : Arbres de classification et de régression (Classification And Regression Trees)
- CH6 : Connecteur de type Channel n°6 (Channel Connector Type 6)
- CNN 1D : Réseau de neurones convolutif unidimensionnel (One-Dimensional Convolutional Neural Network)
- CSA : Association canadienne de normalisation (Canadian Standards Association)
- CSC : Connecteur de cisaillement de type Channel (Channel Shear Connector)
- EC : Connecteur en C amélioré (Enhanced C-Channel)
- EC4 : Eurocode 4
- ECC : Béton composite cimentaire ductile (Engineered Cementitious Composite)

- EDA : Analyse exploratoire des données (Exploratory Data Analysis)
- EF : Éléments finis
- ELM : Machine à apprentissage extrême (Extreme Learning Machine)
- FEM : Méthode des éléments finis (Finite Element Method)
- GBM : Machine de Gradient Boosting (Gradient Boosting Machine)
- HSC : Béton à haute résistance (High Strength Concrete)
- IA : Intelligence artificielle
- IQR : Intervalle interquartile (Interquartile Range)
- k-NN : k plus proches voisins (k-Nearest Neighbors)
- LR : Régression linéaire (Linear Regression)
- MAE : Erreur absolue moyenne (Mean Absolute Error)
- MLP : Perceptron multicouche (Multilayer Perceptron)
- MPG : Méthodes de gradient de politique (Policy Gradient Methods)
- MSE : Erreur quadratique moyenne (Mean Squared Error)
- NB : Classificateur bayésien naïf (Naive Bayes)
- NBC : Code national du bâtiment (National Building Code)
- OLSR : Régression linéaire par moindres carrés ordinaires (Ordinary Least Squares Regression)
- OOF : Prédictions hors pli (Out-of-Fold)
- OR : Rapport de cotes (Odds Ratio)
- PBL : Connecteur Perfobond Leiste (Perfobond Leiste Connector)
- R^2 : Coefficient de détermination

- RBF : Fonction de base radiale (Radial Basis Function)
- RF : Forêt aléatoire (Random Forest)
- RMSE : Racine de l’erreur quadratique moyenne (Root Mean Squared Error)
- RSS : Somme des carrés des résidus (Residual Sum of Squares)
- SQL : Apprentissage Q (Q-Learning)
- SRCW : Mur composite acier–béton (Steel–Concrete Composite Wall)
- SVM : Machine à vecteurs de support (Support Vector Machine)
- SVR : Régression par vecteurs de support (Support Vector Regression)
- TAL : Traitement automatique du langage naturel
- TSS : Somme totale des carrés (Total Sum of Squares)
- UPN : Profilé normalisé en U
- XGB : Extreme Gradient Boosting (XGBoost)

Introduction Générale

Les structures mixtes acier–béton occupent une place prépondérante dans le domaine de la construction moderne, notamment pour les ponts, les bâtiments industriels et les ouvrages de grande portée. Leur efficacité repose sur la combinaison des propriétés complémentaires des deux matériaux : l'acier offre une résistance élevée à la traction ainsi qu'une grande rigidité, tandis que le béton reprend efficacement les efforts de compression. Cette propriété est due aux connecteurs de cisaillement, qui jouent un rôle central dans le transfert des efforts entre les deux éléments.

Parmi les différents types de connecteurs disponibles, les connecteurs étudiés dans ce travail se distinguent par leur efficacité mécanique et leur facilité d'exécution. Cependant, leur comportement sous l'effet des charges de cisaillement est influencé par plusieurs paramètres géométriques et propriétés matérielles, ce qui rend leur modélisation analytique complexe et peut ne pas fournir une précision suffisante dans certains cas.

Les formules analytiques adoptées dans les normes actuelles, notamment l'Eurocode 4, présentent plusieurs limitations importantes, puisqu'elles ont été développées à partir d'un nombre limité d'études expérimentales et ne couvrent pas l'ensemble des configurations géométriques ni les différents types de béton utilisés dans les applications pratiques. En particulier, l'application de ces formules au béton à haute résistance reste insuffisamment documentée.

Dans ce contexte, l'apprentissage automatique (Machine Learning) apparaît comme une alternative prometteuse. Grâce à sa capacité à apprendre des relations complexes et non linéaires à partir des données expérimentales, il permet de développer des modèles prédictifs offrant une meilleure précision et une capacité de généralisation plus élevée.

Cette étude se limite à la prédiction de la résistance au cisaillement des connecteurs en U ancrés dans le béton ordinaire et à haute résistance à partir d'une base de données expérimentale issue d'essais de poussée. Les analyses numériques par éléments finis, les essais expérimentaux supplémentaires et les autres types de connecteurs ne sont pas inclus dans le cadre de ce travail.

Objectifs de l'étude

Cette étude vise à atteindre trois objectifs principaux :

- Développer et comparer plusieurs modèles d'apprentissage automatique, notamment : RF, XGBoost, SVR, MLP et k-NN, afin de prédire la résistance au cisaillement des connecteurs de cisaillement ancrés dans le béton ordinaire et le béton à haute résistance.
- Développer un modèle hybride de type Stacking basé sur la combinaison des prédictions des modèles individuels afin d'améliorer la précision des prédictions et d'augmenter la stabilité des résultats.
- Étudier et analyser l'importance des variables d'entrée afin d'identifier les paramètres physiques ayant le plus d'influence sur la résistance au cisaillement, dans le but de proposer des recommandations pratiques pour la conception des structures.

Structure du mémoire

Cette étude a été organisée en quatre chapitres principaux comme suit :

- **Chapitre 1(Aperçu Général de L'apprentissage Automatique)** Ce chapitre présente les bases théoriques nécessaires à la compréhension du sujet étudié. Il passe en revue les principaux types d'apprentissage automatique, notamment l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement, ainsi que les algorithmes fondamentaux et les différentes étapes du cycle de vie des projets d'apprentissage automatique.

- **Chapitre 2(Revue de littérature et élaboration du dataset)** Ce chapitre traite des études expérimentales, numériques et analytiques relatives aux connecteurs de cisaillement. Il présente également le processus de construction d'une base de données comprenant 618 essais de poussée (Push-Out Tests) collectés à partir de la littérature scientifique internationale.

- **Chapitre 3(Programmation du ML avec Python)** Ce chapitre décrit en détail les différentes étapes du pipeline d'apprentissage automatique, depuis l'analyse exploratoire des données (EDA), en passant par le prétraitement et l'augmentation des données, jusqu'à la sélection et l'optimisation des cinq modèles à l'aide de GridSearchCV, puis le développement d'un modèle hybride de type Stacking basé sur la régression Ridge.

- **Chapitre 4(Résultats et discussion)** Ce chapitre présente et analyse les résultats obtenus à travers une comparaison des performances des six modèles (cinq modèles de base et un modèle hybride), l'interprétation des graphiques des résidus et des diagrammes de Taylor, ainsi qu'une discussion sur l'importance des variables, accompagnée de recommandations et de perspectives futures.

Cette étude s'achève par une conclusion générale qui résume les principales contributions et les résultats obtenus, tout en proposant des perspectives et des orientations pour des recherches futures dans ce domaine.

Chapitre 01

Chapitre 1

Aperçu Général de L'apprentissage Automatique

1.1 Introduction

L'intelligence artificielle (IA) vise à reproduire certaines capacités humaines, notamment la résolution de problèmes. L'apprentissage automatique est une sous-discipline permettant aux systèmes de s'améliorer automatiquement à partir des données et de l'expérience, constituant ainsi le cœur des applications modernes de l'IA.[1]

Selon Mitchell (1997), l'apprentissage automatique est l'étude des algorithmes permettant aux systèmes de s'améliorer automatiquement à partir de l'expérience. Ce domaine a connu un développement important ces dernières années, notamment grâce à ses liens étroits avec le traitement automatique du langage (TAL), comme le soulignent Manning et Schütze (1999).[2]

Ce chapitre présente les bases de l'apprentissage automatique, en introduisant les principaux concepts nécessaires à la compréhension des méthodes d'apprentissage à partir des données. Il aborde les principales catégories (supervisé, non supervisé et par renforcement) ainsi qu'un aperçu des algorithmes les plus utilisés. Enfin, il décrit le cycle de vie d'une tâche d'apprentissage automatique, de l'entraînement des modèles jusqu'à leur évaluation.

1.2 Definition

Un système d'apprentissage automatique construit des modèles à partir de données. Le processus de construction de ces modèles est appelé phase d'apprentissage ou d'entraînement. Les données utilisées durant cette phase sont appelées données d'apprentissage (training data).

En tant que domaine de recherche, l'apprentissage automatique (AA) s'intéresse à l'étude des problèmes pouvant être résolus à l'aide de méthodes d'apprentissage, ainsi qu'au développement de modèles et d'algorithmes capables d'apprendre à partir des données.

L'apprentissage automatique, également appelé apprentissage artificiel, désigne l'ensemble des techniques et algorithmes permettant à un système informatique d'améliorer automatiquement ses performances en exploitant l'expérience et les données disponibles.[3]

1.3 Types d'Apprentissage Automatique

Il existe différentes méthodes pour apprendre automatiquement à partir des données, et le choix de la méthode dépend du type de problème à résoudre ainsi que des données disponibles. La figure 1.1 présente un résumé des principaux types d'apprentissage automatique.[4]

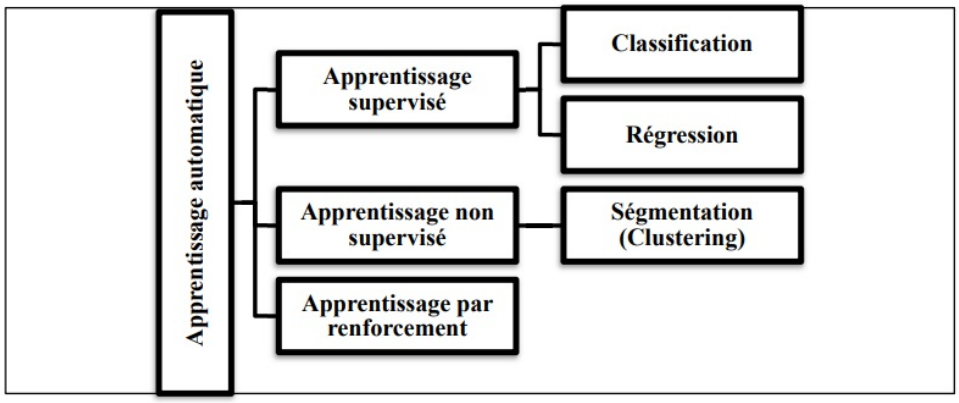


Figure 1.1 : Les grandes classes d'apprentissage automatique [4]

1.3.1 Apprentissage supervisé

On entend par apprentissage supervisé (Supervised Learning) la situation dans laquelle un ensemble de données d'entraînement étiquetées est disponible, c'est-à-dire que chaque échantillon de données est associé à une sortie (valeur cible) connue à l'avance. Ces données sont généralement représentées sous forme de paires entrée–sortie (x_i, y_i), constituant un ensemble de données :

$$D = \{(x_i, y_i)\}_{i=1}^{N}$$

L'objectif est d'entraîner un modèle capable d'apprendre la relation entre les entrées et les sorties, afin de prédire correctement les valeurs de sortie pour de nouvelles données non observées, tout en assurant une bonne capacité de généralisation.

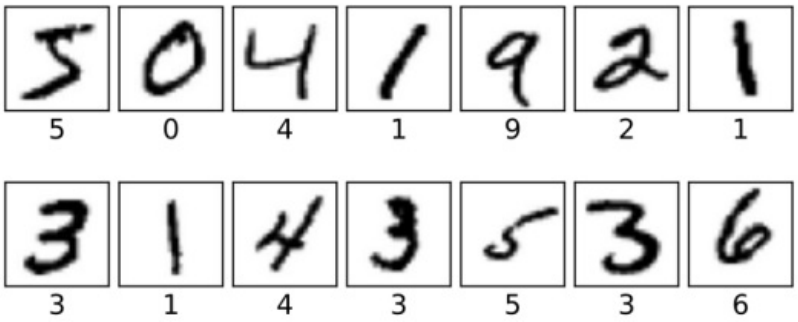


Figure 1.2 : Ensemble de données étiquetées Chaque image d'entrée est associée avec la prédiction voulue

[1]

L'apprentissage supervisé est principalement utilisé dans deux types de problèmes :

1.3.1.1 Régression

Elle consiste à modéliser la relation entre les entrées et une sortie continue. Ce type de problème est utilisé, par exemple, pour estimer la consommation énergétique ou prévoir les prix des marchés financiers.

Tableau 1.1 : Exemple d'un ensemble de données d'apprentissage supervisé (Régression)

[4]

Date	VWAP	SMA 20	Croisement (20,50)	MACD	RSI	BB%	Sentiment	Prix (CAD)
01-08-2018	51.54	51.95	105%	-30%	-40%	45%	Positif	52.3
02-08-2018	52.37	53.31	102%	-30%	-39%	45%	Positif	54.1
03-08-2018	51.57	53.03	91%	-30%	-40%	46%	Négatif	51.5
04-08-2018	50.9	52.86	86%	-31%	-41%	47%	Négatif	51.9
05-08-2018	50.33	52.79	88%	-31%	-41%	48%	Négatif	51.05
06-08-2018	50.03	53	87%	-32%	-42%	49%	Positif	52
07-08-2018	50.66	53.67	79%	-31%	-41%	49%	Positif	50.45
08-08-2018	49.75	53.13	80%	-32%	-39%	49%	Positif	50.25
09-08-2018	48.86	52.07	73%	-33%	-39%	51%	Neutre	49.61
10-08-2018	48.5	51.99	73%	-32%	-38%	53%	Neutre	49.03

1.3.1.2 Classification

Elle vise à attribuer un échantillon à une classe parmi un ensemble limité de catégories. Parmi les exemples, on peut citer la reconnaissance de chiffres manuscrits ou la classification de tumeurs en « bénignes » ou « malignes ».

Il convient de noter que cette distinction entre régression et classification n'est pas exhaustive, car il existe des modèles capables de traiter des sorties de grande dimension, notamment dans le domaine du traitement automatique du langage naturel (TAL), où les données sont souvent représentées sous forme de vecteurs de grande dimension.

Tableau 1.2 : Exemple d'un ensemble de données d'apprentissage supervisé (Classification)

[4]

Données d'entrée								**Cible**
Date	**VWAP**	**SMA 20**	**Croisement de MM (20,50)**	**MACD**	**RSI**	**BB %**	**Sentiment**	**Tendance**
01-08-2018	51.54	51.95	105%	-30%	-40%	45%	Positif	Baisse
02-08-2018	52.37	53.31	102%	-30%	-39%	45%	Positif	Baisse
03-08-2018	51.57	53.03	91%	-30%	-40%	46%	Négatif	Baisse
04-08-2018	50.9	52.86	86%	-31%	-41%	47%	Négatif	Hausse
05-08-2018	50.33	52.79	88%	-31%	-41%	48%	Négatif	Hausse
06-08-2018	50.03	53	87%	-32%	-42%	49%	Positif	Hausse
07-08-2018	50.66	53.67	79%	-31%	-41%	49%	Positif	Neutre
08-08-2018	49.75	53.13	80%	-32%	-39%	49%	Positif	Baisse
09-08-2018	48.86	52.07	73%	-33%	-39%	51%	Neutre	Neutre
10-08-2018	48.5	51.99	73%	-32%	-38%	53%	Neutre	Neutre

Parmi les principaux algorithmes d'apprentissage supervisé, on peut citer :

- les k plus proches voisins (k-NN)
- la régression linéaire
- la régression logistique

- les arbres de décision
- les forêts aléatoires
- les machines à vecteurs de support (SVM)[1]

1.3.2 Apprentissage non-supervisé

On parle d'apprentissage non supervisé lorsque les données disponibles ne sont pas étiquetées, c'est-à-dire qu'aucune sortie cible n'est associée aux observations. Dans ce cas, l'ensemble de données est représenté uniquement par les entrées :

$$D = \{x_i\}_{i=1}^{N}$$

L'objectif est d'identifier des structures ou des régularités intrinsèques au sein des données.



Figure 1.3 : Ensemble de données non-étiquetées Les images d'entrées, extraites de la base de données MNIST Fashion, ne sont pas associées à une sortie cible
[1]

L'apprentissage non supervisé repose principalement sur des techniques de regroupement (clustering), dont le but est de partitionner les données en plusieurs groupes. Les éléments appartenant à un même groupe présentent des caractéristiques similaires, tandis qu'ils sont significativement différents de ceux appartenant aux autres groupes. Ce processus permet ainsi d'organiser les données en classes de manière automatique.

À titre d'exemple, ces méthodes peuvent être utilisées dans le domaine médical pour regrouper des patients selon des profils similaires, afin de mieux anticiper leurs réactions à certains traitements.

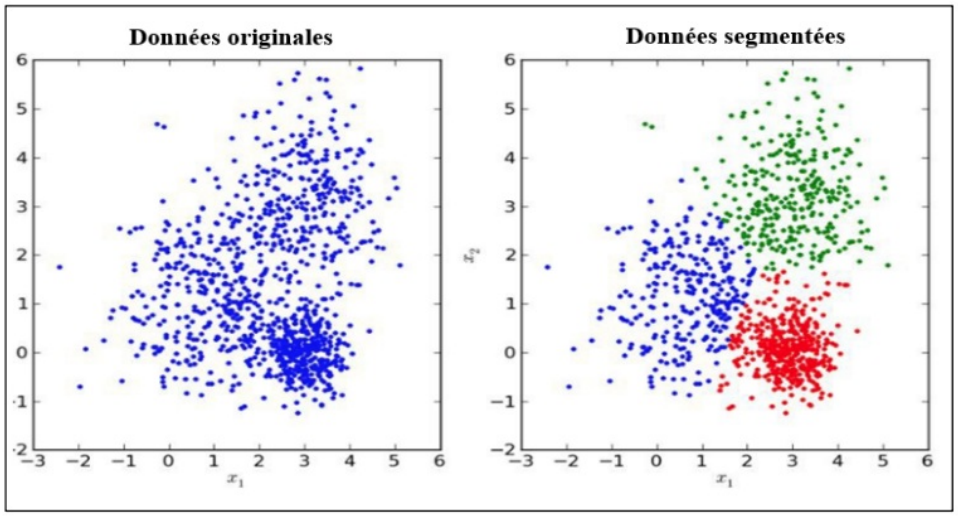


Figure 1.4 : Exemple d'apprentissage non supervisé
[4]

1.3.3 Apprentissage par renforcement

L'apprentissage par renforcement est une méthode d'apprentissage automatique basée sur l'interaction entre un agent et son environnement. L'objectif est d'optimiser une politique de décision à travers un mécanisme de récompenses et de pénalités, permettant de guider l'agent vers des actions optimales.

Le processus est itératif : l'agent exécute une action, observe l'état de l'environnement, reçoit une récompense, puis met à jour ses paramètres afin d'améliorer ses performances. Ce cycle d'apprentissage favorise la convergence vers une stratégie optimale au fil des itérations.

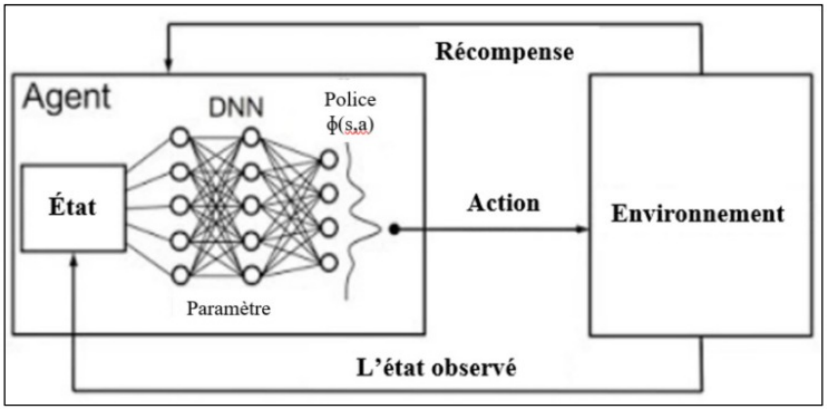


Figure 1.5 : Schéma descriptive de l'apprentissage par renforcement
[4]

Les algorithmes génétiques constituent une approche particulière inspirée de

la sélection naturelle, où une population de solutions est évaluée, sélectionnée puis modifiée par mutation afin d'améliorer progressivement les performances.

Cette approche est largement utilisée dans des domaines complexes tels que la robotique, les jeux (exemple : AlphaGo), la conduite autonome et l'optimisation sous contraintes. Elle est souvent combinée avec l'apprentissage profond (deep learning) pour traiter des environnements à haute dimension.

Ainsi, l'apprentissage par renforcement permet d'apprendre des stratégies optimales dans des environnements dynamiques en s'appuyant sur un retour d'expérience continu.[4]

1.4 Cycle de vie d'une tâche d'apprentissage automatique

Le cycle de vie d'une tâche d'apprentissage automatique suit généralement plusieurs étapes, comme illustré à la figure 1.6. Tout d'abord, il s'agit de choisir un algorithme d'apprentissage approprié. Ensuite, cet algorithme est entraîné à l'aide d'un ensemble de données appelé ensemble d'apprentissage. Enfin, ses performances sont évaluées en le testant sur un autre ensemble de données, appelé ensemble de test.

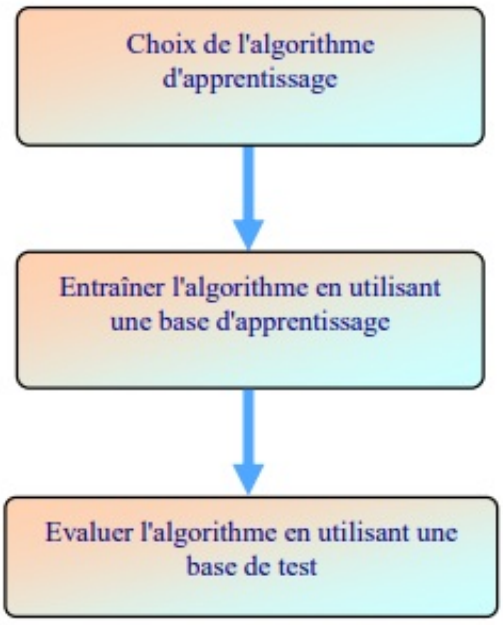


Figure 1.6 : Flux d'apprentissage automatique
[3]

Le choix de l'algorithme dépend de la nature des connaissances à acquérir ainsi que du type de données disponibles en entrée et en sortie. Il existe ainsi différentes catégories d'algorithmes, telles que les algorithmes de règles d'association, les méthodes basées sur des modèles statistiques, les approches de partitionnement, les algorithmes de couverture, les techniques de régression, les modèles linéaires, les méthodes d'apprentissage par instance ainsi que les algorithmes de classification.[3]

1.5 Les algorithmes d'apprentissage automatique

1.5.1 Algorithmes d'apprentissage supervisé

- **K-Plus Proches Voisins (KNN)**

L'algorithme des K-plus proches voisins (KNN) est une méthode d'apprentissage supervisé simple et facilement interprétable, utilisée aussi bien pour des problèmes de classification que de régression.

Pour chaque nouveau point de données, on calcule la distance entre ce point et les observations existantes dans le jeu de données. Le paramètre K représente le nombre de voisins les plus proches à prendre en compte. Une fois ces voisins identifiés, une décision est prise par vote majoritaire afin d'assigner le nouveau point à la classe dominante parmi ses voisins.

La prédiction en KNN dépend donc des K observations les plus proches du point à classifier. En pratique, on choisit souvent une valeur impaire de K afin d'éviter les égalités lors du vote.

Pour mesurer la proximité entre les points, différentes métriques de distance peuvent être utilisées, notamment : ***Distance euclidienne***

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{1.1}$$

bien que d'autres mesures comme la distance de Minkowski, la distance cosinus ou le chi-deux soient également utilisées selon le type de données.

Le KNN peut parfois surpasser d'autres algorithmes dans certains cas où les caractéristiques des données sont difficiles à exploiter directement ou lorsque des valeurs manquantes compliquent la modélisation classique. Il est notamment utilisé dans des domaines tels que l'analyse de texte, la reconnaissance de formes et l'identification d'empreintes digitales, où la classification basée sur la proximité des données s'avère efficace.[5]

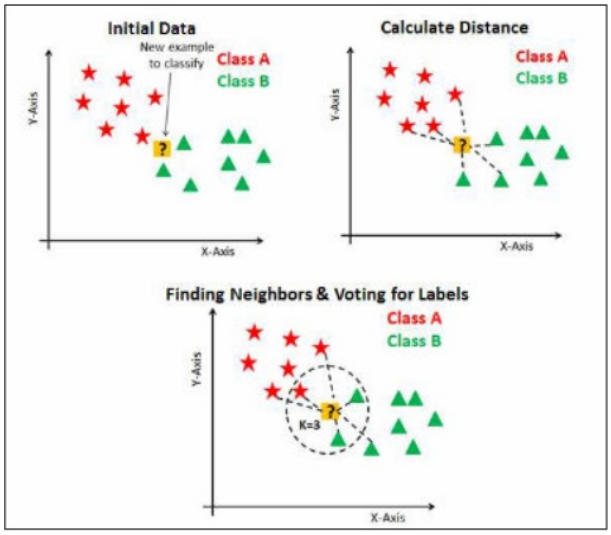


Figure 1.7 : Étapes pour l'algorithme KNN
[5]

- **Machine à Vecteurs De Support(SVM)**

La machine à vecteurs de support (SVM) est une méthode d'apprentissage supervisé particulièrement efficace pour les petits ensembles de données. Elle repose sur le principe de minimisation du risque structurel, contrairement aux approches classiques basées sur le risque empirique, ce qui lui permet d'obtenir de meilleures performances de généralisation.

Le principe du SVM consiste à déterminer un hyperplan optimal capable de séparer deux classes sans erreur dans le cas linéairement séparable. Cet hyperplan, noté H, est entouré de deux hyperplans parallèles H_1 et H_2 passant par les points les plus proches de chaque classe. La distance entre ces deux hyperplans est appelée la marge, que l'on cherche à maximiser.[10]

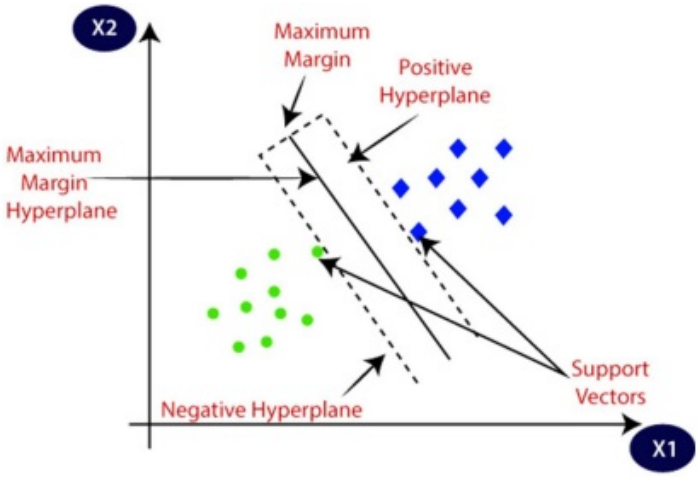


Figure 1.8 : Fonctionnement de la machine à vecteurs de support
[12]

Dans un espace de dimension n, la fonction de décision est donnée par :

$$g(x) = \omega \cdot x + b \tag{1.2}$$

et l'équation de l'hyperplan est :

$$\omega \cdot x + b = 0 \tag{1.3}$$

Pour garantir une classification correcte, les données doivent vérifier la condition :

$$y_i(\omega \cdot x_i + b) \geq 1 \quad \text{pour } i = 1, 2, \ldots, n \tag{1.4}$$

La marge est égale à :

$$\frac{2}{\|\omega\|} \tag{1.5}$$

L'hyperplan optimal est celui qui maximise cette marge, ou de manière équivalente, minimise la norme du vecteur ω. Les points les plus proches de l'hyperplan, appelés vecteurs de support, déterminent la position de la frontière de décision.[10]

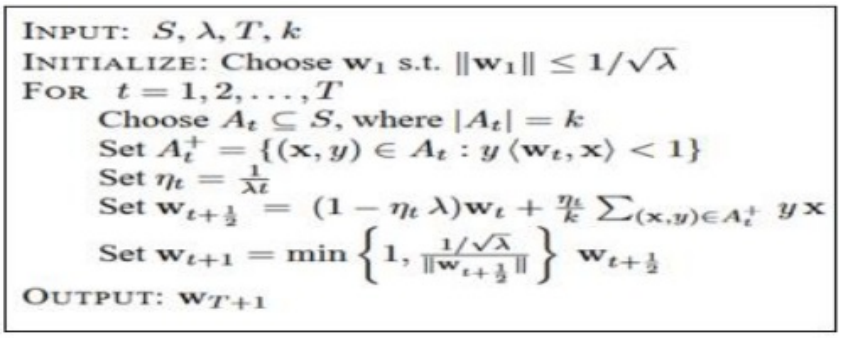


Figure 1.9 : Pseudo-code pour la machine à vecteurs de support [13]

- **Régression par vecteurs de support (SVR)**

Support Vector Regression (SVR) est une méthode inspirée de l'algorithme des Support Vector Machines (SVM) utilisé pour les problèmes de classification binaire. Son principe repose sur l'introduction d'une marge de tolérance appelée ε-sensibilité, qui consiste à ajuster un tube de largeur ε autour de la fonction de prédiction. Les erreurs situées à l'intérieur de ce tube sont considérées comme acceptables et ne sont donc pas pénalisées. En revanche, les observations situées à l'extérieur du tube subissent une pénalisation proportionnelle à leur éloignement par rapport à la fonction prédite, selon un mécanisme similaire à celui utilisé dans les SVM de classification.

À l'instar des SVM, le SVR cherche à déterminer un hyperplan optimal dans un espace de caractéristiques transformé par un noyau (kernel), afin d'obtenir une bonne capacité de généralisation tout en conservant une représentation efficace des données d'origine.[11]

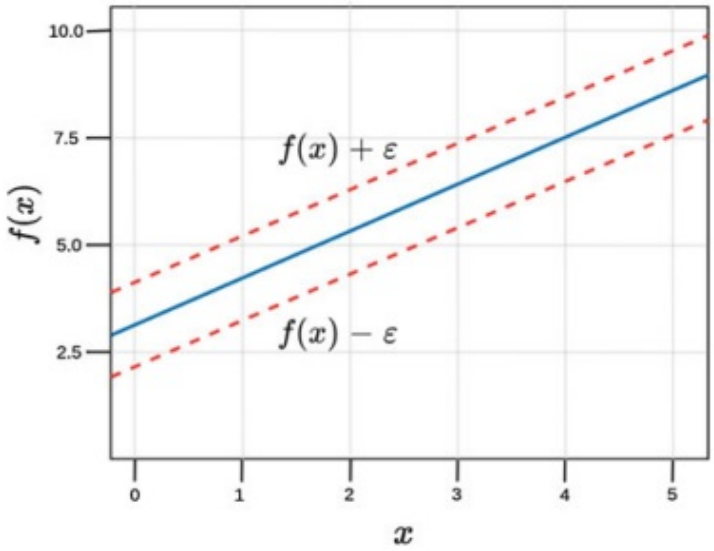


Figure 1.10 : ε-insensitive regression band. The solid blue line represents the estimated regression curve $f(x)$.

[11]

- **Le perceptron multicouche(MLP)**

Le perceptron multicouche (MLP) est un réseau de neurones très utilisé. Les informations circulent généralement dans un seul sens, de l'entrée vers la sortie, sans boucle : on parle de réseau feedforward.

Les couches entre l'entrée et la sortie sont appelées couches cachées. La couche d'entrée sert seulement à transmettre les données.

Il existe aussi des réseaux feedback (ou récurrents), où les signaux peuvent circuler dans les deux sens. Ils sont plus complexes et dynamiques.

L'ajout de plusieurs couches permet de traiter des problèmes plus complexes : une seule couche donne des séparations simples, alors que plusieurs couches permettent des régions de décision plus compliquées, car la composition de fonctions linéaires reste elle-même une fonction linéaire :

$$f(g(x)) = ax + b$$

Enfin, si on utilise des fonctions d'activation linéaires, ajouter des couches n'apporte pas de gain, car une combinaison de fonctions linéaires reste linéaire.[9]

- **Arbres de décision**

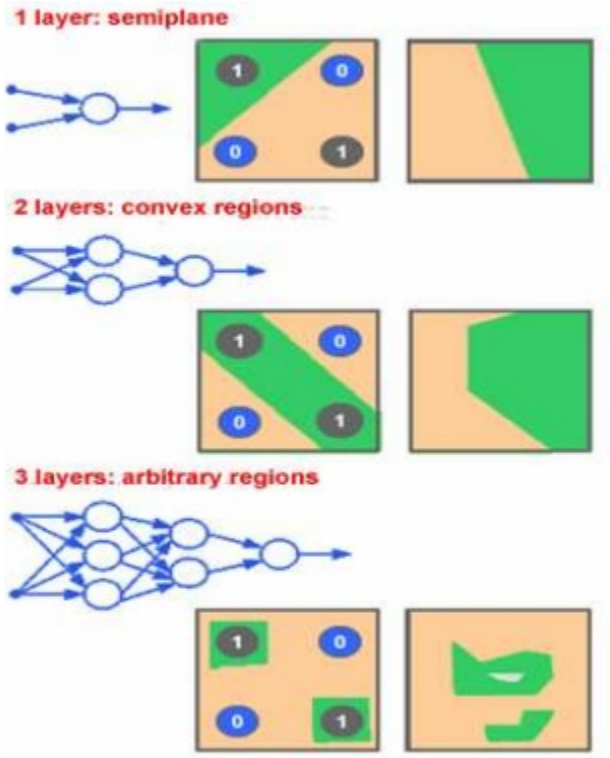


Figure 1.11 : Régions de décision des perceptrons multicouches [9]

Les arbres de décision constituent une méthode de classification basée sur une partition récursive de l'espace des données. Ils sont organisés sous forme d'une structure arborescente composée d'un nœud racine (root) sans arête entrante, de nœuds internes (ou nœuds de test) possédant des arêtes sortantes, et de nœuds terminaux appelés feuilles.

Chaque nœud de test permet de diviser l'espace des données en plusieurs sous-espaces en fonction d'un attribut donné. Dans les cas simples, chaque test porte sur un seul attribut, ce qui permet de partitionner les données selon ses différentes valeurs. Pour les attributs numériques, la séparation se fait généralement à l'aide d'intervalles.

Chaque feuille est associée à une classe représentant la valeur cible la plus appropriée. Elle peut également contenir une distribution de probabilités indiquant la probabilité d'appartenance à chaque classe. La classification d'une instance s'effectue en la faisant parcourir depuis la racine jusqu'à une feuille, en suivant les décisions prises à chaque nœud.

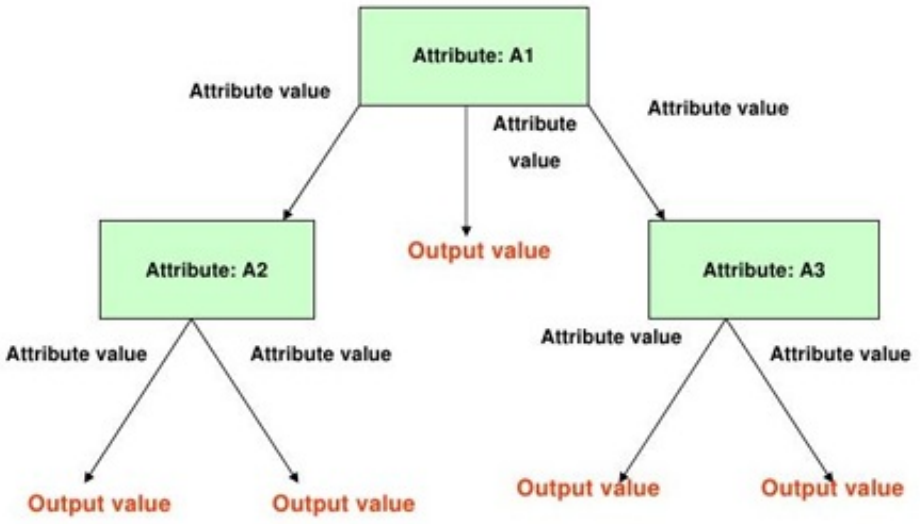


Figure 1.12 : Exemple d'arbre de décision
[6]

D'un point de vue géométrique, les arbres de décision peuvent être interprétés comme un ensemble d'hyperplans orthogonaux aux axes, partitionnant ainsi l'espace des données. En pratique, les arbres de décision simples sont souvent privilégiés, car ils sont plus faciles à interpréter tout en restant efficaces pour la prédiction.[6]

- **forêt aléatoire**

La forêt aléatoire (Random Forest) est une technique d'apprentissage automatique qui combine plusieurs arbres de décision afin de réduire la corrélation entre les variables et d'améliorer les performances du modèle. Elle repose sur deux principes principaux de randomisation : l'échantillonnage aléatoire des données d'apprentissage et la sélection aléatoire d'un sous-ensemble de variables pour construire chaque arbre. Ces mécanismes permettent de réduire le surapprentissage (overfitting) et d'augmenter la précision du modèle.

La complexité computationnelle de la méthode est de l'ordre de $O(n)$, où n représente le nombre d'échantillons, ce qui la rend efficace même pour de grands ensembles de données. De plus, grâce à sa structure composée de plusieurs arbres indépendants, la forêt aléatoire peut être exécutée en parallèle, ce qui améliore la vitesse de calcul.

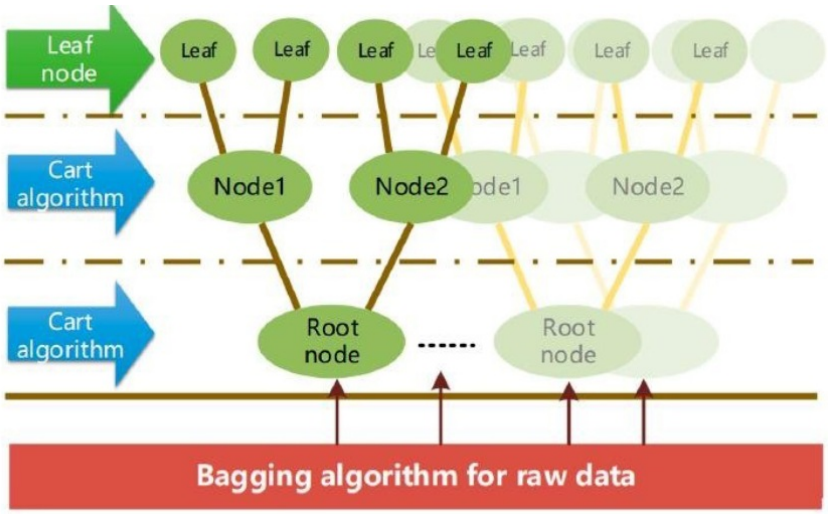


Figure 1.13 : Schéma du diagramme de l'algorithme de forêt aléatoire [14]

Dans certaines applications, les données peuvent être segmentées en intervalles afin d'améliorer la qualité des prédictions. Cette segmentation peut se faire selon deux approches : soit en divisant les données en intervalles de taille égale, soit en adaptant les intervalles en fonction de la densité des données. Cette deuxième approche permet de mieux se concentrer sur les zones les plus significatives.

La segmentation des données présente plusieurs avantages, notamment la réduction des interférences, l'amélioration de la précision des prédictions et une analyse plus efficace des tendances et des patterns. En combinant cette stratégie avec la méthode Random Forest, il est possible d'obtenir des résultats plus précis et plus robustes.[14]

- **XGBoost**

XGBoost est un algorithme d'apprentissage supervisé qui met en œuvre un processus appelé **boosting** afin de produire des modèles très précis.

L'apprentissage supervisé correspond à la tâche qui consiste à inférer un modèle prédictif à partir d'un ensemble d'exemples d'entraînement étiquetés. Ce modèle peut ensuite être utilisé pour prédire les résultats d'exemples jamais vus auparavant. Les entrées de l'algorithme sont des paires d'exemples d'apprentissage $(\tilde{x}_0, y_0), (\tilde{x}_1, y_1), \ldots, (\tilde{x}_n, y_n)$, où $\tilde{x}$ est un vecteur de caractéristiques décrivant l'exemple et y est son étiquette. L'apprentissage supervisé peut ainsi être

vu comme l'apprentissage d'une fonction $F(\tilde{x}) = y$ capable d'attribuer correctement une étiquette à de nouvelles instances. On intègre le modèle d'arbres de décision en utilisant une méthode additive, en supposant un total de K arbres. On note F l'ensemble des modèles d'arbres de base. Le modèle prédictif s'écrit alors :

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in F \tag{1.6}$$

La fonction objective est définie comme suit :

$$L = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{1.7}$$

où l est la fonction de perte, qui mesure l'erreur entre la valeur prédite et la valeur réelle. Ω est une fonction de régularisation utilisée pour éviter le surapprentissage :

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2 \tag{1.8}$$

où T représente le nombre de feuilles de l'arbre, et w représente les poids associés aux feuilles de chaque arbre.

Après une expansion de Taylor au second ordre de la fonction objective et d'autres calculs détaillés dans le matériel complémentaire, on obtient finalement le gain d'information lors de chaque division (split) :

$$\text{Gain} = \frac{1}{2}\left[\frac{\left(\sum_{i\in I_L} g_i\right)^2}{\sum_{i\in I_L} h_i + \lambda} + \frac{\left(\sum_{i\in I_R} g_i\right)^2}{\sum_{i\in I_R} h_i + \lambda} - \frac{\left(\sum_{i\in I} g_i\right)^2}{\sum_{i\in I} h_i + \lambda}\right] - \gamma \tag{1.9}$$

L'apprentissage supervisé permet de résoudre des problèmes de **classification** et de **régression**. Dans la classification, la variable cible y est discrète, tandis qu'en régression elle prend des valeurs continues.

XGBoost est un algorithme de **boosting** basé sur les arbres de décision, appartenant aux méthodes d'apprentissage en ensemble (ensemble learning). Son principe consiste à construire plusieurs modèles successifs, où chaque nouvel

arbre cherche à corriger les erreurs des arbres précédents afin d'améliorer progressivement les performances du modèle global.[16]

1.5.2 Algorithmes d'Apprentissage non-supervisé

- **K-moyennes**

K-Means est l'algorithme le plus basique utilisé pour partitionner des données en un nombre donné de clusters. Pour k clusters, k représentants sont sélectionnés à partir des données initiales. Ces représentants sont appelés **centroïdes**.

La proximité de chaque élément de données avec chaque centroïde est calculée, puis chaque point est affecté au cluster dont la distance au centroïde est la plus faible. La version classique de l'algorithme K-Means utilise la distance euclidienne comme mesure de proximité entre les points de données, généralement représentés dans un espace à deux dimensions.[5]

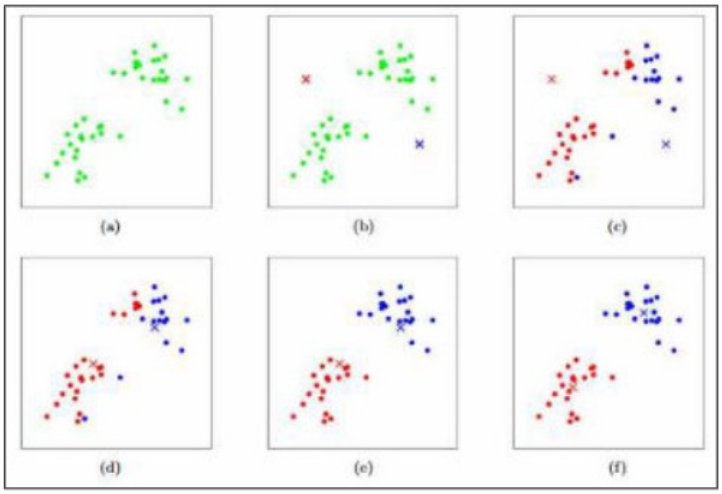


Figure 1.14 : Étapes de l'algorithme K-moyennes
[5]

Une fois tous les points attribués à un cluster, de nouveaux centroïdes sont recalculés en prenant le barycentre (la moyenne) de chaque cluster. Le processus est ensuite répété jusqu'à ce qu'un critère d'arrêt soit atteint, généralement lorsque deux itérations consécutives ne montrent pas de changement significatif dans la répartition des clusters.

Bien que l'algorithme puisse être démontré comme convergent dans tous les cas, il ne garantit pas une partition optimale des données en clusters. De plus,

il n'existe pas de règle stricte pour choisir le nombre de clusters ni les représentants initiaux, ce qui peut conduire à des résultats différents selon les valeurs choisies.[5]

1.5.3 Algorithmes d'Apprentissage par renforcement

- **Speedy Q-Learning (SQL)**

Le **Speedy Q-Learning (SQL)** est une amélioration du Q-learning classique visant à accélérer la convergence tout en conservant une approche sans modèle (*model-free*). Contrairement au Q-learning standard, SQL utilise deux estimations successives de la fonction de valeur (Q_k et Q_{k-1}), permettant une mise à jour plus efficace et une propagation plus rapide de l'information.

La règle de mise à jour est donnée par :

$$Q_{k+1}(x,a)=Q_k(x,a)+\alpha_k(T_kQ_{k-1}(x,a)-Q_k(x,a))+(1-\alpha_k)(T_kQ_k(x,a)-T_kQ_{k-1}(x,a)) \tag{1.10}$$

avec :

$$\alpha_k = \frac{1}{k+1} \tag{1.11}$$

Le SQL améliore ainsi la vitesse de convergence par rapport au Q-learning classique, notamment lorsque le facteur d'actualisation γ est élevé.[17]

- **Algorithme du Bagging**

Le **bagging** (*Bootstrap Aggregating*) est une méthode introduite par Breiman visant à améliorer les performances des modèles d'apprentissage par la combinaison de plusieurs classificateurs entraînés sur des échantillons bootstrap générés aléatoirement avec remise.

Chaque modèle est construit à partir d'un ensemble différent de données, puis les prédictions sont combinées par un mécanisme de vote afin d'obtenir le classificateur final.

La probabilité qu'une observation soit sélectionnée au moins une fois dans un échantillon bootstrap est donnée par :

$$1 - \left(1 - \frac{1}{m}\right)^m \tag{1.12}$$

Lorsque m devient grand :

$$1 - \frac{1}{e} \approx 63,2\% \tag{1.13}$$

Ainsi, chaque échantillon bootstrap contient en moyenne environ $63,2\%$ d'observations distinctes. Le bagging améliore principalement les performances des modèles instables comme les arbres de décision et les réseaux de neurones en réduisant la variance.[19]

- **Algorithme du Boosting**

Le **boosting**, notamment l'algorithme **AdaBoost**, améliore les performances des apprenants faibles en construisant plusieurs modèles de manière séquentielle. Contrairement au bagging, chaque nouveau modèle accorde davantage d'importance aux observations mal classées lors des étapes précédentes.

Le modèle final est obtenu à travers un vote pondéré des classificateurs générés, où chaque modèle contribue selon sa précision.

Les poids des observations sont mis à jour selon :

$$w(x_j) \leftarrow \begin{cases} \dfrac{w(x_j)}{2\varepsilon_i} & \text{si } C_i(x_j) \neq y_j \\ \dfrac{w(x_j)}{2(1-\varepsilon_i)} & \text{sinon} \end{cases} \tag{1.14}$$

Cette stratégie augmente l'influence des exemples mal classés et améliore progressivement les performances du modèle global. L'efficacité du boosting repose sur l'existence d'apprenants faibles ayant une erreur inférieure à $\frac{1}{2}$.[19]

1.6 Conclusion

Ce chapitre a permis de présenter les bases théoriques de l'apprentissage automatique ainsi que ses principales catégories et algorithmes fondamentaux.

Il met en évidence que la performance d'un modèle dépend du choix de la méthode et de la nature des données.

Ainsi, il constitue un socle théorique essentiel pour la suite du travail, où des méthodes plus spécifiques seront étudiées et appliquées à des problèmes concrets.

Chapitre 02

Chapitre 2

Revue de littérature et élaboration du dataset

2.1 Introduction aux structures mixtes acier–béton

2.1.1 Principe de fonctionnement

Les structures mixtes acier–béton exploitent la complémentarité des matériaux : l'acier reprend les efforts de traction et le béton les efforts de compression, ce qui améliore la résistance en flexion.

La dalle en béton contribue également à la stabilité globale de la structure. Le comportement composite est assuré par les connecteurs de cisaillement, qui empêchent le glissement à l'interface acier–béton.

Ainsi, les deux matériaux travaillent ensemble comme un seul élément structural, ce qui se traduit par une augmentation de la rigidité et de la capacité portante.sollicitations.[20]

Ce chapitre présente les structures mixtes acier–béton et les connecteurs de cisaillement, avec un focus sur les connecteurs en U. Il synthétise les études existantes et met en évidence les limites des modèles actuels, justifiant une approche basée sur les données expérimentales.

2.1.2 Rôle des connecteurs de cisaillement

Les connecteurs de cisaillement assurent la liaison entre la poutre en acier et la dalle en béton dans les structures mixtes acier–béton.

Ils transmettent les efforts de cisaillement à l'interface et empêchent le glis-

sement entre les deux matériaux, garantissant ainsi un comportement composite. Cette interaction permet d'améliorer la rigidité, la résistance et la capacité portante globale de la structure.[21]

2.2 Les connecteurs de cisaillement en U (Channel Connectors)

2.2.1 Géométrie et caractéristiques

Le connecteur en U (channel connector) est soudé directement sur la semelle supérieure du profilé métallique sans équipement spécial. Dans l'étude de Fanaie et al. (2015), les sections utilisées sont UNP60, UNP80 et UNP100, avec des longueurs de 40 mm et 80 mm et des espacements variant entre 280 mm et 560 mm selon le nombre de connecteurs.

Leurs propriétés géométriques sont résumées dans le tableau suivant :

Tableau 2.1 : Propriétés géométriques des connecteurs en U [22]

Section	h (mm)	t_w (mm)	b (mm)	t_f (mm)
UNP60	60	6	30	6
UNP80	80	6	45	8
UNP100	100	6	50	8.5

Les connecteurs étudiés sont fabriqués en acier structural de type ST37 et ST52, caractérisé par une résistance mécanique adaptée aux structures mixtes acier–béton. L'utilisation de ces connecteurs présente également l'avantage de ne nécessiter aucun équipement spécial lors de l'installation, ce qui facilite leur mise en œuvre sur chantier.[22]



Figure 2.1 : Échantillon d'essai de poussée sur des connecteurs de cisaillement en U [24]

2.2.2 Mécanisme de transfert de cisaillement

Les connecteurs en U assurent le transfert du cisaillement entre la poutre en acier et la dalle en béton, permettant un comportement composite de la structure. Ce transfert repose principalement sur l'encastrement du profilé en U dans le béton et sur l'interaction à l'interface acier–béton.

Les études montrent que la configuration géométrique influence fortement le comportement mécanique : la disposition face à face améliore la résistance et retarde l'apparition des fissures par rapport à la disposition dos à dos. De plus, l'augmentation de la longueur, de la taille ou du nombre de connecteurs améliore la résistance au cisaillement et la charge de fissuration.

Enfin, le comportement global est peu influencé par le type d'acier, tandis qu'une rigidité suffisante des connecteurs reste essentielle pour limiter le glissement à l'interface acier–béton.[22]

2.3 Revue des travaux de recherche

2.3.1 Études expérimentales

- **Comportement des connecteurs de cisaillement à profilé en U : essais de décollement(Amit Pashan et al, 2006)**

Cette étude analyse le comportement des connecteurs de cisaillement en U (channel shear connectors) par essais de poussée sur dalles pleines et dalles avec bac acier. Elle vise à vérifier la formule CSA et à proposer de nouvelles équations plus précises. Les résultats montrent trois modes de rupture (rupture du connecteur, écrasement du béton, cisaillement du béton) et confirment que la CSA est conservatrice, donc une formulation améliorée a été développée.[23]

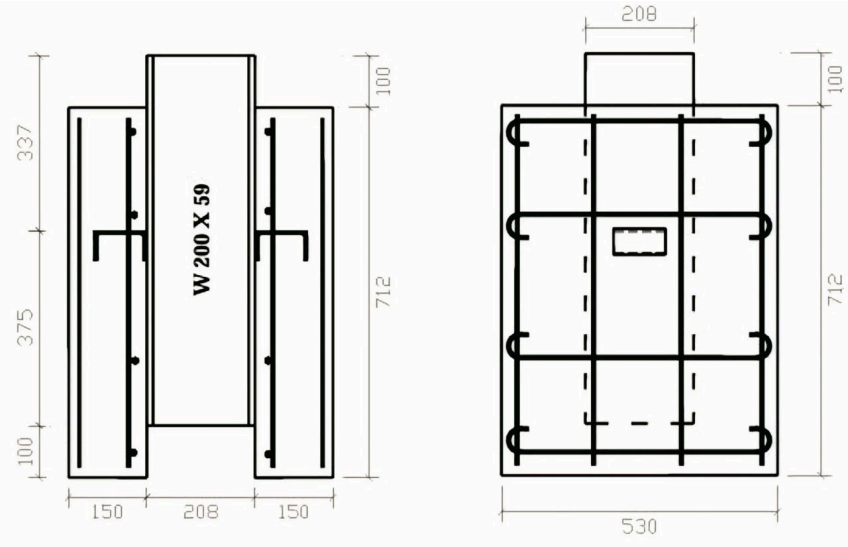


Figure 2.2 : essai de poussée (échantillon)
[23]

- **Comportement des connecteurs de cisaillement en U (S.Maleki, S.Bagheri (2008))**

Les auteurs ont étudié des connecteurs de cisaillement en U intégrés dans différents types de béton afin d'analyser leur comportement et d'évaluer leur performance sous chargement monotone et cyclique. Les résultats ont montré une diminution de résistance de 10 % à 23 % sous chargement cyclique, tandis

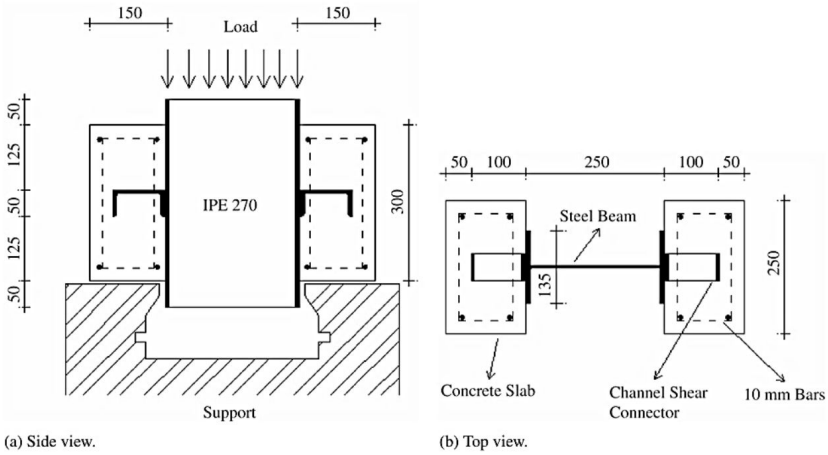


Figure 2.3 : Détails de l'échantillon
[24]

que le béton ECC a permis d'améliorer nettement la résistance et la ductilité du système composite.[24]

- **Résistance au cisaillement des connecteurs de cisaillement en U dans le béton ordinaire, renforcé et léger (Shariati et al, 2011)**

Des essais de poussées sur des connecteurs en U ont montré que le béton léger réduit la résistance et augmente la fragilité sans armatures, tandis que leur présence améliore la résistance et la ductilité. Le connecteur en U présente globalement un bon comportement ductile.[25]



Figure 2.4 : Détails de l'échantillon et configuration du test
[25]

- **Une étude expérimentale sur les connecteurs de cisaillement en U (Eray Baran et al,2012)**

Cette étude examine le comportement des connecteurs de cisaillement en U UPN à travers des essais de poussée. Elle vise à évaluer leur résistance mécanique et à la comparer aux formulations des codes AISC et CSA. Les résultats montrent que la résistance augmente avec la hauteur et la longueur des connecteurs, et que les normes actuelles sont conservatrices. Une nouvelle équation plus précise a ainsi été proposée pour mieux prédire la résistance ultime.[26]

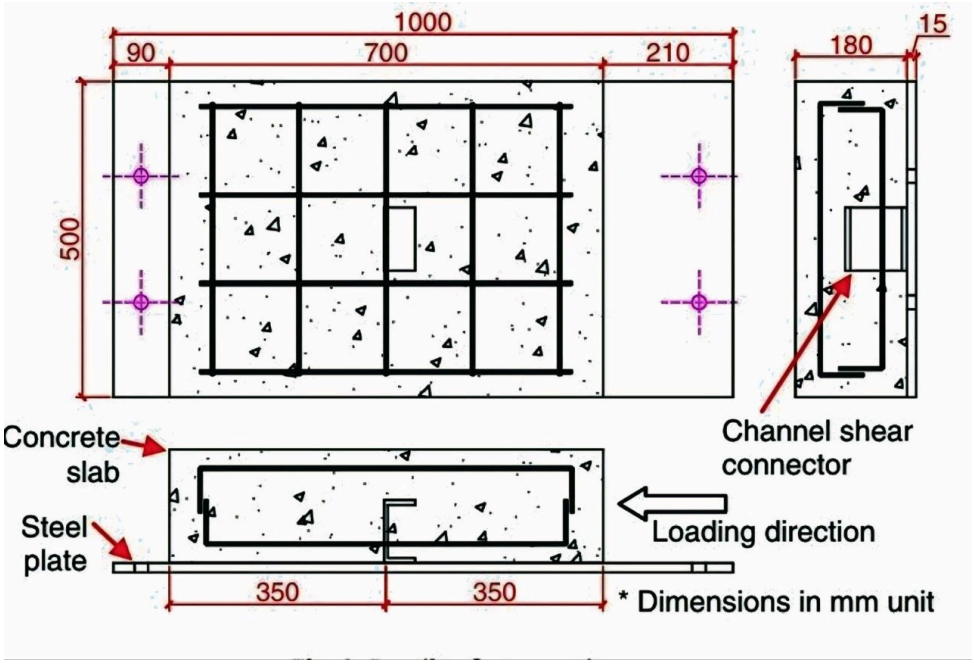


Figure 2.5 : Détails de specimens de tests
[26]

- **Comparaison du comportement entre les connecteurs de cisaillement en U et en angle sous chargement monotone et cyclique entièrement inversé (Mahdi Shariati et al, 2013)**

Cette étude compare les connecteurs en U et angle dans les structures en béton armé. Les résultats montrent une meilleure résistance et ductilité pour les connecteurs en U, tandis que les connecteurs angle présentent une réduction de résistance pouvant atteindre 49 % et une ductilité plus faible.[27]

Tableau 2.2 : Résistance des connecteurs sous chargement monotone et cyclique et taux de dégradation associé

[27]

Spécimen	Charge monotone (kN)	Charge cyclique (kN)	Dégradation (%)
C10050 (U)	152,5	140,0	7,5
A10050 (L)	141,0	106,9	23,6
C7550 (U)	139,7	130,0	21,5
A7550 (L)	109,6	96,8	25,5
C10030 (U)	112,3	102,6	30,6
A10030 (L)	77,9	52,1	49,2
C7530 (U)	109,5	99,6	36,4
A7530 (L)	69,6	63,5	36,2

- **Tests de poussée sur un nouveau connecteur de cisaillement en forme de I (Aida Mazoz et al, 2013)**

Une étude a été menée sur un connecteur de cisaillement de type I utilisé dans les ponts mixtes acier–béton. L'objectif de cette recherche était d'évaluer sa résistance et son comportement expérimental à travers des essais de poussées. Les résultats ont montré une bonne ductilité ainsi qu'une capacité portante satisfaisante. Par conséquent, ce connecteur peut être considéré comme une alternative efficace aux connecteurs traditionnels.[28]

- **Prédiction de la capacité de cisaillement des connecteurs de cisaillement en U à l'aide du modèle ANFIS(Ali Toghroli et al, 2014)**

Cette étude vise à prédire la capacité au cisaillement des connecteurs en U dans des poutres mixtes acier–béton à l'aide du modèle ANFIS basé sur des données expérimentales d'essais e poussées. Une comparaison avec la régression linéaire (LR) montre que le modèle ANFIS offre une meilleure précision et des performances supérieures.[29]

- **Étude expérimentale et modélisation par éléments finis des essais de d'arrachement sur un nouveau connecteur de cisaillement en forme de I minuscule(Messaoud Titoum et al, 2016)**

L'étude compare le nouveau connecteur de cisaillement de type I-shape avec le connecteur en U à travers des essais de poussées et la modélisation par éléments finis. Elle vise à analyser la résistance ultime, la ductilité et les modes de

rupture. Les résultats montrent que le connecteur en forme de I présente un comportement similaire à celui des connecteurs en forme de U, ce qui permet d'utiliser les mêmes équations de conception pour estimer sa capacité portante.[30]

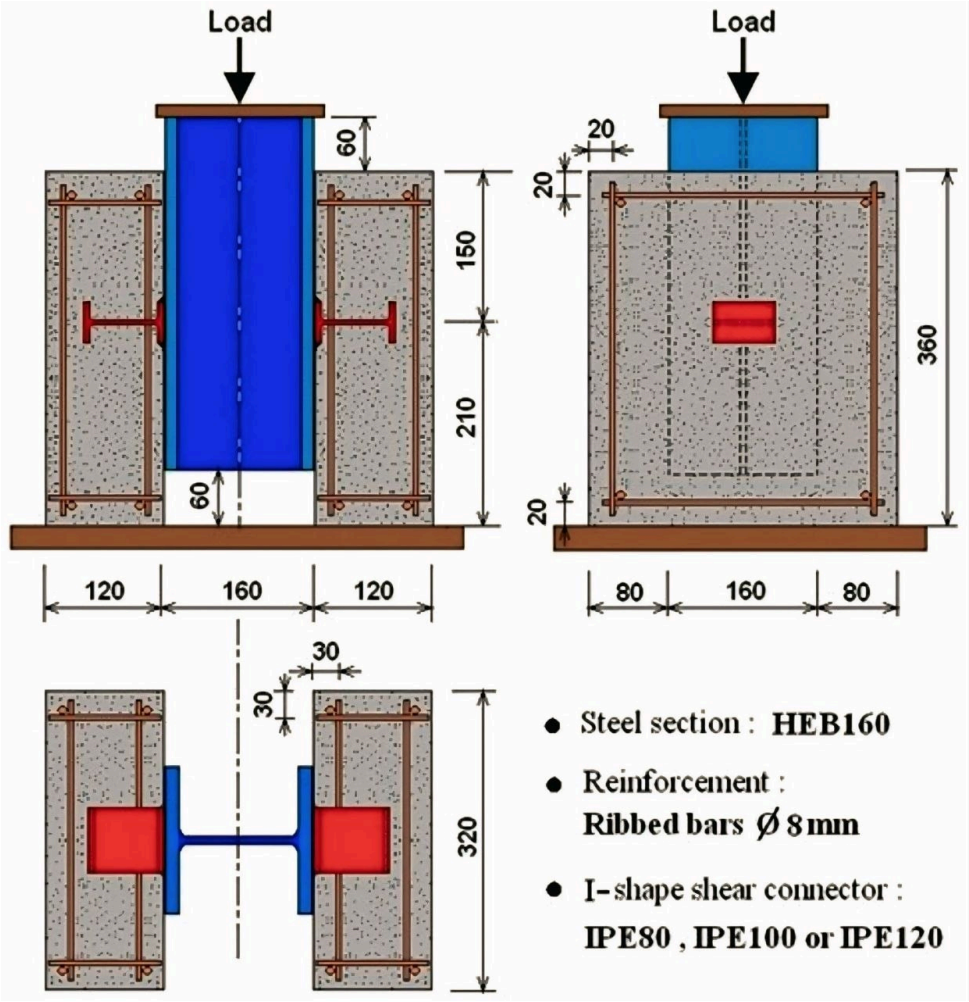


Figure 2.6 : Échantillon de poussée avec connecteur en forme de I [30]

- **Comportement au cisaillement de nouveaux connecteurs en C améliorés dans des structures composites sandwich acier–béton–acier (Jia-Bao Yan et al, 2020)**

L'étude de Yan et al a analysé des connecteurs en C améliorés avec boulons externes dans des structures acier-béton sous charges de cisaillement. Les résultats montrent une meilleure résistance et une capacité de charge plus élevée que les connecteurs traditionnels. Les modèles analytiques ont également bien prédit leur comportement.[31]

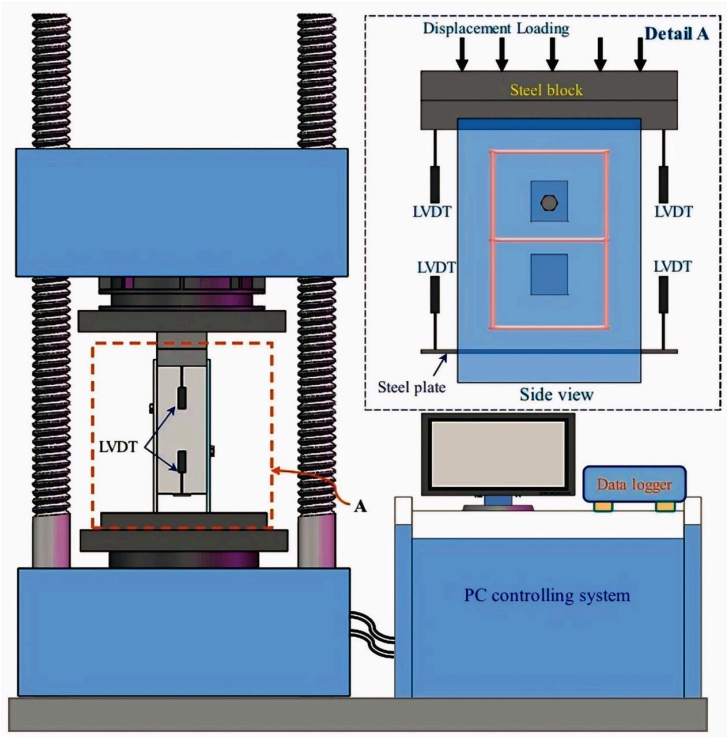


Figure 2.7 : Dispositif d'essai de poussée
[31]

- **Comportement monotone des connecteurs de cisaillement en angle en forme de C et L dans les poutres composites en acier et béton : une enquête expérimentale (Mahdi Shariati et al, 2020)**

Cette recherche s'appuie sur des connecteurs de cisaillement en cornière de deux formes : C-shaped et L-shaped, intégrés dans des poutres mixtes acier-béton. L'objectif était d'analyser leur comportement mécanique sous chargement monotone croissant via des essais de poussées, en étudiant les courbes charge–glissement. Les résultats ont montré des ruptures dominées par l'écrasement/fendage du béton ou par la rupture du connecteur. Il a été confirmé que les connecteurs en C offrent une résistance au cisaillement plus élevée que ceux en L, donc plus performants pour la conception.[32]

- **Évaluation expérimentale de nouvelles formes de connecteurs en cisaillement en U (Bilal Maghaghi et al, 2021)**

Cette étude examine le comportement de connecteurs de cisaillement en U modifiés dans des éléments mixtes acier–béton à travers des essais de poussées.

Les résultats montrent une amélioration de la ductilité et une réduction de la fissuration. Le connecteur CH6 a présenté les meilleures performances, avec une augmentation de la ductilité atteignant 31% et une légère hausse de la résistance globale.[33]

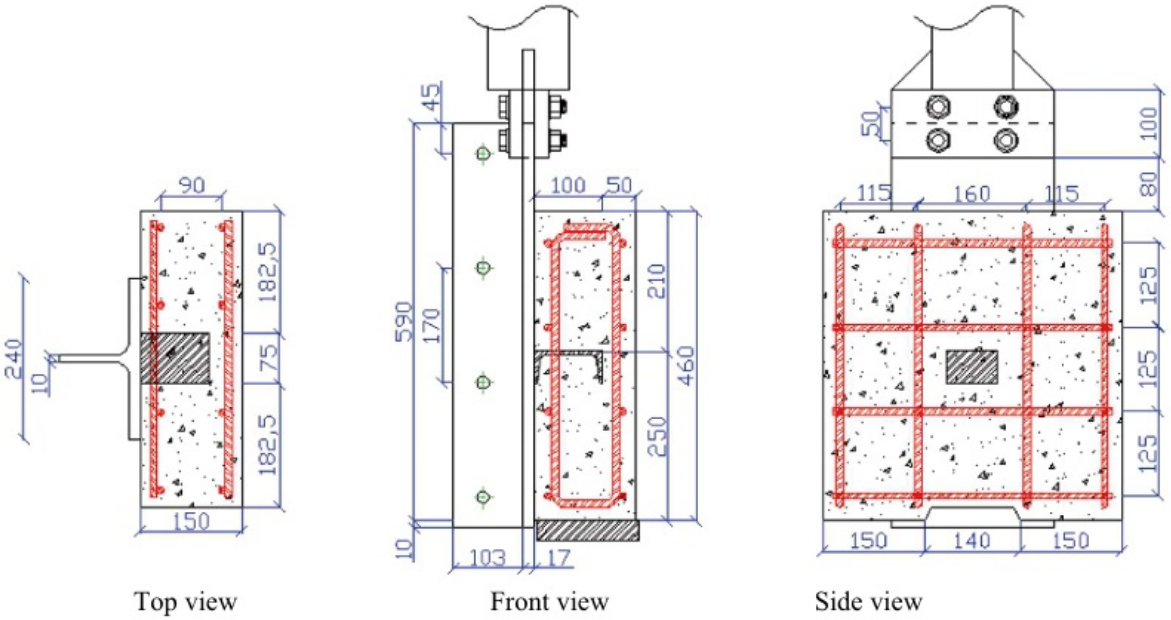


Figure 2.8 : Détails de l'échantillon de test de poussée [33]

- **Une approche novatrice pour prédire la résistance au cisaillement des connecteurs à angle incliné en utilisant des techniques d'intelligence artificielle(Mahdi Shariati et al, 2021)**

Cette étude utilise l'intelligence artificielle pour prédire la résistance au cisaillement des conducteurs inclinés. Le glissement et l'angle d'inclinaison sont les deux facteurs les plus influents. Ainsi, le modèle ELM est le plus efficace.[34]

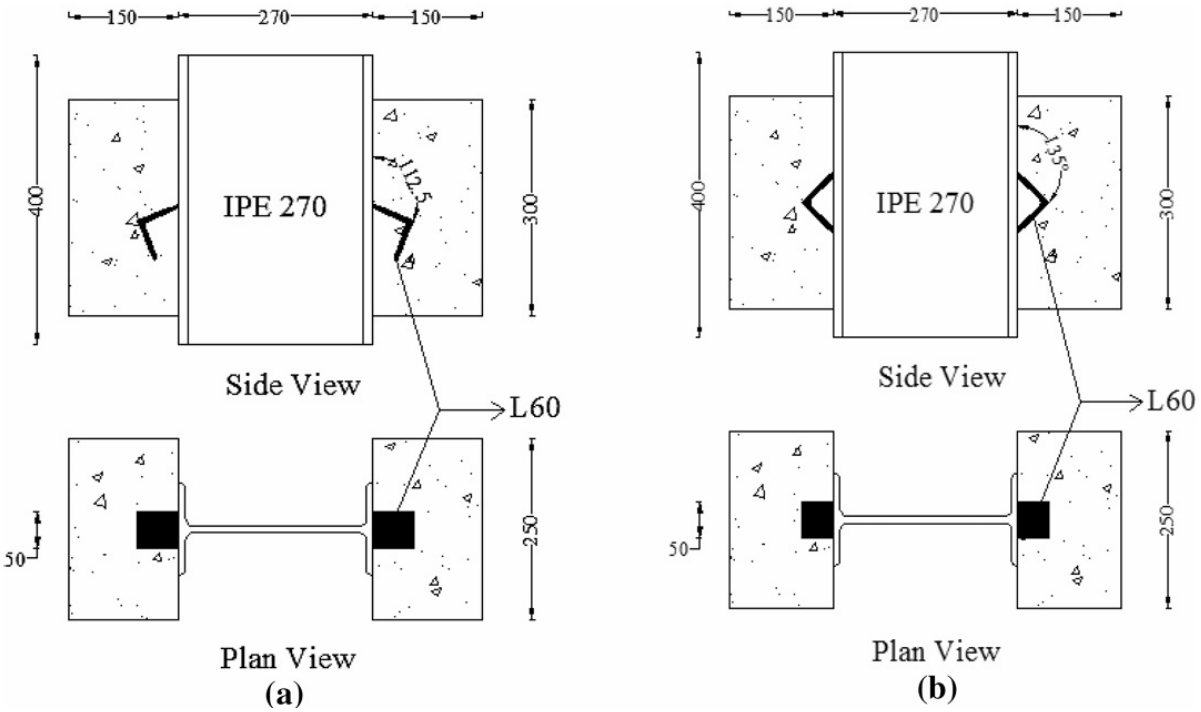


Figure 2.9 : Présentation schématique des connecteurs à angles inclinés avec un degré d'inclinaison de : a 122,5° ; b 135°

[34]

- **Étude paramétrique des connecteurs en cisaillement en forme de I avec différentes orientations dans le test de poussée (Farid Boursas et al, 2021)**

Cette étude expérimentale analyse le comportement de connecteurs de cisaillement de type I sous essais de poussées. Les résultats montrent que leur orientation influence fortement la résistance et le glissement, avec une configuration optimale offrant de meilleures performances. La rupture est dominée par l'écrasement du béton, tandis que le connecteur présente un comportement ductile.[35]



Figure 2.10 : Échantillons de poussée
[35]

- **Performance structurelle des connecteurs de cisaillement en U et en angle dans le béton à granulats de brique(Md Rabiul Alam et al, 2023)**

L'étude porte sur des connecteurs de cisaillement en U et angle dans un béton à agrégats de briques sous essai de poussée. Les résultats montrent la supériorité des connecteurs en U en résistance au cisaillement avec un comportement ductile, et que l'augmentation de la longueur améliore la résistance ainsi que l'absorption d'énergie.[36]

Tableau 2.3 : Propriétés géométriques des sections en acier utilisées dans l'étude [36]

Désignation	**Propriétés**					
	Poids (kg/m)	**Aire** (mm^2)	**Épaisseur de l'âme** t_w (mm)	**Hauteur de l'âme** h_w (mm)	**Épaisseur de la semelle** t_f (mm)	**Largeur de la semelle** b_f (mm)
W4X13	19.3461	2470.963	7.112	105.664	8.763	103.124
C2×6	8.93	763.8694	8.763	50.8	6.5	31.75
C3×6	8.93	1135.482	9.0424	76.2	6.9342	40.64

Désignation	**Propriétés**				
	Poids (kg/m)	**Aire** (mm^2)	**Hauteur** h (mm)	**Largeur** w (mm)	**Épaisseur** t (mm)
L2×2×0.5	9.27	1148.38	50.8	50.8	12.7
L3×3×0.5	13.9887	1780.642	76.2	76.2	12.7

- **Performance des connecteurs de cisaillement en U dans une structure en acier avec des murs de remplissage en béton armé(Jiali Yu et al, 2024)**

Des connecteurs en U en structures SRCW ont été étudiés sous chargements monotones et cycliques. Les résultats montrent une baisse de 20 % de la capacité portante et de 40 % du glissement ultime en chargement cyclique, avec une formulation analytique améliorée plus précise que les modèles normatifs.[37]

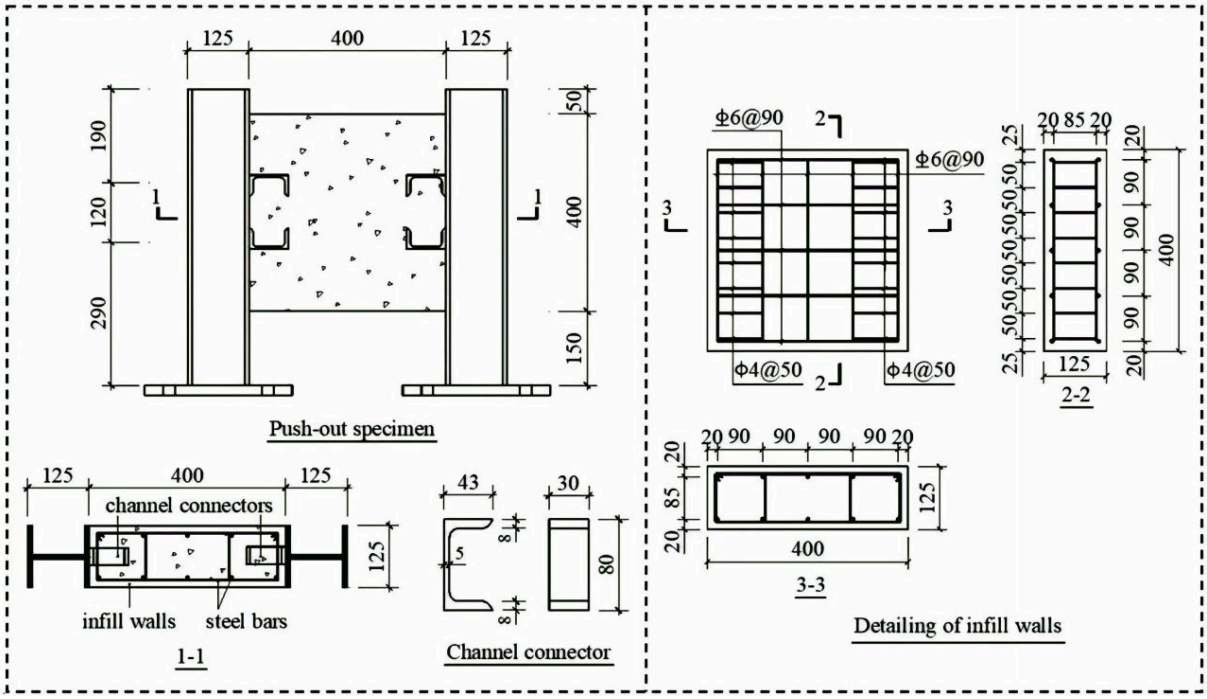


Figure 2.11 : Géométrie et détails des échantillons de test [37]

- **Comparaison du connecteur en forme de I avec des canaux et des goujons à tête : étude expérimentale et analyse par éléments finis(Aida Mazoz et al, 2024)**

L'étude a utilisé un connecteur de cisaillement I-shape pour évaluer son efficacité dans la liaison acier–béton. Les résultats montrent une résistance et une rigidité proches du connecteur en U et supérieures au goujon, ce qui confirme son efficacité pour les structures composites.[38]

- **Enquête expérimentale sur les connecteurs de cisaillement en U dans les dalles composites à caisson en acier(H.C.A. Tavares et al, 2025)**

Des connecteurs channel en dalles composites avec tôle profilée ont été étudiés par essais e poussées. Les résultats montrent une réduction de la résistance et de la ductilité en présence du steel deck, avec rupture par cône de béton, tandis que les armatures et l'augmentation de la hauteur améliorent la capacité portante et le glissement ultime.[39]

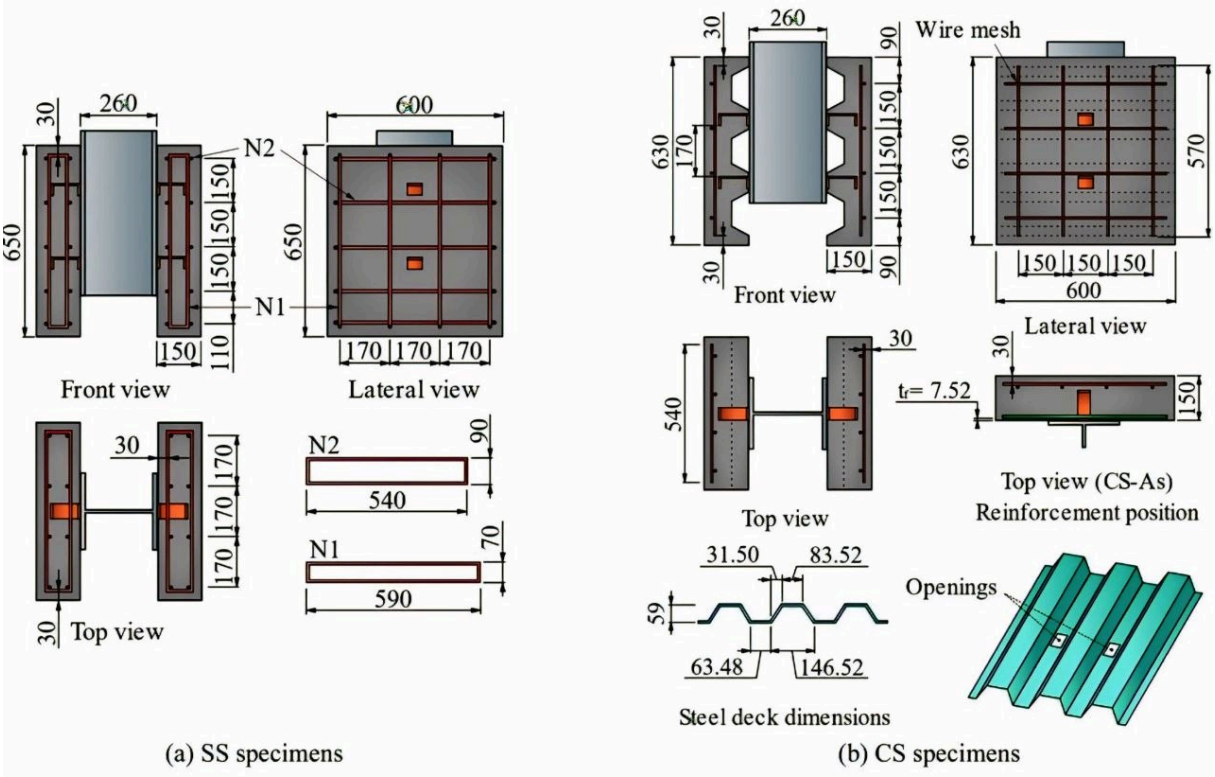


Figure 2.12 : Dimensions géométriques des échantillons, en mm [39]

- **Performance au cisaillement des poutres composites acier-béton en forme de U avec des connecteurs de cisaillement en U(S.Y.Qiu et al, 2025)**

Des connecteurs en U ont été étudiés dans des poutres composites acier–béton afin d'analyser leur comportement au cisaillement. Les résultats montrent un bon comportement global avec un glissement limité et une influence notable des propriétés du béton et de la géométrie. Une nouvelle équation de résistance a été proposée et validée par une bonne concordance expérimentale.[40]

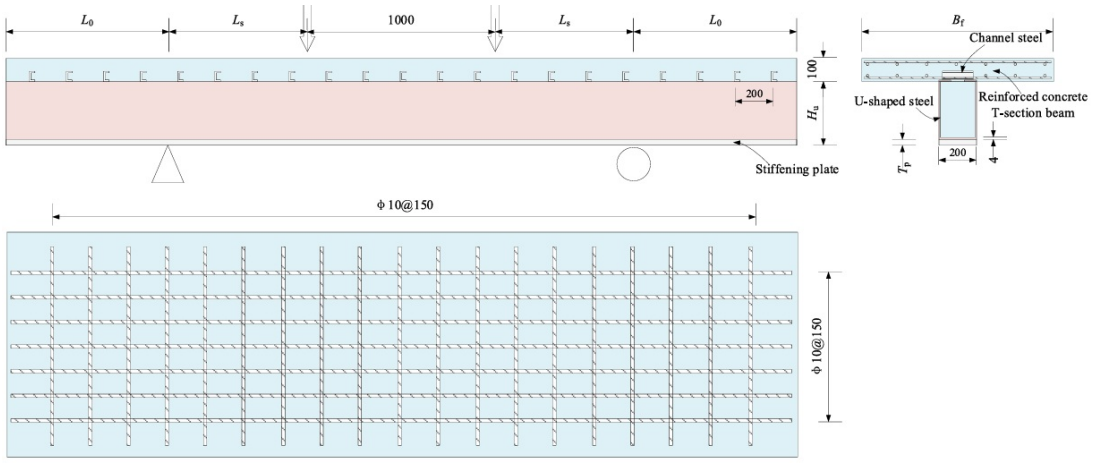


Figure 2.13 : Configuration structurelle et détails de chargement des échantillons (mm) [40]

- **Prédiction du comportement charge-glissement dans des poutres composites avec des configurations variées en utilisant une approche expérimentale, numérique et d'apprentissage automatique (P. Sangeetha et al, 2025)**

L'étude repose sur l'utilisation des connecteurs de cisaillement en U afin d'analyser le comportement charge–glissement. Les résultats obtenus montrent une amélioration significative du transfert de cisaillement, en particulier avec l'utilisation de tôles à embossments, qui renforcent l'interaction acier–béton. De plus, le profil trapézoïdal s'est révélé le plus performant en termes de résistance et d'efficacité.[41]

2.3.2 Études numériques(FEM)

- **Analyse numérique des connecteurs en U sous feu et comparaison des performances avec différents types de connecteurs de cisaillement soumis au feu(S.E.M. Shahabi et al (2015))**

L'étude repose sur une analyse numérique des connecteurs de cisaillement en U intégrés dans un béton à haute résistance et soumis au feu. Les résultats indiquent une diminution significative de la capacité portante avec l'élévation de la température, le connecteur atteignant 100–150 °C de plus que le béton environnant. Le mode de rupture évolue vers un renversement plutôt qu'un cisaille-

ment classique, tout en conservant une performance structurelle et un avantage économique.[42]

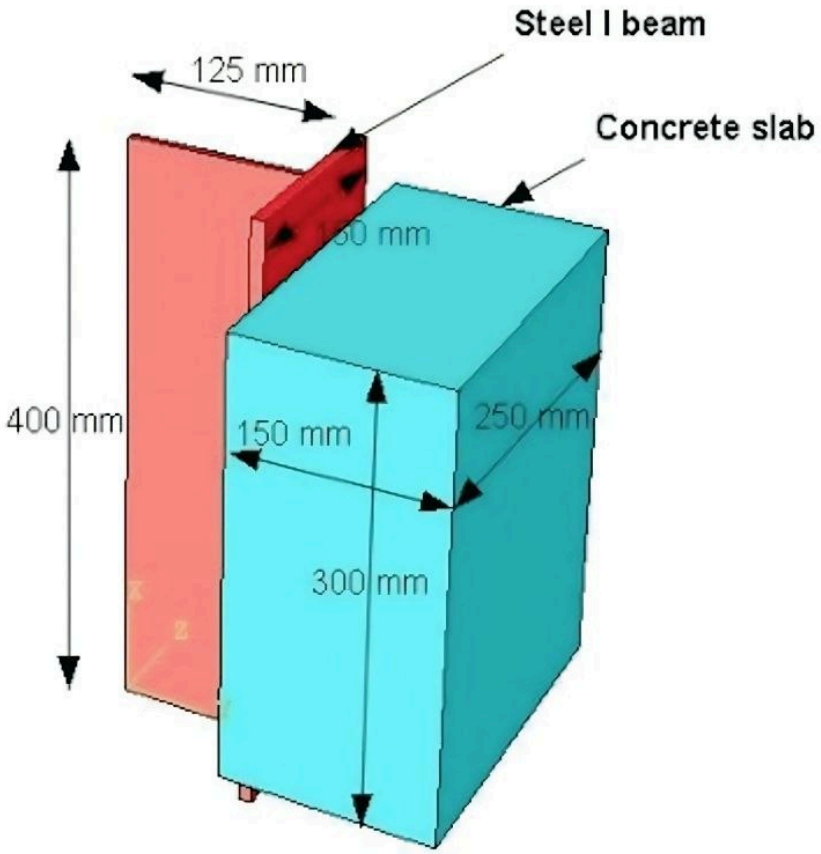


Figure 2.14 : Modèle géométrique de test de poussée
[42]

- **Capacité de cisaillement des connecteurs de cisaillement en angle en forme de C et en forme de L (Farzad Tahmasbi1 et al (2016))**

Cette étude numérique modélise les connecteurs de cisaillement de type C-shaped et L-shaped sous ABAQUS afin de simuler l'essai de poussée. L'objectif est d'analyser le comportement non linéaire du béton et de l'acier et d'évaluer la résistance au cisaillement. Les résultats montrent une bonne concordance avec les essais expérimentaux et confirment que l'augmentation de la longueur améliore la capacité portante. Une nouvelle équation plus précise est également proposée pour les connecteurs C-shaped.[43]

- **Équation de capacité de cisaillement pour les connecteurs de cisaillement en U dans les poutres composites acier-béton (M.Paknahad et al (2018))**

L'étude numérique a modélisé des connecteurs en U (CSC) dans un béton à haute résistance par FEM non linéaire sous ABAQUS. L'objectif était d'étudier l'effet des dimensions et de la résistance du béton et de comparer avec le code NBC. Les résultats ont montré que NBC est peu précis pour le HSC, donc une équation modifiée a été proposée.[44]

- **Étude paramétrique des connecteurs en cisaillement en forme de I avec différentes orientations dans le test de poussée (Farid Boursas et al, 2021)**

L'étude numérique a modélisé les connecteurs de cisaillement de type I-shaped par une analyse tridimensionnelle par éléments finis (ABAQUS) afin d'étudier l'effet de l'orientation. Les résultats ont montré que la configuration P03 offre la meilleure résistance. De plus, l'augmentation de la résistance du béton améliore la capacité ultime mais réduit la ductilité.[35]

- **Performance des connecteurs de cisaillement en U dans une structure en acier avec des murs de remplissage en béton armé(Jiali Yu et al, 2024)**

Cette étude numérique porte sur un connecteur de cisaillement en U intégré dans une paroi SRCW modélisée par éléments finis 3D. L'objectif était d'évaluer la résistance au cisaillement et le glissement du système. Les résultats montrent que la capacité dépend principalement de la résistance du béton et des dimensions du connecteur, tandis qu'une nouvelle équation a été proposée pour améliorer la précision par rapport aux codes existants.[37]

- **Enquête numérique sur les connecteurs de cisaillement en U dans les dalles composites à caisson en acier(H.C.A. Tavares et al, 2025)**

L'étude numérique a porté sur un connecteur de cisaillement en U dans une dalle composite avec bac acier, modélisée sous ABAQUS/Explicit. L'objectif était d'évaluer l'effet du bac acier, du frottement et des paramètres géométriques sur la résistance. Les résultats montrent que le frottement dalle–appui est le facteur principal, que l'augmentation de la hauteur du connecteur améliore la résistance et la ductilité, tandis que le bac acier réduit la capacité par rapport à une dalle pleine.[39]

- **Performance au cisaillement des poutres composites acier-béton en forme de U avec des connecteurs de cisaillement en U(S.Y.Qiu et al, 2025)**

Une analyse numérique par FEM a été menée sur une poutre mixte acier-béton en U munie de connecteurs en U afin d'évaluer sa capacité au cisaillement. Les résultats montrent une augmentation de la résistance avec la hauteur de section et la résistance du béton, et une diminution avec l'augmentation du rapport de cisaillement. Une nouvelle expression de calcul a été proposée avec un écart limité à $\pm 15\%$.[40]

- **Prédiction du comportement charge-glissement dans des poutres composites avec des configurations variées en utilisant une approche expérimentale, numérique et d'apprentissage automatique (P. Sangeetha et al, 2025)**

Une analyse numérique a été réalisée sur des poutres composites acier–béton avec tôles profilées et connecteurs de cisaillement en U, afin d'étudier la relation charge–glissement. Un modèle par éléments finis sous ANSYS a permis d'évaluer l'influence de la géométrie du profil et des embossments. Les résultats montrent que les embossments améliorent l'action composite en augmentant la résistance et en réduisant le glissement. Le profil trapézoïdal a donné les meilleures performances globales.[41]

2.3.3 Modèles analytiques existants

- **Comportement des connecteurs de cisaillement en U(S.Maleki, S.Bagheri (2008))**

L'étude analyse des connecteurs de cisaillement en U via un modèle EF 3D simulant l'essai de poussée. L'objectif est de prédire la résistance ultime et la relation cisaillement–glissement selon les paramètres géométriques et du béton. Les résultats montrent une augmentation de résistance avec la résistance du béton, l'épaisseur et la longueur du connecteur, tandis que la hauteur a peu d'influence. L'inversion de l'orientation réduit la résistance et la rigidité d'environ 16 %.[45]

- **Comportement au cisaillement de nouveaux connecteurs en C améliorés dans des structures composites sandwich acier–béton–acier (Jia-Bao Yan et al, 2020)**

L'étude analyse des connecteurs de cisaillement de type **C amélioré (EC)** dans des structures acier–béton–acier afin d'améliorer la résistance au cisaillement et de limiter le glissement. Les résultats montrent que les boulons externes augmentent la résistance ultime par rapport aux connecteurs en C classiques, avec une rupture progressive (cisaillement des boulons puis du profilé). Un modèle analytique a également été proposé pour prédire la résistance et la relation cisaillement–glissement.[31]

- **Comparaison du connecteur en forme de I avec des canaux et des goujons à tête : étude expérimentale et analyse par éléments finis(Aida Mazoz et al, 2024)**

L'étude compare le connecteur de cisaillement en I avec le connecteur à profilé en C et le Headed Stud afin d'évaluer leur résistance au cisaillement, leur rigidité et leur glissement. Les résultats montrent que le connecteur en I présente des performances proches de celles du connecteur à profilé en C et supérieures à celles du Headed Stud en termes de résistance au cisaillement et de rigidité, tandis que le Headed Stud offre une meilleure ductilité. Le connecteur en I satisfait néanmoins aux exigences de ductilité de l'Eurocode 4.[38]

2.4 Paramètres influençant la résistance au cisaillement

La résistance au cisaillement des connecteurs en U dans les poutres mixtes acier–béton dépend principalement des caractéristiques du béton et de la géométrie du connecteur.

- **Caractéristiques du béton**

Le béton entourant le connecteur influence directement la résistance au cisaillement, avec deux modes principaux de rupture : l'écrasement local au niveau de l'âme du connecteur et la fissuration accompagnée de l'éclatement de la dalle.

La capacité portante dépend fortement de la résistance à la compression, de la présence d'armatures transversales et du type de béton utilisé. L'ajout d'armatures améliore sensiblement la résistance et la ductilité, tandis que l'emploi d'un béton léger non armé entraîne un comportement plus fragile et une réduction de capacité pouvant atteindre 20 % par rapport au béton ordinaire.[25][24]

- **Géométrie et matériau du connecteur**

La géométrie du connecteur en U influence fortement la résistance au cisaillement, notamment via la hauteur H, la longueur L_c et l'épaisseur t_w et t_f. L'augmentation de ces paramètres améliore la capacité portante, mais de façon non linéaire avec un rendement décroissant pour les grandes longueurs.

Par exemple, pour un connecteur UPN65, passer de 50 à 75 mm augmente la résistance d'environ 34 %, tandis que l'augmentation de 75 à 100 mm ne donne qu'environ 10 % de gain supplémentaire.[26]

Ainsi, la résistance au cisaillement des connecteurs en U résulte essentiellement de l'interaction entre les propriétés mécaniques du béton et les caractéristiques géométriques du connecteur.

2.5 Limites des modèles existants

- **Approches empiriques**

Les formules proposées pour l'évaluation de la résistance des connecteurs en U reposent essentiellement sur des corrélations empiriques établies à partir d'un nombre limité d'essais expérimentaux.

La relation proposée par l'AISC (2005) est donnée par :

$$Q_n = 0.3\,(t_f + 0.5\,t_w)\,L_c\,\sqrt{f'_c \cdot E_c}$$

Elle a été développée à partir des résultats de 41 éprouvettes testées à l'Université de Lehigh.

Les comparaisons entre les résultats expérimentaux et les prédictions issues des règlements montrent que cette formulation est globalement conservative. En effet, le rapport entre les valeurs prédites et mesurées présente une moyenne de 0.77 selon l'AISC et de 0.55 selon la norme canadienne CSA, avec des valeurs minimales respectives de 0.57 et 0.41.[26][24]

- **Manque de généralisation**

Les modèles existants de résistance des connecteurs en U présentent plusieurs limitations. Ils ne prennent pas en compte l'interaction complète des paramètres influents, notamment la hauteur du connecteur, le type de béton, le ferraillage et les effets de chargement cyclique. Leur domaine de validité est également restreint, car basé principalement sur des connecteurs nord-américains de hauteur fixe (100 mm), ce qui limite leur application aux profils européens UPN (65 à 140 mm). Enfin, le manque de données expérimentales sur les connecteurs en U, comparativement aux connecteurs à goujons, réduit la robustesse statistique et la généralisation des modèles proposés.[26][25]

2.6 Construction du dataset

Cette section présente de façon détaillée la méthodologie adoptée pour l'élaboration de la base de données expérimentale utilisée dans cette étude. La trans-

parence sur les sources, les critères de sélection et le protocole de nettoyage est indispensable pour assurer la reproductibilité des résultats et la fiabilité des modèles de Machine Learning développés.

2.6.1 Sources des données

Le dataset regroupe 618 essais de poussée compilés à partir de 23 références scientifiques publiées entre 2006 et 2025 dans des revues internationales. Toutes ces études portent sur des connecteurs de cisaillement (U/C) intégrés dans des structures mixtes acier–béton.

Tableau 2.4 : Récapitulatif des 23 sources du dataset (618 observations)

#	**Auteurs (Année)**	**Revue / Source**	**n**
1	S. Maleki & S. Bagheri (2008) — Part I	JCSR	16
2	S. Maleki & S. Bagheri (2008) — Part II	JCSR	17
3	Mahdi Shariati et al. (2011)	QUT ePrints	5
4	Mahdi Shariati et al. (2013)	Constr. Build. Mater.	16
5	Mahdi Shariati et al. (2020)	Steel Compos. Struct.	8
6	Mahdi Shariati et al. (2021)	Engineering with Computers	8
7	Jia-Bao Yan et al. (2020)	JCSR	16
8	J. Yu et al. (2024)	Structures	72
9	H.C.A. Tavares et al. (2025)	Structures	10
10	S.-Y. Qiu et al. (2025)	Structures	6
11	Md Rabiul Alam et al. (2023)	ResearchGate	8
12	Amit Pashan (2006)	Univ. Saskatchewan	78
13	Eray Baran & C. Topkaya (2012)	JCSR	15
14	Ali Toghroli et al. (2014)	ResearchGate	16
15	Farzad Tahmasbi et al. (2016)	PLOS ONE	8
16	Masoud Paknahad et al. (2018)	Steel Compos. Struct.	44
17	S.E.M. Shahabi et al. (2015)	Steel Compos. Struct.	12
18	Bilal Maghaghi et al. (2021)	Int. J. Steel Struct.	11
19	F. Boursas et al. (2021)	ResearchGate	76
20	Messaoud Titoum et al. (2016)	Adv. Steel Constr.	96
21	Aida Mazoz et al. (2013)	Int. J. Steel Struct.	48
22	Aida Mazoz et al. (2024)	Asian J. Civil Eng.	24
23	P. Sangeetha et al. (2025)	Asian J. Civil Eng.	9
		Total	**618**

Note : f_c = résistance en compression du béton ; H = hauteur du connecteur ; L_c = longueur du connecteur ; Q_u = résistance ultime au cisaillement par connecteur. Les références 1 et 2 (Maleki & Bagheri, 2008) correspondent aux Parts I et II d'une même campagne expérimentale.

Les connecteurs étudiés sont exclusivement des connecteurs en U soudés sur une poutre métallique et ancrés dans une dalle en béton. Toutes les données expérimentales retenues dans cette étude concernent ce type de connecteur, afin d'assurer l'homogénéité de la base de données et la fiabilité des modèles développés.

2.6.2 Description des variables

Le dataset comprend 12 variables d'entrée couvrant la géométrie de la dalle, les propriétés mécaniques des matériaux et la géométrie du connecteur, ainsi qu'une variable de sortie correspondant à la résistance ultime au cisaillement Q_u.

Le cisaillement étudié est un cisaillement longitudinal, correspondant au mode de transfert des efforts entre la poutre en acier et la dalle en béton dans les structures mixtes.

Tableau 2.5 : Description et statistiques descriptives des 13 variables du dataset

#	Variable	Symb.	Unité	Rôle
1	Largeur de la dalle	–	mm	Entrée — géométrie dalle
2	Épaisseur de la dalle	–	mm	Entrée — géométrie dalle
3	Longueur de la dalle	–	mm	Entrée — géométrie dalle
4	Armature de la dalle	–	mm²	Entrée — renforcement
5	Résistance du béton	f_c	MPa	Entrée — matériau béton
6	Hauteur du connecteur	H	mm	Entrée — géométrie connecteur
7	Largeur du connecteur	b_f	mm	Entrée — géométrie connecteur
8	Épaisseur de l'âme	t_w	mm	Entrée — géométrie connecteur
9	Épaisseur de la semelle	t_f	mm	Entrée — géométrie connecteur
10	Longueur du connecteur	L_c	mm	Entrée — géométrie connecteur
11	Limite élastique acier	f_y	MPa	Entrée — matériau acier
12	Espacement des connecteurs	s	mm	Entrée — disposition
13	Résistance au cisaillement	Q_u	kN	Sortie — variable cible

2.6.3 Organisation du dataset

Les données sont structurées sous forme d'un tableau de 618×13, où chaque ligne représente un essai push-out indépendant et chaque colonne une variable. Cette organisation tabulaire facilite directement l'analyse statistique et l'entraînement des modèles de Machine Learning sans transformation structurelle préalable.

Répartition par plage de résistance du béton :

- $f_c < 30$ MPa $\rightarrow$ ≈310 observations (50 %) ;
- $30 \leq f_c < 50$ MPa $\rightarrow$ ≈245 observations (40 %) ;
- $f_c \geq 50$ MPa $\rightarrow$ ≈63 observations (10 %).

Cette distribution reflète la prédominance des bétons ordinaires dans la littérature expérimentale disponible sur les connecteurs de type channel.

2.6.4 Critères d'inclusion et d'exclusion

L'intégration de chaque observation dans le dataset a obéi à des critères stricts visant à garantir l'homogénéité du jeu de données et la pertinence des modèles prédictifs.

Tableau 2.6 : Critères d'inclusion et d'exclusion appliqués à la construction du dataset

Critère	✓ **Inclusion**	× **Exclusion**
Type de connecteur	Profil en acier laminé à chaud (U/C/T/I)	Goujons (studs), cornières
Type de chargement	Chargement monotone statique (tests de poussées)	Chargement cyclique, dynamique ou essais à haute température
Type de béton	Béton ordinaire ou à haute résistance avec f_c renseigné	Béton polymère, géopolymère ou sans mesure de f_c
Complétude	Toutes les variables présentes : H, b_f, t_w, t_f, L_c, f_c, f_y, Q_u	Toute observation avec ≥ 1 variable manquante
Source	Publications revues par les pairs ou thèses universitaires reconnues	Données non publiées ou sans résultats numériques complets
Cohérence physique	Valeurs dans des plages physiquement réalistes ($Q_u > 0$, $f_c > 0$, $H > 0$)	Valeurs négatives, nulles ou manifestement erronées

Les essais de poussée utilisés dans cette étude sont soumis à un chargement uniaxial de cisaillement. Ce type de chargement est le plus couramment adopté dans les études expérimentales sur les connecteurs de cisaillement et permet d'évaluer leur résistance ultime dans des conditions représentatives du comportement des structures mixtes acier–béton.

Remarque : les essais sous chargement cyclique ont été intentionnellement exclus, la résistance au cisaillement sous ce type de chargement pouvant être inférieure de 10 à 49 % à celle obtenue sous chargement monotone. Leur inclusion introduirait un biais systématique dans la modélisation de la résistance statique, qui constitue la variable cible de cette étude.

2.7 Conclusion

L'étude montre que la performance des structures mixtes dépend fortement de l'efficacité des connecteurs de cisaillement, en particulier des connecteurs en forme de U. Son comportement est influencé par plusieurs critères géométriques et mécaniques ainsi que par les conditions de chargement. Malgré les avancées expérimentales et numériques, les modèles analytiques restent limités en précision et en généralisation. Ces limites justifient l'utilisation de bases de données expérimentales pour développer des modèles prédictifs plus fiables.

Chapitre 03

Chapitre 3

Programmation du ML en python

3.1 Introduction au notebook et environnement de travail

Cette étude a été réalisée sous **Python** à l'aide de **Jupyter Notebook**, permettant une exécution interactive et progressive du **pipeline** de **Machine Learning**, depuis le traitement des données jusqu'à l'interprétation des résultats.

La base de données utilisée provient d'un fichier Excel contenant 618 essais et 13 variables décrivant les caractéristiques des connecteurs en U dans les structures mixtes acier–béton. La variable cible est la résistance au cisaillement, tandis que les variables explicatives regroupent les propriétés des matériaux et les paramètres géométriques.

Le pipeline repose sur des bibliothèques Python essentielles :

- **NumPy** : est une librairie open-source pour Python, facilitant la programmation en mathématiques, en sciences, en ingénierie et en science des données. Il s'agit d'une bibliothèque très pratique pour réaliser des opérations mathématiques et statistiques avec Python.[46]
- **Pandas** : est une bibliothèque Python dédiée à l'analyse et au traitement des données. Créée par Wes McKinney en 2008, elle met à disposition des structures de données performantes, notamment Series et DataFrame, facilitant le chargement, la manipulation, la préparation et l'analyse efficace des données.[47]
- **Matplotlib** : est une bibliothèque de représentation graphique très utilisée en Python. Elle permet la visualisation de données et s'intègre facilement

avec NumPy, Pandas et d'autres bibliothèques associées.[48]

- **Seaborn** : est une bibliothèque de représentation graphique basée sur Matplotlib. Elle facilite la visualisation des données et est particulièrement adaptée à la manipulation des DataFrames Pandas.[48]
- **Scikit-learn** :est une bibliothèque Python de apprentissage automatique qui permet de créer et d'évaluer des modèles, avec des outils pour préparer les données et optimiser les performances.[49]
- **XGBoost** :Nous avons présenté cet algorithme en détail dans le premier chapitre.

Après une analyse descriptive et exploratoire (**EDA**), les données ont été nettoyées, analysées et divisées en ensembles d'entraînement et de test avec standardisation.

L'objectif principal est de construire un modèle prédictif performant, comparer plusieurs algorithmes, optimiser leurs performances et interpréter les résultats de manière fiable.

Ce chapitre présente la mise en œuvre d'un pipeline de apprentissage automatique en Python pour la prédiction de la résistance au cisaillement des connecteurs en U. Il couvre l'exploration des données, l'optimisation de cinq modèles de régression (RF, XGBoost, SVR, MLP et k-NN) via **GridSearchCV**, ainsi que la construction d'un méta-modèle de stacking basé sur la régression Ridge.

3.2 Chargement et exploration des données

3.2.1 Importation du dataset

L'étude s'appuie sur une base de données constituée à partir d'une compilation expérimentale de la littérature scientifique, qui rassemble des essais de cisaillement réalisés sur des connecteurs en U dans des structures mixtes acier–béton. Cette base de données est importée sous Python à partir d'un fichier Excel à l'aide de la bibliothèque `Pandas`, avec un processus systématique de norma-

lisation des noms de colonnes afin de supprimer les espaces et les caractères de retour à la ligne susceptibles de provoquer des erreurs lors de l'indexation.

3.2.2 Vérification de la structure des données

Le dataset contient 618 observations et 13 colonnes, dont 12 variables explicatives (**features**) et 1 variable cible (**target**).

Les colonnes sont présentées comme suit :

Tableau 3.1 : Description des variables du dataset

	Variable	**Rôle**
1	Largeur de la dalle (mm)	Variable explicative
2	Épaisseur de la dalle (mm)	Variable explicative
3	Armature (mm^2)	Variable explicative
4	Longueur de la dalle (mm)	Variable explicative
5	Résistance du béton de la dalle f_c (MPa)	Variable explicative
6	Hauteur du connecteur (mm)	Variable explicative
7	Largeur du connecteur (mm)	Variable explicative
8	Épaisseur de l'âme du connecteur (mm)	Variable explicative
9	Épaisseur de la semelle du connecteur (mm)	Variable explicative
10	Longueur du connecteur en U (mm)	Variable explicative
11	Nuance d'acier du connecteur (MPa)	Variable explicative
12	Espacement des connecteurs (mm)	Variable explicative
13	Résistance au cisaillement par connecteur (kN)	Variable cible

3.2.3 Analyse des dimensions et variables

Les 12 variables du dataset se regroupent en trois catégories physiques, comme présenté dans le tableau suivant :

Tableau 3.2 : Classification des variables du jeu de données

Catégorie	**Variables**
Dalle en béton	Largeur, Épaisseur, Armature, Longueur, Résistance du béton f_c
Connecteur en profilé U	Hauteur, Largeur, Épaisseur de l'âme, Épaisseur des ailes, Longueur, Nuance d'acier, Espacement
Variable cible	Résistance au cisaillement par connecteur (kN)

3.2.4 Vérification des types de données

Les types de données des variables du dataset sont résumés comme suit :

Tableau 3.3 : Types des variables

Variable	**Type**
Largeur de la dalle (mm)	int64
Espacement des connecteurs (mm)	int64
Toutes les autres variables	float64

3.2.5 Détection des valeurs manquantes

Le résultat obtenu montre qu'aucune valeur manquante n'est présente dans l'ensemble du dataset. Ainsi, le jeu de données est complet et aucune opération d'imputation n'est nécessaire, ce qui constitue un avantage majeur pour la fiabilité du processus de modélisation.

3.2.6 Analyse statistique descriptive initiale

Les statistiques clés sont résumées dans le tableau suivant :

Tableau 3.4 : Statistiques descriptives des variables du jeu de données

Variable	**Min**	**Moyenne**	**Médiane**	**Max**	**Écart-type**
Largeur de la dalle (mm)	150	348.9	320	1100	112.8
Épaisseur de la dalle (mm)	100	135.2	125	250	26.9
Armature (mm^2)	0	545.1	402.1	2042	342.4
Longueur de la dalle (mm)	300	417.1	360	1000	154.7
Résistance à la compression du béton f_c (MPa)	19.44	32.1	30	82	14.4
Hauteur du connecteur (mm)	40	94.3	100	140	21.7
Largeur du connecteur (mm)	20	49.3	50	100	10.7
Épaisseur de l'âme (mm)	2	5.2	4.8	12.7	1.6
Épaisseur des ailes (mm)	2	6.8	6.3	12.7	1.6
Longueur du connecteur (mm)	20	58.4	60	152.4	26.4
Limite d'élasticité de l'acier (MPa)	220	288	235	527	82.4
Espacement des connecteurs (mm)	/	41.1	/	250	66.3
Résistance au cisaillement (kN)	36	163.2	103.5	1700	156.7

3.2.7 Vérification de la cohérence physique des variables

Les variables présentent uniquement des valeurs positives ou nulles, ce qui est cohérent physiquement. On note 21 cas avec armature nulle et 437 avec es-

pacement nul, correspondant à des configurations particulières à vérifier expérimentalement. Par ailleurs, 177 lignes dupliquées (28,6 %) ont été détectées. L'analyse statistique approfondie de la distribution de la variable cible et de ses asymétries est présentée dans le Chapitre 04.

3.3 Analyse exploratoire des données (EDA) — 4 visualisations principales

3.3.1 Matrice de Corrélation (Heatmap)

La matrice de corrélation de Pearson sera calculée pour identifier les variables les plus influentes. Les résultats détaillés sont présentés au chapitre 4 (Tab. 4.1)

3.3.2 Distribution Statistique des Variables (Histogrammes)

La majorité des variables présente une distribution asymétrique à droite, indiquant une concentration des données autour de valeurs faibles avec quelques valeurs élevées. Les asymétries les plus marquées concernent la résistance cible , la résistance du béton et la longueur du connecteur .

À l'inverse, l'épaisseur de semelle et la hauteur du connecteur présentent des distributions proches de la symétrie.

Ces résultats suggèrent l'intérêt de modèles robustes aux distributions non gaussiennes et éventuellement d'une transformation logarithmique de la variable cible pour stabiliser la variance et améliorer certains modèles comme SVR et MLP.

3.3.3 Analyse des Relations Variables–Résistance (Scatter plots)

Les scatter plots feature–cible, avec une droite de régression linéaire, confirment globalement les corrélations observées. La relation armature–résistance montre une tendance croissante claire. La largeur de dalle présente aussi une évolution positive.

En revanche, la hauteur du connecteur et l'épaisseur de semelle présentent une forte dispersion, indiquant des effets non linéaires et des interactions entre variables.

On observe également une hétéroscédasticité : la dispersion de la résistance augmente avec certaines variables, ce qui suggère que des modèles non linéaires (XGBoost, RF) sont plus adaptés que des modèles linéaires.

3.3.4 Détection des Valeurs Aberrantes (Boxplots)

La méthode IQR sera utilisée pour détecter les valeurs aberrantes." Les résultats sont présentés au chapitre 4.

3.3.5 Apport de l'EDA dans la Compréhension Physique du Matériau

L'EDA montre que la résistance au cisaillement d'un connecteur en U dépend de plusieurs variables avec des relations non linéaires. L'armature est la variable la plus influente, suivie de la largeur de dalle et de l'espacement.

Les paramètres du connecteur (hauteur, épaisseur d'âme, longueur) ont un effet plus faible individuellement, mais agissent en interaction avec la géométrie de la dalle.

Ces dépendances croisées justifient l'utilisation de modèles non linéaires comme XGBoost et Random Forest, capables de capturer ces interactions, et valide la stratégie de stacking adoptée dans le pipeline.

3.4 Augmentation des Données et Séparation Train/Test

3.4.1 Justification Scientifique de l'Augmentation des Données

Le jeu de données initial contient 618 observations, ce qui reste limité au regard de la complexité du phénomène étudié et du nombre de paramètres des modèles d'ensemble. Dans ce contexte expérimental coûteux en ingénierie des structures, l'augmentation des données par ajout de bruit gaussien constitue une solution scientifiquement justifiée. Elle permet de reproduire la variabilité naturelle des matériaux ainsi que les incertitudes de mesure, sans générer de don-

nées irréalistes. Cette approche contribue également à améliorer la robustesse des modèles, en réduisant le surapprentissage et en favorisant une meilleure généralisation.

3.4.2 Méthodes Utilisées pour Enrichir le Dataset

Afin d'améliorer la capacité de généralisation des modèles d'apprentissage automatique et de limiter les effets liés à la taille réduite du dataset, une augmentation des données par bruit gaussien a été appliquée. Cette méthode consiste à générer des copies synthétiques des données originales en ajoutant un bruit aléatoire de faible amplitude, proportionnel à l'écart-type des variables :

$$X_{noise} = X + \mathcal{N}(0, 0.02 \cdot \sigma_X), \quad y_{noise} = y + \mathcal{N}(0, 0.02 \cdot \sigma_y)$$

où σ_X et σ_y représentent respectivement l'écart-type des variables d'entrée et de sortie. Le niveau de bruit choisi (2%) reste cohérent avec les incertitudes expérimentales généralement observées dans les essais mécaniques. Cette approche est couramment utilisée pour améliorer la robustesse des modèles prédictifs lorsque les données expérimentales sont limitées. Des travaux récents ont montré que l'augmentation par bruit gaussien améliore les performances et la capacité de généralisation des modèles d'apprentissage automatique [50]. Grâce à cette méthode, le dataset final est passé de 618 à 2472 observations.

3.4.3 Prétraitement et Normalisation

Après augmentation, une standardisation Z-score est appliquée via **StandardScaler**, qui centre et réduit chaque variable :

$$X_{scaled} = \frac{X - \mu}{\sigma}$$

Le scaler est ajusté uniquement sur le jeu d'entraînement, puis appliqué au jeu de test afin d'éviter toute fuite d'information.

Cette normalisation est essentielle pour SVR et MLP, et utile pour les méthodes basées sur les distances (k-NN). Les modèles d'arbres (XGBoost, RF) ne sont pas sensibles à cette transformation.

3.4.4 Définition des Variables d'Entrée (X) et de Sortie (y)

Les données sont séparées en variables explicatives et variable cible.

X — 12 variables d'entrée : largeur de la dalle, épaisseur de la dalle, armature, longueur de la dalle, résistance à la compression du béton (f_c), hauteur du connecteur, largeur du connecteur, épaisseur de l'âme, épaisseur des semelles, longueur du connecteur, nuance d'acier et espacement des connecteurs.

y — Variable cible : résistance au cisaillement par connecteur (kN), représentant la résistance au cisaillement mesurée expérimentalement.

Toutes les variables sont numériques, donc aucun encodage n'est nécessaire.

3.4.5 Division Train/Test

Le dataset augmenté (2 472 observations) est divisé selon un ratio 80/20 avec **random_state = 42** pour assurer la reproductibilité.

Ensemble	Observations	Proportion
Entraînement	1 977	80%
Test	495	20%

Aucune stratification n'est appliquée (problème de régression). Le mélange aléatoire permet de conserver une bonne représentation des différentes plages de résistance dans les deux ensembles.

3.4.6 Importance du Split pour la Généralisation

La séparation train/test est essentielle pour évaluer correctement la capacité de généralisation d'un modèle. L'évaluation sur les mêmes données d'entraînement conduirait à des résultats biaisés et trop optimistes.

Le jeu de test, jamais utilisé durant l'entraînement, permet de simuler de nouvelles configurations expérimentales et d'obtenir une estimation réaliste des performances.

La validation croisée à 5 folds (KFold, cv = 5) est appliquée uniquement sur le jeu d'entraînement pour l'optimisation des hyperparamètres. Le jeu de test reste totalement isolé jusqu'à l'évaluation finale, garantissant des métriques fiables (MAE, RMSE, R^2).

3.4.7 Métriques d'évaluation des modèles (R^2, MAE, RMSE, MSE)

Quatre métriques de performance complémentaires ont été retenues pour évaluer et comparer les modèles :

- **R^2(coefficient de détermination)** : mesure la proportion de variance expliquée par le modèle. Une valeur proche de 1 indique une très bonne capacité prédictive.

  Formule :

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

- **MAE (Mean Absolute Error)** : représente l'erreur moyenne absolue en kN. Cette métrique est robuste aux valeurs extrêmes et facile à interpréter physiquement.

  Formule :

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

- **RMSE (Root Mean Squared Error)** : pénalise davantage les grandes erreurs de prédiction, ce qui la rend pertinente pour les applications de sécurité structurelle.

  Formule :

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

- **MSE (Mean Squared Error)** : correspond au carré du RMSE. Elle est sensible aux erreurs importantes et permet de détecter les prédictions fortement éloignées des valeurs réelles.

  Formule :

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

3.5 Entraînement des modèles et optimisation des hyperparamètres (GridSearchCV)

3.5.1 Présentation des Modèles Sélectionnés

Cinq algorithmes de régression ont été retenus pour ce pipeline, couvrant des familles d'apprentissage distinctes : les méthodes d'ensemble par bagging (RF) et boosting (XGBoost), les machines à noyau (SVR), les réseaux de neurones artificiels (MLP) et les méthodes à base de similarité (k-NN).

Cette diversité est délibérée : chaque famille capture différemment les non-linéarités et les interactions entre variables, ce qui constitue une condition favorable à la construction d'un méta-modèle de stacking performant.

Tous les modèles sont encapsulés dans des (Pipeline) de (scikit-learn), afin de garantir l'étanchéité du prétraitement lors de la validation croisée.

3.5.2 Régression par Forêt Aléatoire

Forêt Aléatoire est une méthode d'ensemble par bagging qui construit un grand nombre d'arbres de décision indépendants sur des sous-échantillons bootstrap du jeu d'entraînement, puis agrège leurs prédictions par moyenne.

Cette approche permet de réduire la variance sans augmenter le biais, ce qui la rend naturellement robuste au surapprentissage.

Les hyperparamètres explorés sont :

- le nombre d'arbres (`n_estimators` : 100, 200, 300),
- la profondeur maximale (`max_depth` : None, 10, 20),
  - le nombre minimal d'échantillons pour diviser un nœud (`min_samples_split` : 2, 5),
  - le nombre minimal d'échantillons dans une feuille (`min_samples_leaf` : 1, 2).

Sa robustesse aux valeurs aberrantes (outliers) et sa capacité à estimer l'importance des variables explicatives en font un modèle de référence solide pour

ce type de données structurelles.

3.5.3 Régresseur XGBoost

XGBoost est une méthode d'ensemble par boosting par gradient qui construit les arbres de manière séquentielle, chaque nouvel arbre venant corriger les résidus du précédent.

Elle intègre une régularisation L1/L2 native, ce qui la distingue du gradient boosting classique et lui confère une meilleure résistance au surapprentissage sur des jeux de données de taille modérée.

La grille de recherche des hyperparamètres comprend :

- le nombre d'estimateurs (`n_estimators` : 100, 200, 300),
- la profondeur des arbres (`max_depth` : 3, 4, 5),
- le taux d'apprentissage (`learning_rate` : 0.05, 0.1, 0.2),
- le sous-échantillonnage des observations (`subsample` : 0.8, 1.0),
- le sous-échantillonnage des variables (`colsample_bytree` : 0.8, 1.0).

XGBoost est généralement le modèle de base le plus performant dans ce pipeline, en raison de sa capacité à modéliser des interactions d'ordre élevé entre variables géométriques.

3.5.4 Régression par vecteurs de support (SVR)

SVR applique le principe des machines à vecteurs de support à la régression en cherchant une fonction qui approxime la cible avec une erreur inférieure à un seuil ε, tout en maximisant la marge.

Le noyau **RBF** (Radial Basis Function) transforme implicitement les données dans un espace de grande dimension, où des relations non linéaires deviennent plus facilement séparables.

La grille de recherche des hyperparamètres comprend :

- le paramètre de régularisation (`C` : 0.1, 1, 10, 100),

- la largeur du tube d'insensibilité (`epsilon` : 0.01, 0.1, 0.5),
- le type de noyau (`kernel` : rbf, poly),
- le coefficient du noyau (`gamma` : scale, auto).

SVR est particulièrement sensible à l'échelle des données, ce qui rend indispensable la normalisation appliquée en amont dans le pipeline.

3.5.5 Perceptron Multi-Couches (MLP)

Le **MLP** est un réseau de neurones entièrement connecté (feedforward) capable d'approximer toute fonction continue par composition de transformations non linéaires.

Il est particulièrement adapté à la capture de relations complexes et d'interactions croisées entre les variables explicatives.

Les hyperparamètres optimisés portent sur :

- l'architecture du réseau (`hidden_layer_sizes` : (64,32), (128,64), (64,64,32)),
- la fonction d'activation (`activation` : relu, tanh),
- le coefficient de régularisation L2 (`alpha` : 0.0001, 0.001, 0.01),
- le taux d'apprentissage initial (`learning_rate_init` : 0.001, 0.01).

Le nombre maximal d'itérations est fixé à 500 afin de garantir la convergence, et **random_state=42** assure la reproductibilité de l'initialisation des poids.

3.5.6 Régression par K-Plus Proches Voisins (k-NN)

k-NN est un algorithme non paramétrique basé sur la similarité, qui prédit la résistance d'une nouvelle configuration en calculant la moyenne (éventuellement pondérée) des résistances des k observations d'entraînement les plus proches dans l'espace des variables explicatives, selon une distance donnée.

Sa logique est particulièrement adaptée à ce problème : deux connecteurs de géométries similaires, intégrés dans des dalles de propriétés proches, sont physiquement susceptibles de présenter des résistances voisines.

Les hyperparamètres optimisés sont :

- le nombre de voisins (`n_neighbors` : 3, 5, 7, 9, 11),
- la fonction de pondération (`weights` : uniform, distance),
- la métrique de distance (`metric` : euclidean, manhattan).

Étant donné que k-NN est fortement sensible à l'échelle des variables, la normalisation via **StandardScaler** appliquée en amont dans le pipeline constitue une condition indispensable à ses performances.

3.5.7 Principe de GridSearchCV

GridSearchCV est une procédure d'optimisation exhaustive des hyperparamètres de **scikit-learn**. Elle explore toutes les combinaisons possibles de la grille de paramètres définie, évalue chacune par validation croisée sur le jeu d'entraînement, puis retourne la combinaison qui maximise la métrique cible — ici le coefficient de détermination R^2.

Son fonctionnement peut être résumé comme suit :

- Pour chaque combinaison $(p_1, p_2, \ldots, p_n)$ dans la grille :
  - effectuer une validation croisée (*k-fold CV*) sur X_{train},
  - calculer le score moyen R^2 sur les k folds.
- retourner la combinaison avec le meilleur score moyen.

La métrique R^2 a été choisie comme critère d'optimisation car elle mesure la proportion de variance expliquée par le modèle, ce qui est plus informatif que le MAE ou le RMSE pour comparer des configurations de modèles hétérogènes en termes d'échelle de prédiction.

3.5.8 Recherche des Hyperparamètres Optimaux

Chaque modèle dispose de sa propre grille d'hyperparamètres, dont la taille totale varie de quelques dizaines à plusieurs centaines de combinaisons. Pour XGBoost, par exemple, la grille couvre :

$$3 \times 3 \times 3 \times 2 \times 2 = 108$$

Chacune de ces configurations est évaluée par validation croisée à 5 folds, soit 540 entraînements indépendants pour ce seul modèle.

L'argument **n_jobs=-1** permet de paralléliser l'exécution sur tous les cœurs disponibles, rendant cette exploration calculable dans un temps raisonnable.

Les meilleurs hyperparamètres sont automatiquement sélectionnés via **best_estimator_**, puis réinjectés dans le modèle final, qui est ensuite utilisé comme apprenant de base dans l'étape de stacking.

3.5.9 Validation Croisée Intégrée

La validation croisée à 5 **folds** (cv=5) est intégrée directement dans `GridSearchCV`, ce qui signifie qu'elle s'effectue entièrement à l'intérieur du jeu d'entraînement, sans jamais toucher au jeu de test.

À chaque **fold**, 80% des données d'entraînement servent à ajuster le modèle, tandis que 20% servent à l'évaluer. Ce processus est répété en permutant les rôles jusqu'à ce que chaque observation ait été utilisée une fois en validation.

Le score final retenu correspond à la moyenne des 5 scores R^2, ce qui fournit une estimation stable et peu sensible au découpage particulier des données.

Cette approche limite le risque de surapprentissage aux hyperparamètres et garantit que la sélection du meilleur modèle repose sur une performance réellement généralisable, et non sur un découpage favorable des données.

3.6 Construction du Modèle hybride

3.6.1 Concept d'Apprentissage en Ensemble

L'apprentissage en ensemble repose sur le principe fondamental selon lequel la combinaison de plusieurs modèles permet d'obtenir une prédiction plus robuste et plus précise qu'un modèle individuel pris isolément.

Cette supériorité s'explique par la diversification des erreurs : des modèles issus de familles algorithmiques différentes commettent des erreurs sur des régions distinctes de l'espace des variables explicatives, et leur agrégation tend à réduire ces erreurs locales.

Il existe trois grandes stratégies d'ensemble :

- le *bagging*,
- le *boosting*,
- le *stacking*.

Ces approches se distinguent par la manière dont les modèles sont construits et combinés. Le stacking, retenu dans ce travail, est la stratégie la plus flexible, car il apprend de manière optimale comment combiner les prédictions des modèles de base, plutôt que de les agréger par une moyenne simple.

3.6.2 Sélection des Modèles de Base

Les cinq modèles améliorés à l'étape précédente constituent les modèles de base (**base learners**) pour l'empilement (**stacking**) :Forêt Aléatoire , XGBoost, SVR, MLP et K-NN.

Ce choix est motivé par leur diversité algorithmique. Forêt Aléatoire et XGBoost sont des méthodes basées sur des arbres capables de capturer des interactions complexes entre les variables. Le SVR modélise des relations non linéaires dans un espace à noyau, tandis que le MLP apprend des représentations hiérarchiques à travers des transformations successives. Enfin, le modèle K-NN repose sur la similarité locale entre observations pour estimer la résistance au cisaillement sans hypothèse explicite sur la forme de la relation étudiée.

Cette diversité est une condition essentielle pour que le stacking fonctionne bien : des modèles très similaires produiront des prédictions fortement corrélées, ce qui limite l'avantage potentiel de leur combinaison.

3.6.3 Principe du Stacking Regressor

Le stacking fonctionne selon une architecture à deux niveaux (two-level learning).

Au niveau 0, chaque modèle de base est entraîné sur le jeu d'entraînement à l'aide d'une validation croisée à k folds (cv=5). Pour chaque fold, le modèle est ajusté sur $k-1$ folds et génère des prédictions hors-échantillon sur le fold restant.

Ce mécanisme, appelé **cross-val predict**, garantit que les prédictions utilisées pour entraîner le méta-apprenant n'ont jamais été vues par le modèle qui les produit, ce qui évite toute fuite d'information.

L'ensemble de ces prédictions forme une matrice de méta-features de dimension $(n_{train} \times 5)$, où chaque colonne correspond aux prédictions d'un modèle de base.

Au niveau 1, un méta-apprenant est entraîné sur cette matrice afin d'apprendre la combinaison optimale des prédictions issues des modèles de base.

Schéma du stacking :

$$
\begin{aligned}
X_{train} &\rightarrow RF \rightarrow \hat{y}_{RF} \\
&\rightarrow XGB \rightarrow \hat{y}_{XGB} \\
&\rightarrow SVR \rightarrow \hat{y}_{SVR} \Rightarrow [\hat{y}_{RF} \mid \hat{y}_{XGB} \mid \hat{y}_{SVR} \mid \hat{y}_{MLP} \mid \hat{y}_{kNN}] \\
&\qquad\qquad\qquad\qquad\qquad\qquad \rightarrow \text{Ridge} \rightarrow \hat{y}_{final} \\
&\rightarrow MLP \rightarrow \hat{y}_{MLP} \\
&\rightarrow k\text{-}NN \rightarrow \hat{y}_{kNN}
\end{aligned}
$$

3.6.4 Choix de Ridge comme Méta-Apprenant

La régression Ridge est une régression linéaire régularisée par une pénalité L2 appliquée aux coefficients :

$$\hat{\beta} = \arg\min_{\beta} \|y - X\beta\|^2 + \alpha\|\beta\|^2$$

Son choix comme méta-apprenant est justifié par plusieurs raisons.

Premièrement, les méta-features d'entrée, constituées des prédictions des modèles de base, sont déjà des estimations de la cible et présentent souvent une forte corrélation entre elles. La régression Ridge traite naturellement cette multicolinéarité, contrairement à une régression linéaire ordinaire (**OLSR**) potentiellement instable.

Deuxièmement, sa simplicité structurelle limite le risque de surapprentissage au niveau du méta-modèle, les modèles de base ayant déjà capturé les principales non-linéarités du problème.

Troisièmement, les coefficients appris demeurent interprétables : ils indiquent la pondération attribuée à chaque modèle de base, offrant ainsi une lecture directe de leur contribution relative à la prédiction finale.

3.6.5 Architecture Finale du Modèle Hybride

L'architecture complète du modèle hybride peut être représentée comme suit :

Tableau 3.5 : Architecture finale du modèle hybride par stacking

Niveau	Composant	Rôle
Prétraitement	StandardScaler	Normalisation des 12 variables
Niveau 0 – Base	RF	Bagging et robustesse
Niveau 0 – Base	XGBoost	Boosting et interactions non linéaires
Niveau 0 – Base	SVR	Modélisation non linéaire par noyau
Niveau 0 – Base	kNN	Approche locale par similarité
Niveau 0 – Base	MLP	Apprentissage profond non linéaire
Niveau 1 – Méta	Ridge ($\alpha = 1$)	Fusion des prédictions

Le paramètre **passthrough=False** est conservé à sa valeur par défaut, ce qui signifie que le méta-apprenant reçoit uniquement les prédictions des modèles de base, sans accès direct aux variables explicatives initiales. Ce choix permet de limiter la complexité du modèle final ainsi que le risque de surapprentissage.

L'ensemble du pipeline est encapsulé dans un objet **StackingRegressor** de **scikit-learn**, permettant l'entraînement complet du modèle au moyen d'une seule instruction **fit()**.

3.6.6 Avantages en Robustesse et Précision

Le modèle de stacking offre plusieurs avantages par rapport à un modèle unique.

En termes de précision, il combine les forces complémentaires des modèles de base. Par exemple, XGBoost peut être performant sur les configurations fréquentes, tandis que SVR et MLP peuvent mieux interpoler certaines zones moins représentées.

En termes de robustesse, la diversité des modèles réduit l'impact d'une prédiction aberrante produite par l'un d'eux, le méta-apprenant assurant une combinaison plus stable.

Enfin, le mécanisme de validation croisée intégré permet une meilleure généralisation, en corrigeant les biais spécifiques de chaque modèle de base.

3.7 Conclusion

Ce chapitre a permis de construire, étape par étape, un pipeline complet de Machine Learning pour la prédiction de la résistance au cisaillement des connecteurs en U.

L'analyse exploratoire a mis en évidence la nature non linéaire du phénomène, justifiant le recours à cinq algorithmes complémentaires — RF, XGBoost, SVR, MLP et k-NN — optimisés via GridSearchCV et combinés au sein d'un méta-modèle de stacking basé sur la régression Ridge.

Chapitre 04

Chapitre 4

Résultats et discussion

4.1 Introduction

L'analyse exploratoire des données, dont la méthodologie est décrite au chapitre 3 , a permis d'acquérir une compréhension approfondie de la structure, de la distribution et des relations intrinsèques du jeu de données expérimental (618 observations, 12 variables d'entrée).

4.1.1 Interprétation de la matrice de corrélation

La matrice de corrélation de Pearson (**figure 4.1**) fournit une évaluation quantitative des relations linéaires entre l'ensemble des variables du jeu de données. L'analyse met en évidence une hiérarchie claire des paramètres influençant la résistance au cisaillement cible, permettant d'identifier les variables ayant l'impact le plus significatif sur le comportement mécanique des connecteurs.

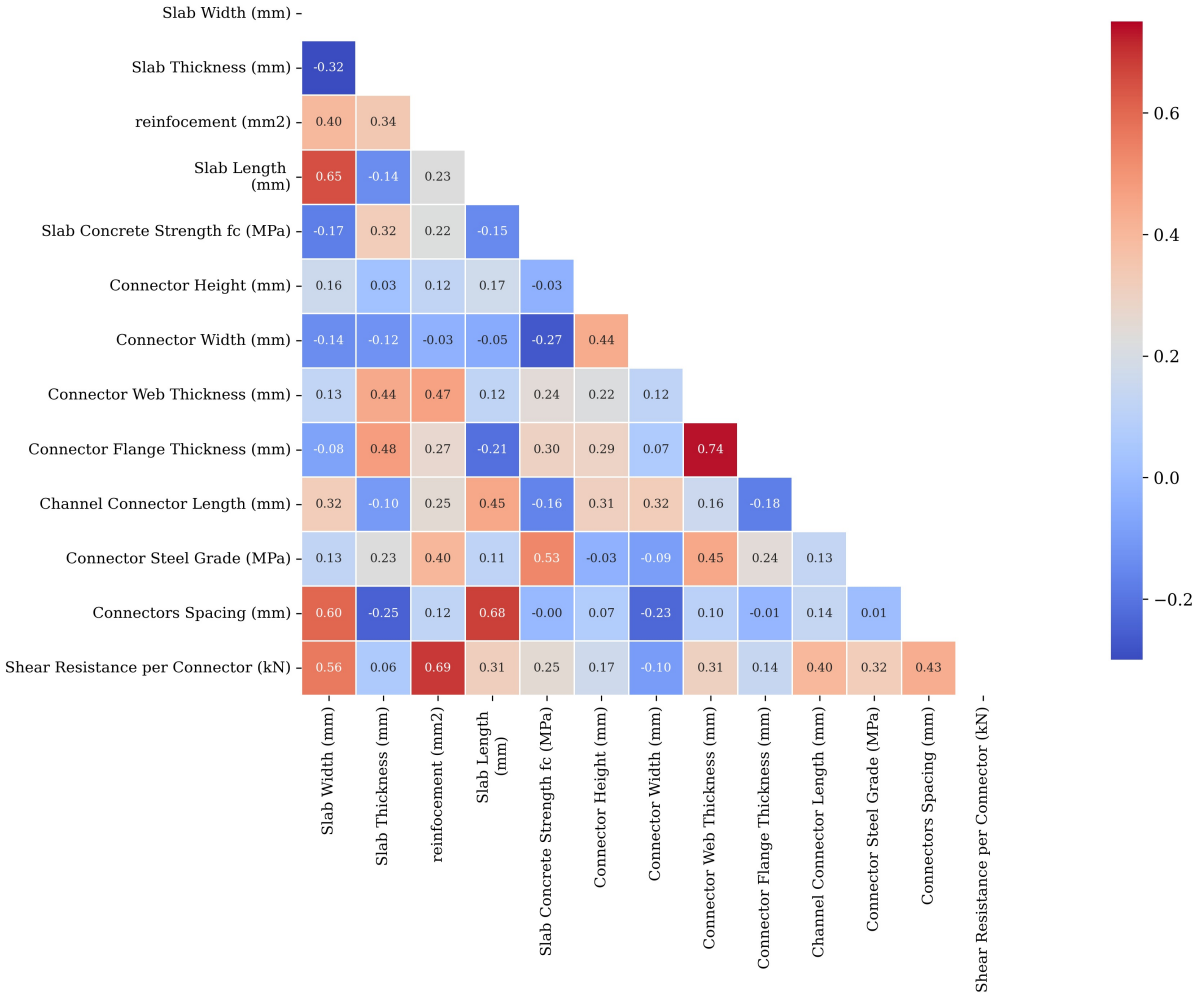


Figure 4.1 : Heatmap corrélation Pearson

Tableau 4.1 : Corrélation de Pearson entre les paramètres d'entrée et la résistance

Paramètre	**r**	**Interprétation**
Armature (mm^2)	0.689	Forte corrélation positive — une augmentation de l'armature entraîne une augmentation de la résistance
Largeur de dalle (mm)	0.564	Forte corrélation positive — meilleur engagement du béton
Espacement des connecteurs (mm)	0.430	Corrélation positive modérée — influence liée à la configuration du jeu de données
Longueur du connecteur (mm)	0.400	Corrélation positive modérée — surface de contact accrue
Nuance d'acier du connecteur (MPa)	0.323	Corrélation positive modérée — acier plus résistant
Longueur de dalle (mm)	0.315	Corrélation positive modérée
Épaisseur de l'âme du connecteur (mm)	0.307	Corrélation positive modérée — rigidité accrue de la section
Résistance du béton f_c (MPa)	0.253	Corrélation modérée — influence du béton
Hauteur du connecteur (mm)	0.171	Faible corrélation positive
Épaisseur de la semelle du connecteur (mm)	0.139	Faible corrélation positive
Épaisseur de dalle (mm)	0.061	Très faible corrélation — impact quasi nul
Largeur du connecteur (mm)	-0.099	Légère corrélation négative

Le renforcement de la dalle présente la corrélation la plus forte avec la résistance ($r = 0{,}689$), confirmant le rôle déterminant de l'armature passive dans l'amélioration de la rigidité et de la capacité portante de l'assemblage acier–béton. La largeur de la dalle affiche également une corrélation positive significative ($r = 0{,}564$), liée à l'augmentation de la zone d'interaction entre le connecteur et le béton.

L'espacement des connecteurs montre une corrélation positive apparente ($r = 0{,}430$), mais celle-ci est fortement biaisée par la structure du dataset : 437 observations sur 618 (70,7 %) présentent un espacement nul ($s = 0$), correspondant à des configurations à connecteur unique ou à des cas où l'espacement n'est pas applicable. Ces cas sont associés à une résistance moyenne de 122,6 kN, contre 261,1 kN pour les configurations avec espacement non nul ($n = 181$), ce qui conduit à une inflation artificielle de la corrélation. Lorsque l'analyse est restreinte aux valeurs positives, la corrélation diminue à $r = 0{,}194$, indiquant que la relation globale est principalement induite par la proportion élevée de zéros et ne traduit pas un effet causal direct.

Enfin, la longueur du connecteur en canal présente une corrélation positive modérée ($r = 0{,}400$), suggérant une amélioration du transfert des efforts de cisaillement via les mécanismes combinés de frottement et d'emboîtement mécanique. Une observation particulière concerne la largeur du connecteur, qui présente une faible corrélation négative ($r = -0.099$). attribuée aux interactions géométriques complexes entre les dimensions du connecteur et certains paramètres constructifs.

4.1.2 Analyse des distributions statistiques et courbes KDE (Kernel Density Estimation)

La variable cible (**la résistance au cisaillement par connecteur**) montre une distribution fortement décalée vers la droite (skewness positif), avec un point le plus bas de 36.0 kN et un point le plus haut de 1700.0 kN, une moyenne de 163.2 kN et une déviation standard de 156.7 kN. Cette forte variabilité ($CV \approx 96\%$) justifie l'utilisation d'algorithmes d'apprentissage automatique capables

de saisir des relations complexes non linéaires.

La plupart des paramètres géométriques montrent des distributions multimodales typiques de jeux de données provenant de compilations expérimentales multi-sources, chaque mode correspondant à une série d'essais spécifique. La résistance du béton f_c varie de 19.44 MPa et 82.0 MPa , couvrant l'ensemble des classes de béton de structure usuelles (C20 à C80).

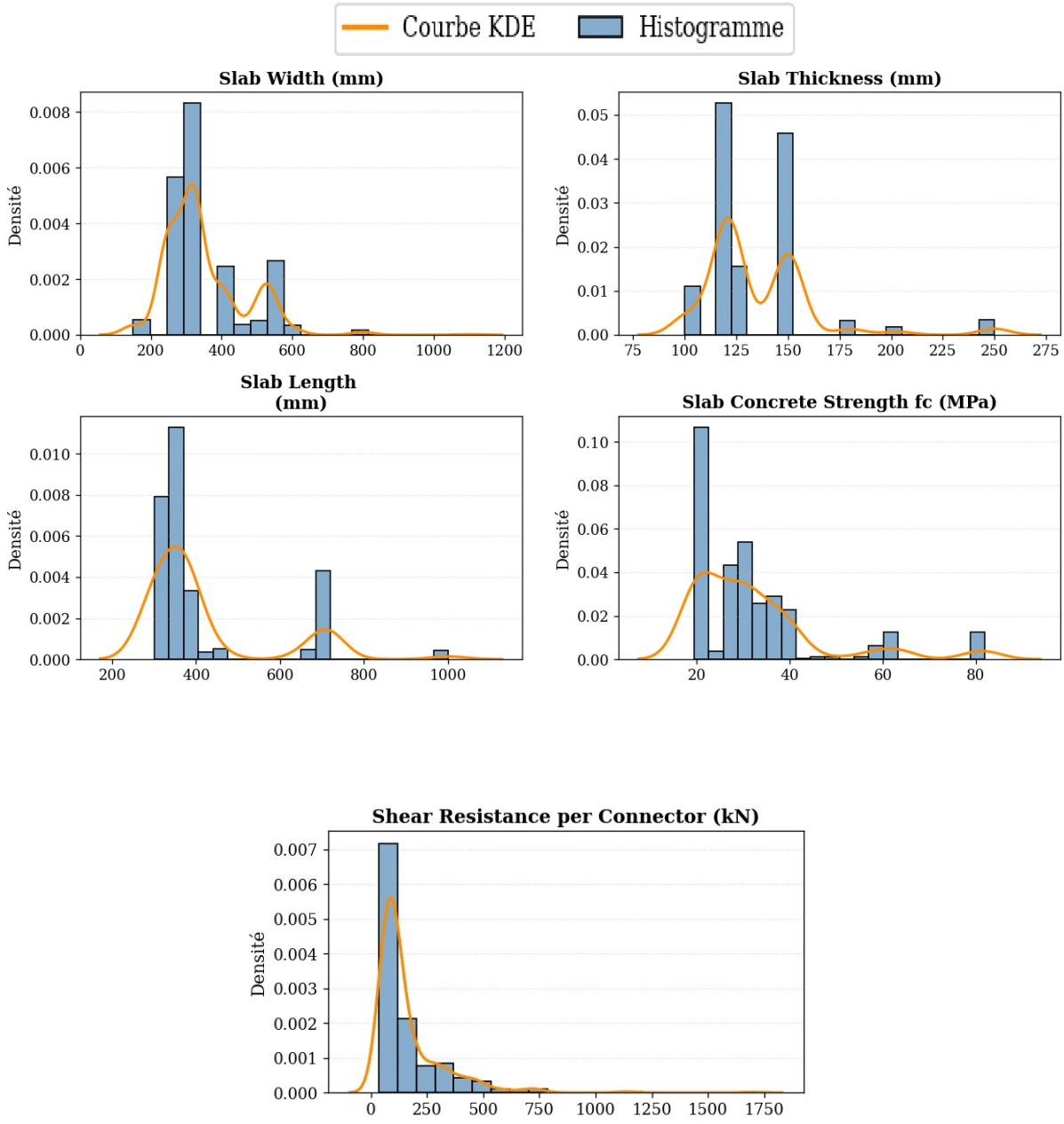


Figure 4.2 : Distributions des variables(a)

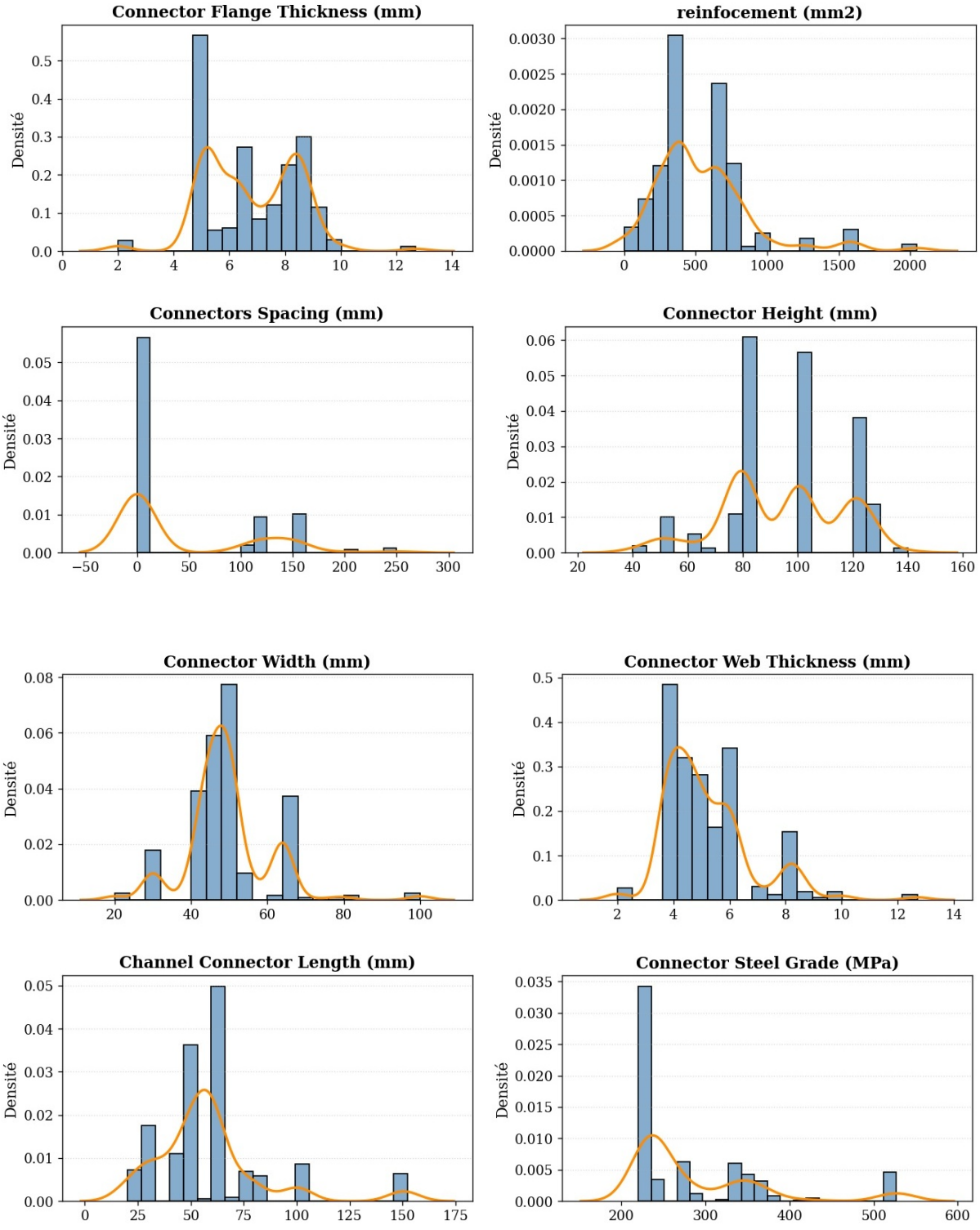


Figure 4.3 : Distributions des variables(b)

4.1.3 Relations entre paramètres d'entrée et résistance

Les diagrammes de dispersion (**scatter plots**) entre chaque paramètre d'entrée et la résistance cible (**Figure 4.4 et 4.5**) révèlent la nature fondamentalement

non-linéaire des relations physiques à l'œuvre.

L'armature de dalle, la longueur du connecteur et la résistance du béton présentent une influence positive sur la résistance au cisaillement, avec des relations globalement cohérentes avec le comportement mécanique attendu et les formulations analytiques classiques.

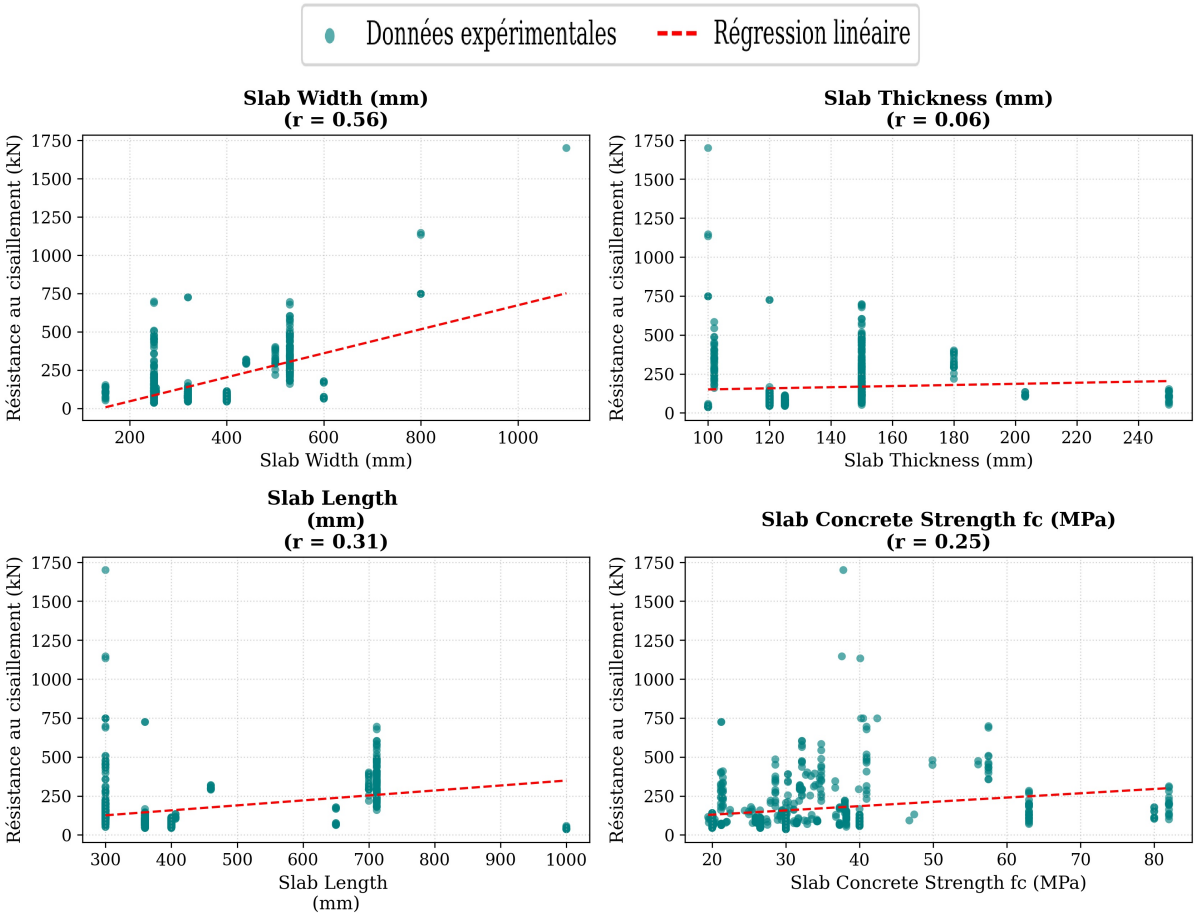


Figure 4.4 : Scatter plots paramètres vs résistance(a)

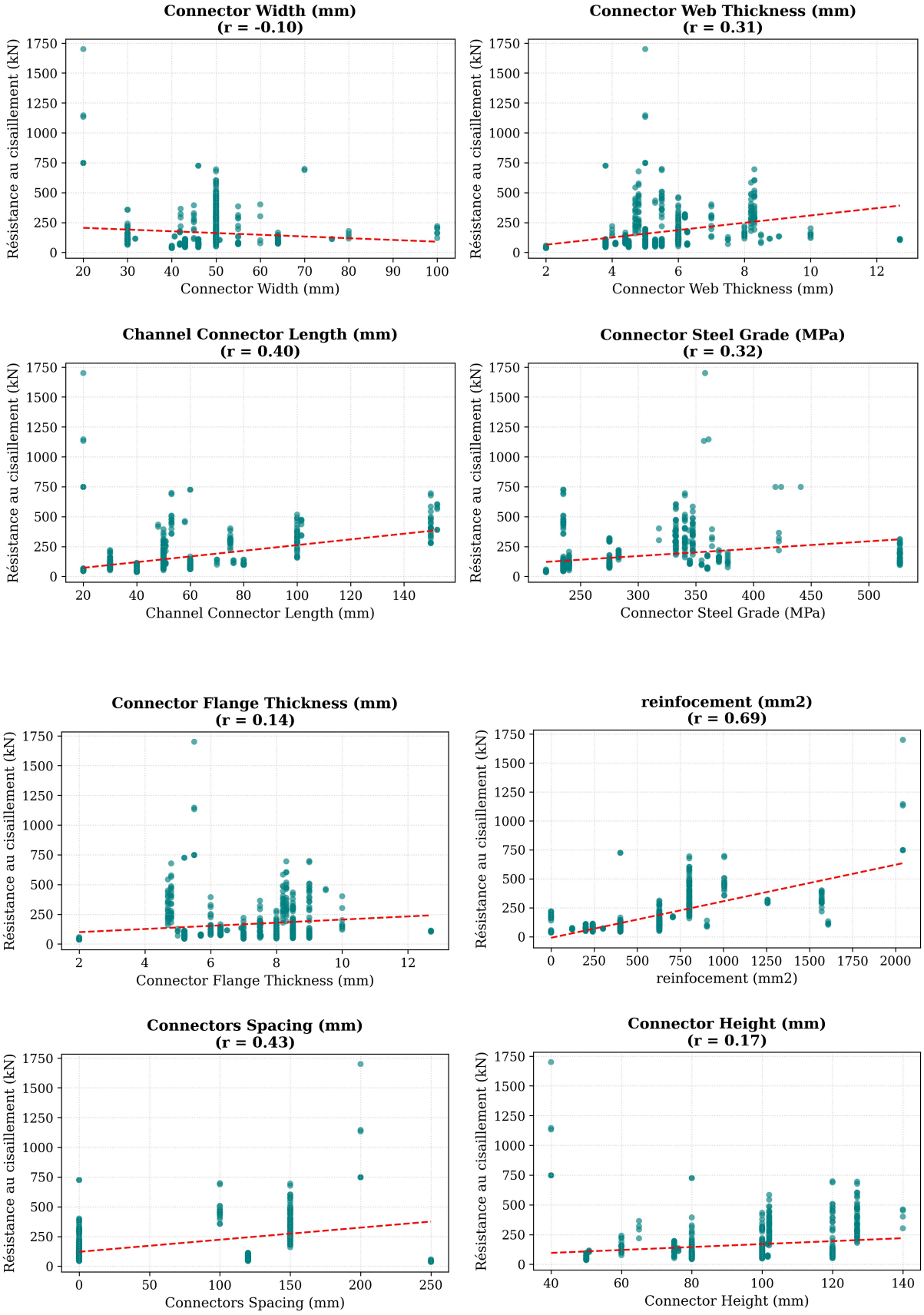


Figure 4.5 : Scatter plots paramètres vs résistance(b)

4.1.4 Détection et impact des valeurs aberrantes

L'analyse des box plots standardisés (Figure 4.6 et 4.7) met en évidence les observations situées au-delà de $\pm 1.5\, IQR$. La résistance cible atteint $1700\ kN$, soit environ neuf fois la médiane ($103.5\ kN$). Les valeurs aberrantes n'ont pas été supprimées car elles correspondent à des essais expérimentaux réels.L'augmen-tation par bruit gaussien (voir au chapitre 3 (section 3.4.2)) a permis de limiter leur influence.

L'analyse exploratoire a permis de dégager plusieurs points importants qui orientent le choix des modèles :

- La résistance au cisaillement dépend simultanément de plusieurs facteurs : armature, géométrie du connecteur et propriétés des matériaux.
- Les relations observées sont fondamentalement non linéaires, justifiant le recours à des modèles d'apprentissage avancés.
- La forte variabilité de la variable cible ($CV \approx 96\%$) confirme l'intérêt des modèles RF et XGBoost.
- Le jeu de données ne contient aucune valeur manquante (0 sur $618 \times 13 = 8034$ cellules).

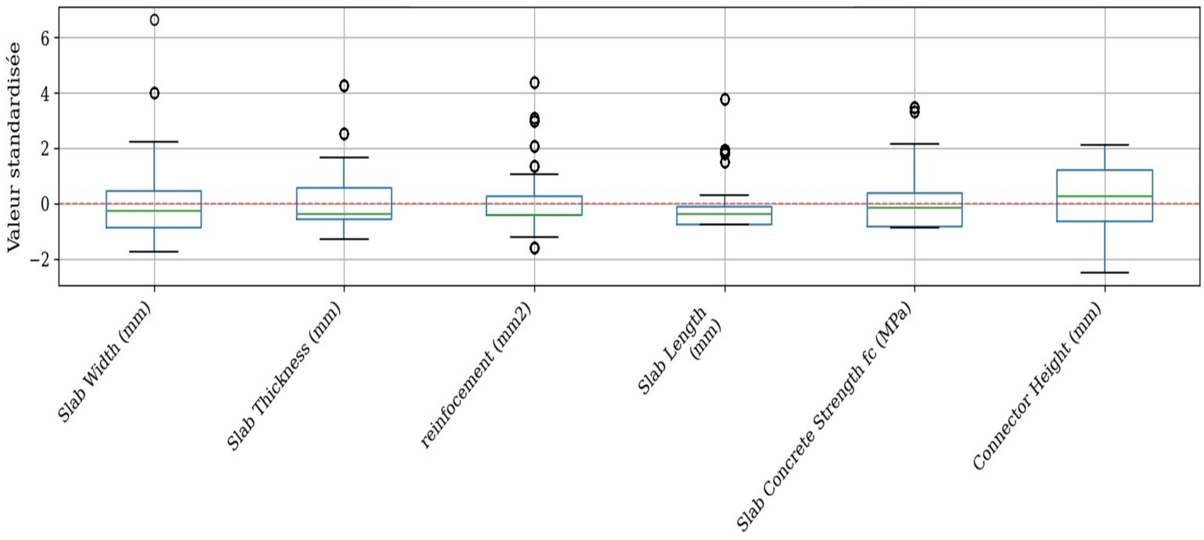


Figure 4.6 : Box plots standardisés(a)

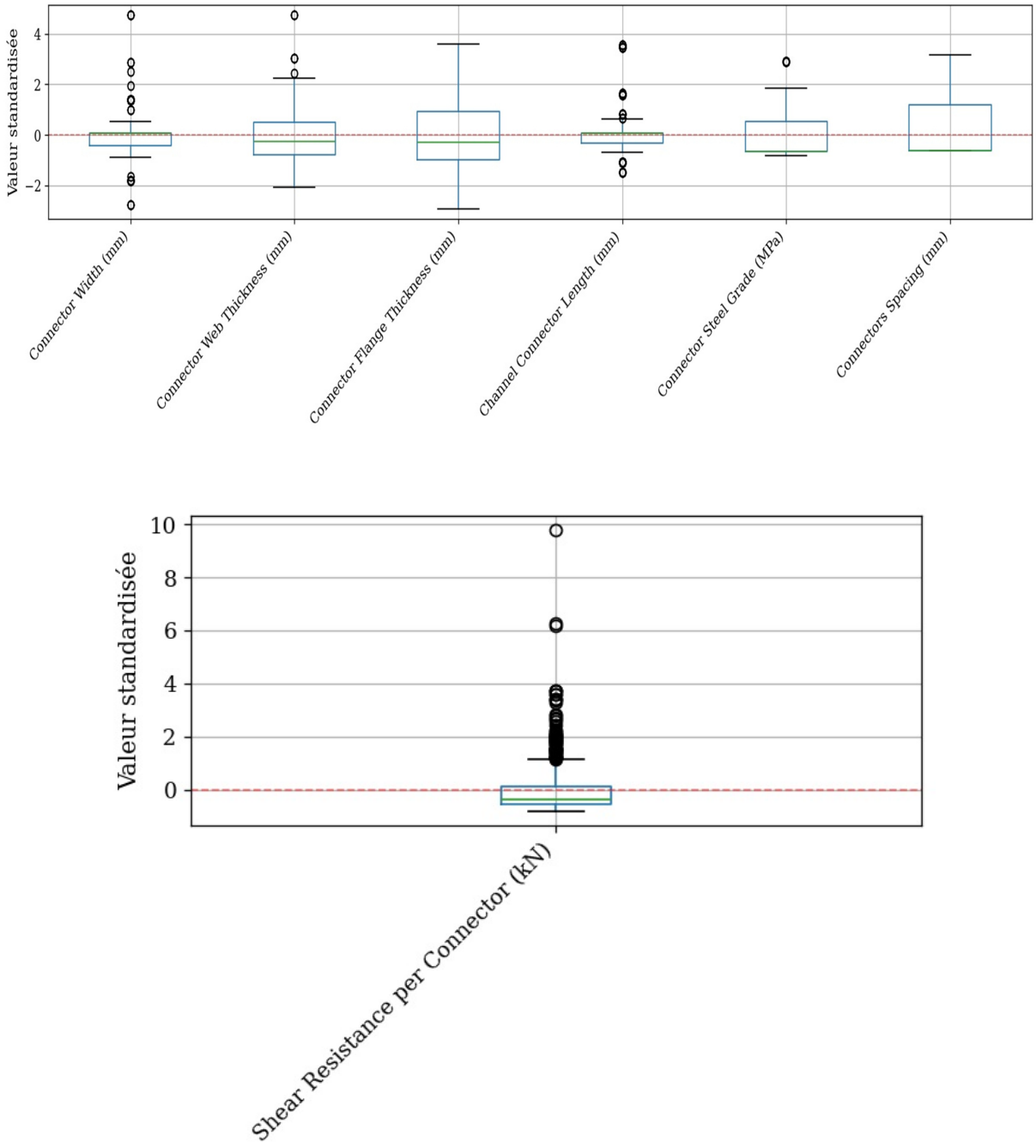


Figure 4.7 : Box plots standardisés(b)

4.2 Performances des modèles — métriques et comparaison

Cette section présente une évaluation quantitative rigoureuse des six modèles développés : XGBoost, RF, SVR , MLP , k-NN et le méta-modèle Stacking basé sur un régresseur Ridge.

L'évaluation est réalisée sur le jeu de test (20% du dataset augmenté, soit environ 494 observations après une augmentation par un facteur 4).

4.2.1 Critères d'évaluation (R², RMSE, MAE, MSE)

Les résultats de l'évaluation des six modèles sur l'ensemble de test sont résumés dans le Tableau 4.2, permettant une comparaison directe de leurs performances selon quatre mesures d'évaluation : R² , MAE, RMSE et MSE, présentées au Chapitre 3.

4.2.2 Résultats individuels des modèles de base

Chaque modèle a été optimisé via **GridSearchCV** (voir section 3.5.7). Les résultats obtenus sont présentés dans la figure 4.8.

4.2.3 Performance du modèle Stacking

Le modèle Stacking combine les prédictions des cinq modèles de base. La prédiction finale est obtenue via le méta-modèle Ridge ($\alpha = 1.0$), ce qui permet d'obtenir des résultats plus stables et plus précis que tout modèle individuel.

4.2.4 Comparaison quantitative

Tableau 4.2 : Performances comparatives de tous les modèles sur les jeux d'entraînement et de test

Modèle	R^2 **Test**	R^2 **Train**	**MSE Test**	**MAE Test**	**RMSE Test**	**Rang**
Stacking	0.8974	0.9784	2555.2860	20.4375	50.5498	Meilleur
k-NN	0.8862	0.9851	2833.9704	20.1814	53.2350	2ème
Random Forest	0.8832	0.9730	2910.8349	19.6984	53.9522	3ème
XGBoost	0.8750	0.9606	3113.3314	23.8620	55.7972	4ème
MLP	0.8712	0.9351	3209.8804	20.6176	56.6558	5ème
SVR	0.8477	0.9272	3792.9445	21.2133	61.5869	6ème

Analyse du Tableau

L'ensemble des modèles évalués affiche un coefficient de détermination R^2 supérieur à $0{,}84$ sur le jeu de test, ce qui témoigne d'une bonne capacité prédictive générale pour l'estimation de la résistance au cisaillement des connecteurs en acier. Cependant, des écarts non négligeables existent entre les modèles, tant sur le plan de la précision que sur celui de la généralisation.

Le modèle **Stacking** se distingue nettement avec un $R^2_{\text{test}} = 0{,}8974$, soit le plus élevé de tous les modèles. L'écart entre son $R^2_{\text{train}} = 0{,}9784$ et son R^2_{test} est de $0{,}081$, ce qui indique un niveau de surapprentissage maîtrisé. En comparaison, le modèle **k-NN** présente le R^2_{train} le plus élevé ($0{,}9851$) mais un écart train/test plus important ($\Delta = 0{,}099$), révélant une sensibilité accrue aux données d'entraînement et un risque de surapprentissage plus prononcé. Quant au modèle **SVR**, il enregistre les performances les plus faibles avec $R^2_{\text{test}} = 0{,}8477$, soit un écart de $0{,}050$ par rapport au Stacking — ce qui représente une perte relative de $\approx 5{,}5\%$ de pouvoir explicatif.

Analyse des erreurs absolues (MAE) et quadratiques (RMSE, MSE)

En ce qui concerne l'erreur absolue moyenne (MAE), c'est le modèle **Random Forest** qui se révèle le plus précis avec une valeur de MAE $= 19{,}70$ kN, suivi de près par **k-NN** ($20{,}18$ kN) et le modèle **Stacking** ($20{,}44$ kN). En revanche, **XGBoost** présente la MAE la plus défavorable ($23{,}86$ kN), ce qui peut s'expliquer par sa sensibilité aux valeurs aberrantes dans certaines régions de l'espace des variables. En effet, les arbres boostés optimisent généralement une fonction de perte quadratique, ce qui peut amplifier l'influence des erreurs ponctuelles importantes.

Concernant la racine de l'erreur quadratique moyenne (RMSE), le modèle **Stacking** conserve son avantage avec RMSE $= 50{,}55$ kN, soit un gain d'environ 11 kN par rapport au modèle **SVR** ($61{,}59$ kN). Cet écart, correspondant à une réduction d'environ $17{,}8\,\%$ du RMSE, confirme que le modèle Stacking gère mieux les cas extrêmes. Cette caractéristique est particulièrement importante dans un contexte d'ingénierie structurale, où les sous-estimations des résistances élevées peuvent entraîner des conséquences significatives sur la sécurité des ouvrages.

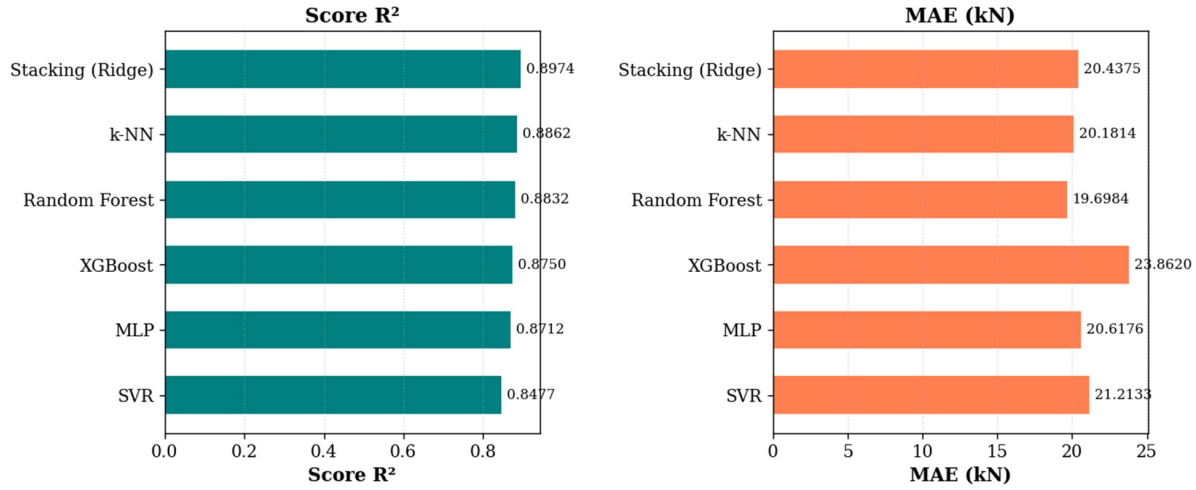


Figure 4.8 : Comparaison R², MAE en barres horizontales

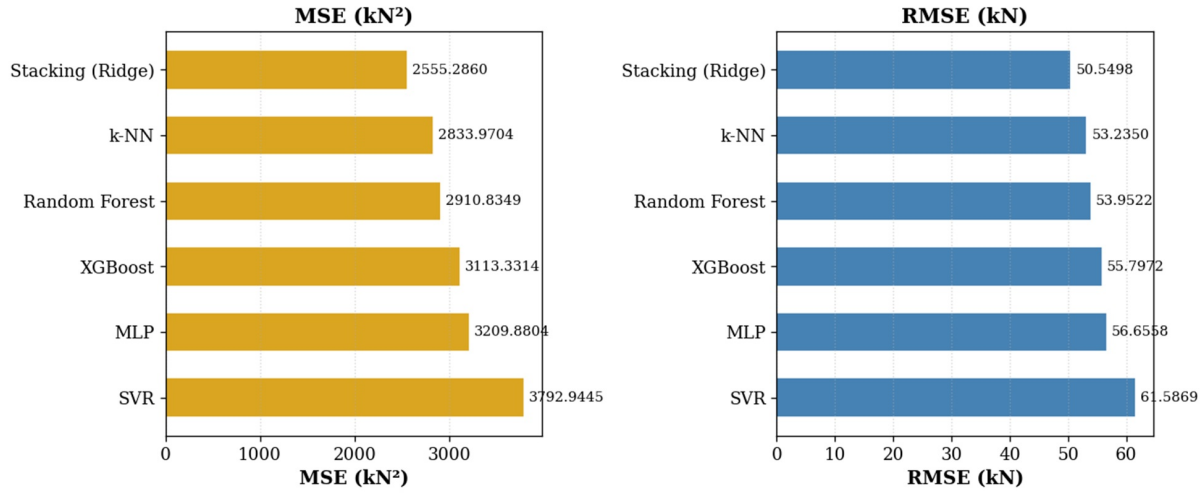


Figure 4.9 : Comparaison MSE,RMSE en barres horizontales

Les Figures 4.8 et 4.9 présentent une comparaison visuelle des quatre métriques de performance — R^2, MAE, MSE et RMSE — pour les six modèles évalués sur le jeu de test.

Score R^2 Le classement est dominé par le modèle Stacking (Ridge) avec $R^2 = 0{,}8974$, suivi de près par k-NN ($0{,}8862$) et Random Forest ($0{,}8832$). XGBoost ($0{,}8750$) et MLP ($0{,}8712$) occupent les positions intermédiaires, tandis que SVR affiche la performance la plus faible ($0{,}8477$). L'écart entre le meilleur et le moins bon modèle reste relativement modeste ($\approx 0{,}05$), ce qui témoigne de la qualité globale du pipeline d'apprentissage. Néanmoins, cet écart demeure significatif en termes d'application pratique, où chaque point de R^2 représente une meilleure capacité à expliquer la variance de la résistance au cisaillement.

MAE Contrairement au classement basé sur R^2, c'est Random Forest qui présente la MAE la plus faible ($19{,}70$ kN), devant k-NN ($20{,}18$ kN) et Stacking ($20{,}44$ kN). XGBoost enregistre la MAE la plus élevée ($23{,}86$ kN), ce qui indique que, malgré un R^2 satisfaisant, ses erreurs absolues individuelles sont plus importantes. Cette divergence entre R^2 et MAE pour XGBoost suggère que le modèle réalise des erreurs ponctuelles élevées sur certains échantillons, même si sa performance globale reste correcte.

MSE et RMSE Ces deux métriques, plus sensibles aux erreurs extrêmes, confirment la supériorité du modèle Stacking, avec MSE $= 2555{,}29$ kN^2 et RMSE $= 50{,}55$ kN, soit les valeurs les plus faibles parmi tous les modèles. Le modèle SVR présente les valeurs les plus élevées (MSE $= 3792{,}94$ kN^2 et RMSE $= 61{,}59$ kN), confirmant sa difficulté à gérer les configurations à résistance élevée. L'écart de RMSE entre Stacking et SVR (≈ 11 kN) est notable à l'échelle des applications de génie civil, où une telle différence peut influencer le dimensionnement des connecteurs.

En synthèse, aucun modèle individuel ne domine simultanément l'ensemble des métriques. Random Forest excelle en termes de MAE, tandis que Stacking se distingue par les meilleures performances en R^2, MSE et RMSE. Cette complémentarité entre modèles constitue précisément le principe du Stacking, qui combine les prédictions de plusieurs algorithmes afin de réduire à la fois les erreurs globales et les erreurs ponctuelles. Ces résultats justifient ainsi le choix du modèle Stacking comme modèle final de référence dans cette étude.

4.2.5 Classement des modèles

Le classement des modèles basé sur le coefficient de détermination R² sur le jeu de test établit la hiérarchie suivante : Stacking (0.8974) $>$ k-NN (0.8862) $>$ Random Forest (0.8832) $>$ XGBoost (0.8750) $>$ MLP (0.8712) $>$ SVR (0.8477).

Le modèle Stacking obtient les meilleures performances globales, suivi de près par k-NN et Random Forest, ce qui confirme l'efficacité des méthodes d'assemblage et des modèles non linéaires pour ce problème de régression.

4.2.6 Discussion scientifique des écarts de performance

Les méthodes d'ensemble donnent les meilleures performances, notamment RF ($R^2 = 0{,}8832$) et XGBoost ($R^2 = 0{,}8750$), grâce à leur capacité à capter les relations non linéaires et les interactions entre les 12 variables structurelles.

Le modèle MLP ($R^2 = 0{,}8712$) donne aussi de bons résultats mais reste sensible au choix des hyperparamètres. Le modèle k-NN ($R^2 = 0{,}8862$) présente également de bonnes performances, avec un léger risque de surapprentissage.

Le modèle SVR est le moins performant ($R^2 = 0{,}8477$), ce qui peut être expliqué par sa difficulté à représenter des relations complexes entre les variables et sa sensibilité aux paramètres du noyau.

Enfin, le stacking Ridge ($R^2 = 0{,}8974$) donne les meilleures performances globales, car il combine les prédictions des différents modèles de base et améliore la stabilité des résultats.

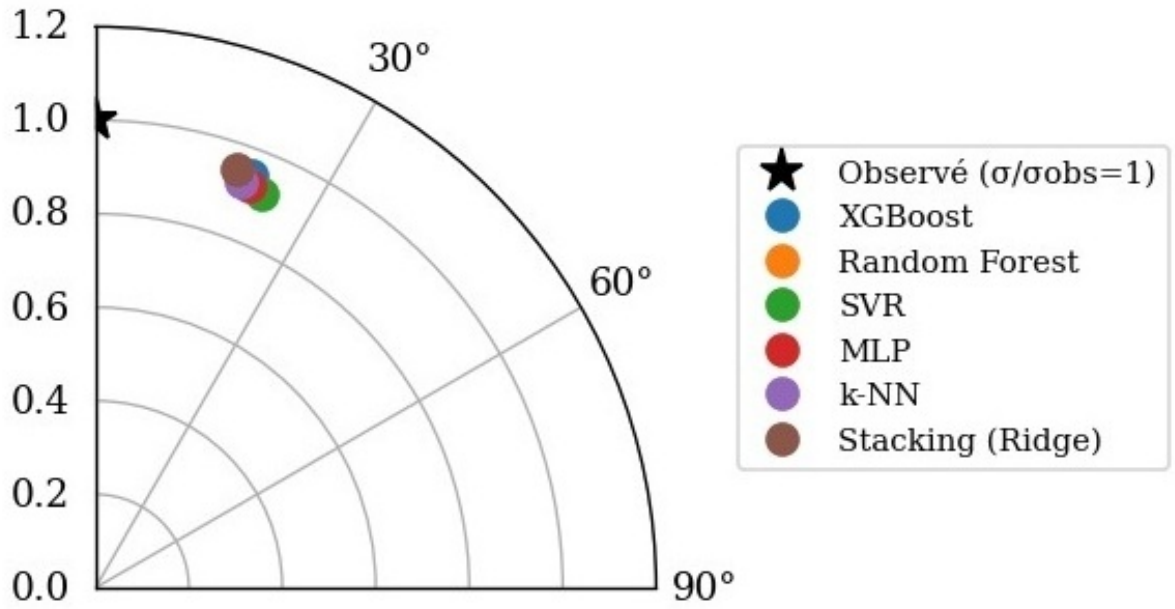


Figure 4.10 : Diagramme de Taylor

Le diagramme de Taylor indique que l'ensemble des modèles présentent une corrélation élevée ($r > 0{,}95$) avec les observations, reflétée par un angle polaire quasi nul. Néanmoins, le rapport $\sigma/\sigma_{obs} < 1$ pour tous les modèles, avec des valeurs approximatives comprises entre $0{,}85$ et $0{,}92$, révèle une sous-estimation systématique de la variance des prédictions, attribuable à la régression vers la

moyenne inhérente aux algorithmes entraînés par minimisation de l'erreur quadratique. Cette sous-dispersion, bien que n'affectant pas la tendance centrale, compromet la représentativité statistique des cas extrêmes et peut conduire à des estimations non conservatives de la résistance au cisaillement des connecteurs.

4.3 Analyse des résidus et erreurs

L'analyse des résidus complète l'évaluation des modèles en étudiant les erreurs de prévision et leur comportement. Un bon modèle doit produire des résidus distribués aléatoirement autour de zéro, sans tendance particulière ni structure visible.

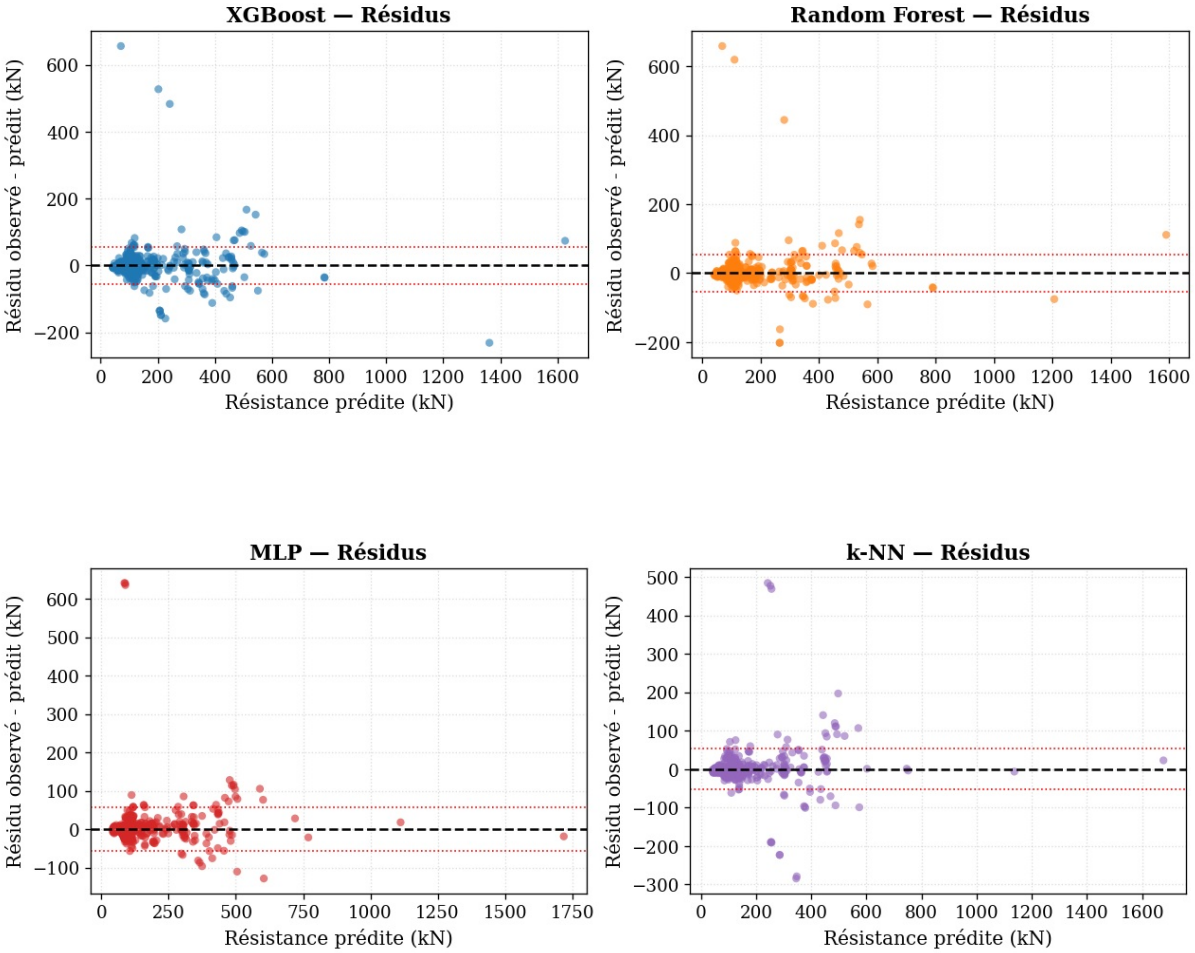


Figure 4.11 : Graphes de résidus(a)

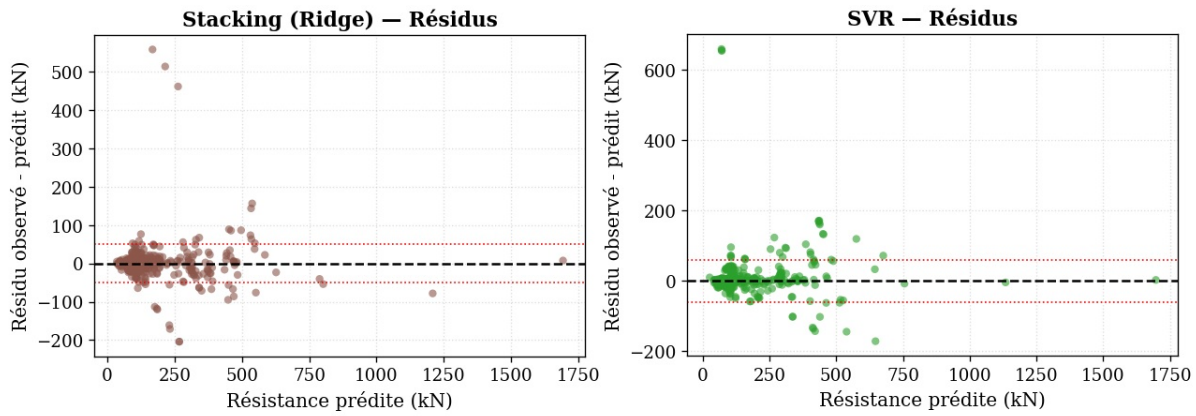


Figure 4.12 : Graphes de résidus(b)

L'examen des graphes de résidus (Figure 4.11,4.12) pour les six modèles révèle des comportements distincts. Pour XGBoost et Random Forest, les résidus sont globalement centrés autour de zéro sur la majeure partie de la plage de résistance prédite, avec une légère dispersion croissante au-delà de 600 kN, traduisant une sous-estimation ponctuelle pour les valeurs extrêmes.

Le modèle SVR présente une distribution plus resserrée dans la zone centrale (0–500 kN), mais affiche des résidus élevés et dispersés au-delà de 750 kN, ce qui révèle une capacité de généralisation limitée pour les configurations à haute résistance. Le MLP montre un comportement similaire dans la zone basse, avec quelques résidus extrêmes atteignant +600 kN, indiquant une instabilité pour certains échantillons.

Le modèle k-NN, quant à lui, présente une dispersion plus symétrique mais avec une amplitude de résidus plus large (de -300 à $+500$ kN), ce qui reflète la sensibilité de cette méthode aux valeurs atypiques dans l'espace des voisins.

Le modèle Stacking (Ridge) se distingue nettement : les résidus sont majoritairement contenus dans un intervalle resserré autour de zéro sur l'ensemble de la plage prédite, avec une répartition plus homogène et équilibrée entre résidus positifs et négatifs. Aucun biais systématique n'est observable, ce qui confirme l'absence de sur- ou sous-estimation structurelle. Cette stabilité est la conséquence directe de la combinaison des prédictions des modèles de base, qui compense les erreurs individuelles et améliore la régularité globale du modèle.

En synthèse, l'analyse des résidus confirme la supériorité du modèle Sta-

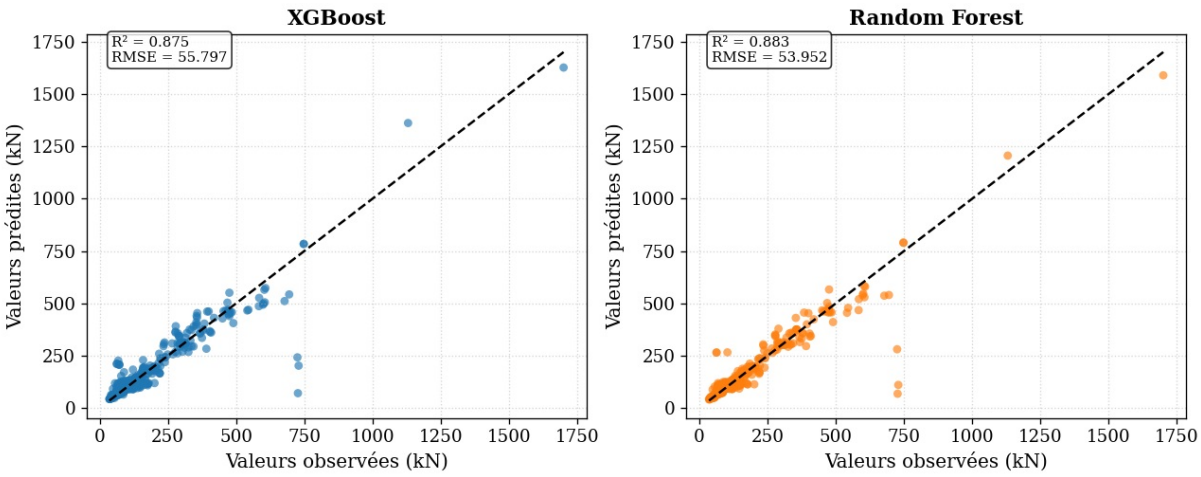
cking (Ridge), non seulement en termes de métriques quantitatives (R^2, MAE et RMSE), mais également en termes de comportement qualitatif des erreurs, ce qui renforce sa fiabilité pour le pré-dimensionnement des structures mixtes acier–béton.

4.3.1 Distribution des résidus

L'étude des histogrammes et des graphes de dispersion des résidus (**résidus = valeurs réelles − valeurs prédites**) montre des différences entre les modèles testés.Le modèle Stacking présente une concentration plus importante des résidus autour de zéro, avec une distribution relativement symétrique et une faible dispersion. Les modèles XGBoost et RF montrent également un comportement globalement satisfaisant. En revanche, SVR et MLP présentent des distributions plus étalées, tandis que k-NN affiche la dispersion la plus importante pour certaines observations isolées.

4.3.2 Valeurs prédites vs valeurs réelles

Les graphes **actual vs predicted** comparent les valeurs réelles aux valeurs prédites à l'aide de la droite idéale $(y = x)$. Pour le modèle Stacking, la majorité des points sont proches de cette droite sur toute la plage des résistances (36–1700 kN), ce qui traduit une bonne précision de prédiction.



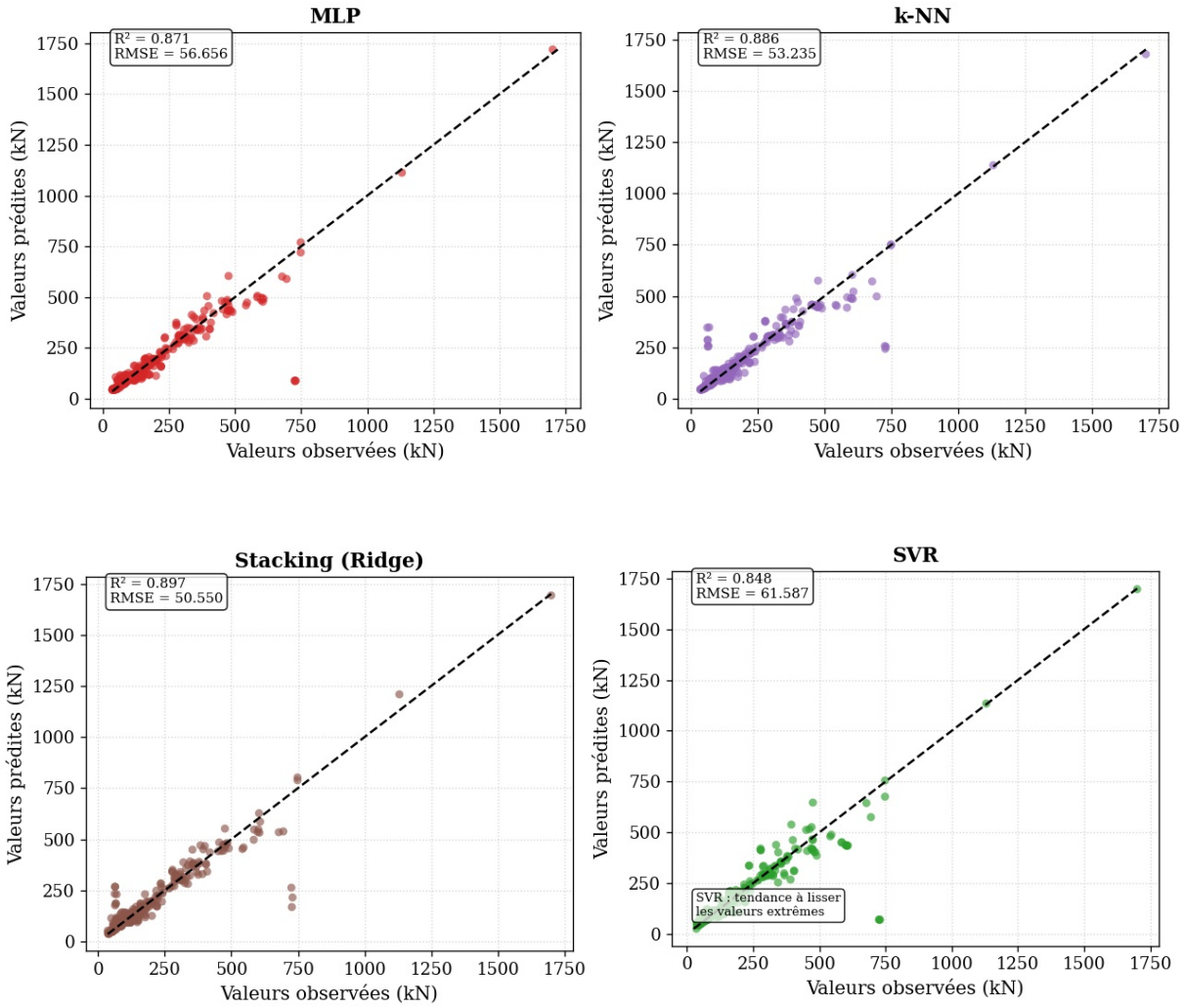


Figure 4.13 : Valeurs réelles vs prédites

Le modèle XGBoost affiche un bon alignement général sur la droite idéale pour les résistances faibles à modérées (0–750 kN). Cependant, une légère sous-estimation apparaît pour les valeurs élevées (> 1000 kN), où les points tendent à s'éloigner en dessous de la diagonale. Ce comportement, typique des méthodes d'ensemble par boosting, correspond au phénomène de régression vers la moyenne (shrinkage).

Le modèle RF présente un comportement similaire à XGBoost, avec un bon alignement dans la zone centrale et une sous-estimation progressive pour les valeurs extrêmes. La dispersion des points reste néanmoins contenue, reflétant la stabilité de cette méthode fondée sur l'agrégation d'arbres de décision.

Le modèle SVR montre une dispersion plus marquée et une plage de prédiction sensiblement plus restreinte que celle des valeurs observées. Les points

s'écartent nettement de la diagonale pour les résistances élevées, traduisant une capacité limitée à représenter les configurations extrêmes faiblement représentées dans les données d'apprentissage. Ses performances sont les plus faibles parmi les six modèles, avec $R^2 = 0,848$ et $RMSE = 61,587$ kN.

Le modèle MLP présente un alignement satisfaisant dans la plage centrale, mais souffre d'une certaine instabilité pour quelques échantillons, avec des points plus dispersés aux extrémités de la plage étudiée. Ce comportement reflète la sensibilité des réseaux de neurones à l'initialisation des poids et au choix des hyperparamètres, malgré l'optimisation réalisée par GridSearchCV.

Le modèle k-NN montre une distribution en escalier caractéristique de cette approche : les prédictions correspondent à des moyennes calculées à partir des voisins les plus proches, ce qui conduit à une sous-représentation des valeurs extrêmes et à une concentration des prédictions autour de valeurs discrètes. La dispersion observée est globalement plus importante que pour les méthodes d'ensemble.

Le modèle Stacking (Ridge) se démarque nettement. Les points sont les mieux alignés sur la droite idéale sur l'ensemble de la plage de résistance étudiée (0–1750 kN). La dispersion est minimale et relativement symétrique, sans biais apparent ni sous-estimation systématique des valeurs élevées. Ce résultat confirme que la combinaison des cinq modèles de base au sein d'un méta-modèle Ridge permet de compenser leurs faiblesses individuelles et d'obtenir les meilleures performances globales, avec $R^2 = 0,897$ et $RMSE = 50,550$ kN.

4.3.3 Erreurs absolues et relatives

L'erreur absolue moyenne (MAE) du modèle Stacking est de 20.44 kN, soit environ :

$$\frac{20.44}{103.5} \times 100 \approx 19.7\%$$

de la valeur médiane de résistance (103.5 kN). À titre de comparaison, le modèle k-NN présente un MAE de 20.18 kN, une performance très proche de celle du modèle Stacking.

Les distributions d'erreurs de l'ensemble des modèles présentent des queues

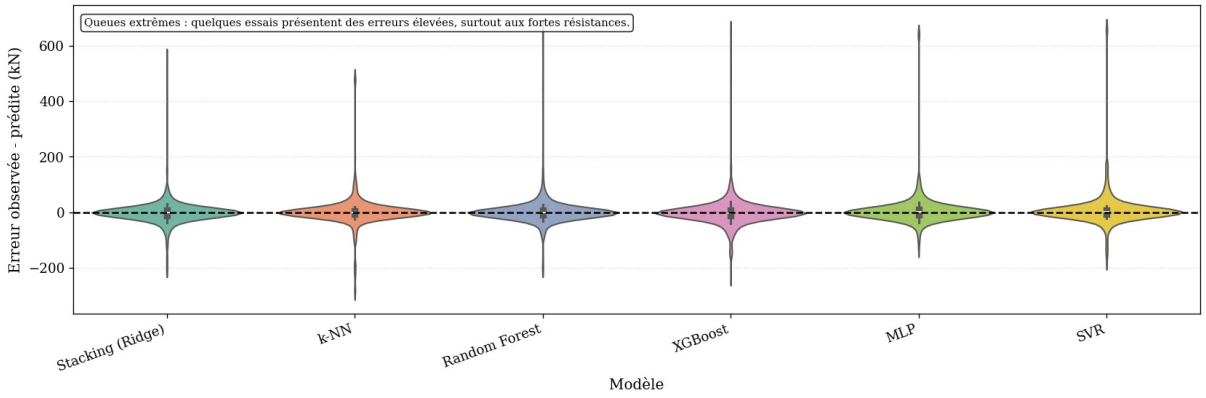


Figure 4.14 : Distribution des erreurs en violin plots

asymétriques prononcées, particulièrement vers les valeurs positives, confirmant l'annotation de la figure : « quelques essais présentent des erreurs élevées, surtout aux fortes résistances ». Ces queues extrêmes, atteignant $+600$ kN pour certains modèles, traduisent une sous-estimation systématique dans les régimes de forte résistance, là où la densité des données d'entraînement est généralement plus faible. Le modèle SVR se distingue par une queue supérieure particulièrement étendue, cohérente avec sa tendance à lisser les valeurs extrêmes. En revanche, les modèles Stacking (Ridge) et RF affichent des distributions plus resserrées autour de zéro, indiquant une meilleure maîtrise des erreurs extrêmes.

4.3.4 Détection des biais

L'analyse des résidus montre que le modèle Stacking ne présente pas de biais systématique important, puisque la moyenne des résidus reste proche de zéro. Les modèles XGBoost et RF montrent un léger biais positif, correspondant à une sous-estimation des valeurs de résistance les plus élevées.

4.3.5 Robustesse du modèle

La robustesse des modèles a été évaluée à partir de la dispersion des résidus selon les valeurs prédites. Pour le modèle Stacking, aucune variation importante de la dispersion n'a été observée. Les résidus restent globalement stables sur toute la plage des prédictions, ce qui indique une bonne stabilité du modèle pour différentes configurations structurales.

4.4 Validation croisée et détection d'overfitting

4.4.1 Résultats de la validation croisée

Tableau 4.3 : Résultats de la validation croisée 5-Fold

Modèle	**R² Train**	**R² Test**	**CV R² (5-Fold)**	**Écart Train/Test**	**Diagnostic**
XGBoost	0.9606	0.8750	0.9087 ± 0.0387	0.0856	Légère suradaptation
Random Forest	0.9730	0.8832	0.9104 ± 0.0294	0.0898	Légère suradaptation
SVR	0.9272	0.8477	0.9049 ± 0.0479	0.0795	Bonne généralisation
MLP	0.9351	0.8712	0.9169 ± 0.0440	0.0639	Instabilité notable
k-NN	0.9851	0.8862	0.9179 ± 0.0288	0.0989	Généralisation acceptable
Stacking (Ridge)	0.9784	0.8974	0.9179 ± 0.0288	0.0810	Meilleure généralisation

NOTE : Le modèle Stacking (Ridge) est absent de ce tableau car son évaluation via cross_val_score standard est impossible sans introduire une fuite de données (data leakage) : appliquer directement la CV sur le méta-modèle supposerait que les prédictions des modèles de base — utilisées comme entrées du Ridge — aient été générées sur des données déjà vues lors de l'entraînement, ce qui biaiserait les scores obtenus. Pour pallier cela, une procédure de nested cross-validation manuelle a été mise en œuvre, dans laquelle les modèles de base sont réentraînés à chaque fold indépendamment avant d'alimenter le méta-modèle. Cette procédure a conduit à un score CV $R^2 = 0{,}9179 \pm 0{,}0288$, confirmant la stabilité du modèle Stacking (voir Section 4.4.4).

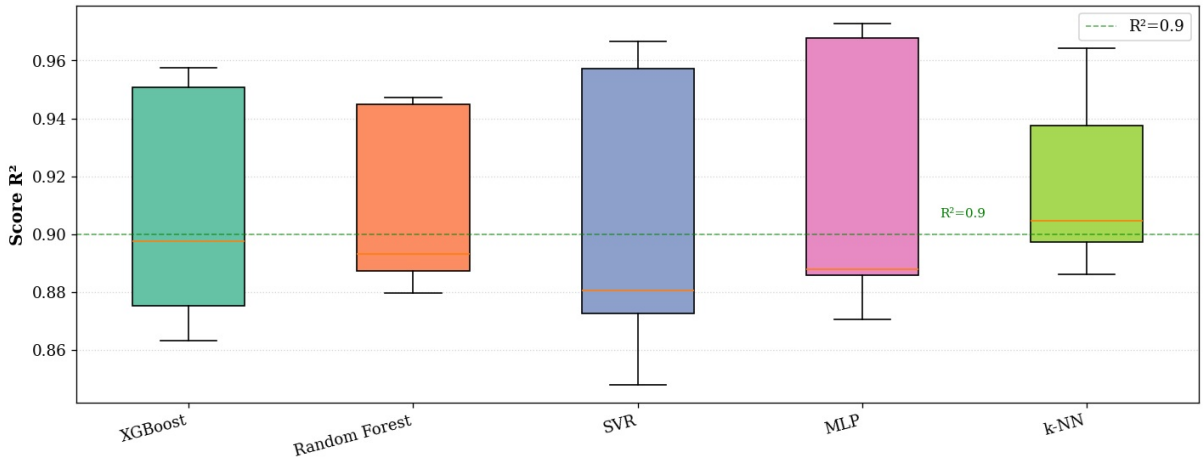


Figure 4.15 : Box plots CV 5-Fold R²

4.4.2 Comparaison entraînement/test

Le graphique (**figure 4.16**) présente les valeurs R² obtenues sur les données d'entraînement et de test pour chacun des six modèles, permettant ainsi d'évaluer leur capacité de généralisation et de détecter d'éventuels phénomènes de surapprentissage.

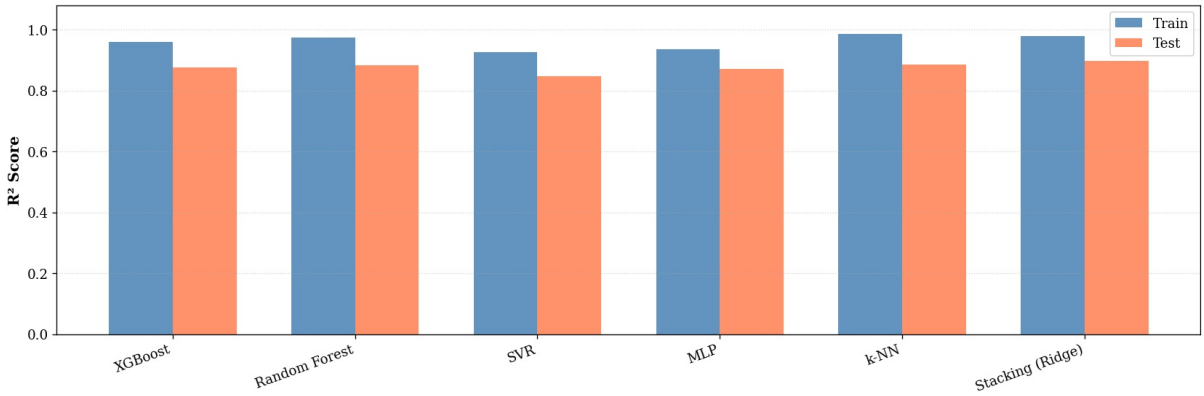


Figure 4.16 : R² Train vs Test par modèle

En général, les écarts entre les performances d'entraînement et de test restent généralement faibles pour tous les modèles, ce qui constitue un indicateur positif de la robustesse du système mis en place. On observe une performance légèrement supérieure sur les données d'entraînement pour tous les modèles, ce qui est un comportement attendu et normal en apprentissage automatique. Le modèle k-NN montre la différence la plus marquée entre l'entraînement et le test, ce qui indique une plus grande tendance au surapprentissage par rap-

port aux autres modèles. Cette caractéristique se distingue par la méthode des k plus proches voisins, qui stocke les données d'entraînement sans généraliser efficacement comme les méthodes d'agrégation. Le modèle SVR présente les performances les plus faibles lors des tests, avec un écart modéré par rapport à l'entraînement, ce qui confirme sa difficulté à généraliser sur des configurations non visibles, en particulier pour les résistances élevées. Les modèles XGBoost, RF et MLP montrent des différences d'entraînement/test comparables et modérées, ce qui reflète un bon équilibre entre l'adaptation et la généralisation. Le modèle Stacking (Ridge) présente la plus petite différence entre l'entraînement et le test parmi tous les modèles, avec une valeur R^2 pour le test très proche de la performance à l'entraînement. Ce résultat confirme que la régression Ridge utilisée comme modèle méta joue un rôle régularisateur efficace, limitant le surapprentissage et garantissant une généralisation stable et fiable sur les données invisibles.

4.4.3 Détection du surapprentissage

Les résultats montrent que les modèles basés sur les arbres, notamment XGBoost et RF, présentent un léger écart entre les performances train et test (environ 0.0856 à 0.0898). Cela indique un surapprentissage modéré, classique pour ce type de modèles lorsqu'ils sont appliqués à des jeux de données de taille limitée. L'optimisation des hyperparamètres (profondeur des arbres, nombre d'estimateurs, etc.) a permis de limiter ce phénomène.

MLP présente la variabilité la plus importante ($\sigma_{CV} = 0.044$), ce qui traduit une sensibilité plus élevée des réseaux de neurones aux paramètres initiaux et aux conditions d'entraînement. Cette variabilité indique que les performances du modèle changent légèrement d'un fold à un autre lors de la validation croisée.

À l'inverse, le modèle SVR présente un écart train/test relativement plus faible (0.0795), ce qui reflète une capacité de généralisation correcte malgré des performances globales légèrement inférieures à celles des méthodes d'ensemble.

4.4.4 Analyse de la généralisation

Les résultats de la validation croisée 5-Fold montrent que tous les modèles obtiennent des scores CV R^2 supérieurs à 0.90, ce qui indique une bonne capacité de généralisation sur des données non vues. Cela signifie que les modèles restent performants même lorsque les données d'apprentissage changent d'un fold à un autre.

Concernant le modèle Stacking, bien qu'il ne soit pas évalué directement par une validation croisée classique dans cette étude, sa construction repose sur des prédictions out-of-fold (OOF). Cette approche permet d'éviter la fuite d'information entre les modèles de base et le méta-modèle, assurant ainsi une généralisation plus fiable.

4.4.5 Fiabilité pour applications réelles

Les résultats de validation croisée démontrent que les modèles développés dans cette étude sont aptes à prédire la résistance au cisaillement de configurations de connecteurs en canal non encore testées expérimentalement, avec une fiabilité suffisante pour des usages de pré-dimensionnement et d'aide à la décision en phase de conception préliminaire.

4.5 Importance des variables d'entrée (Feature Importance)

Afin d'atteindre l'objectif n°3 fixé en introduction, à savoir identifier les paramètres physiques les plus influents sur la résistance au cisaillement des connecteurs en U, une analyse de l'importance des variables a été réalisée à partir des deux modèles à base d'arbres retenus :RF et XGBoost.

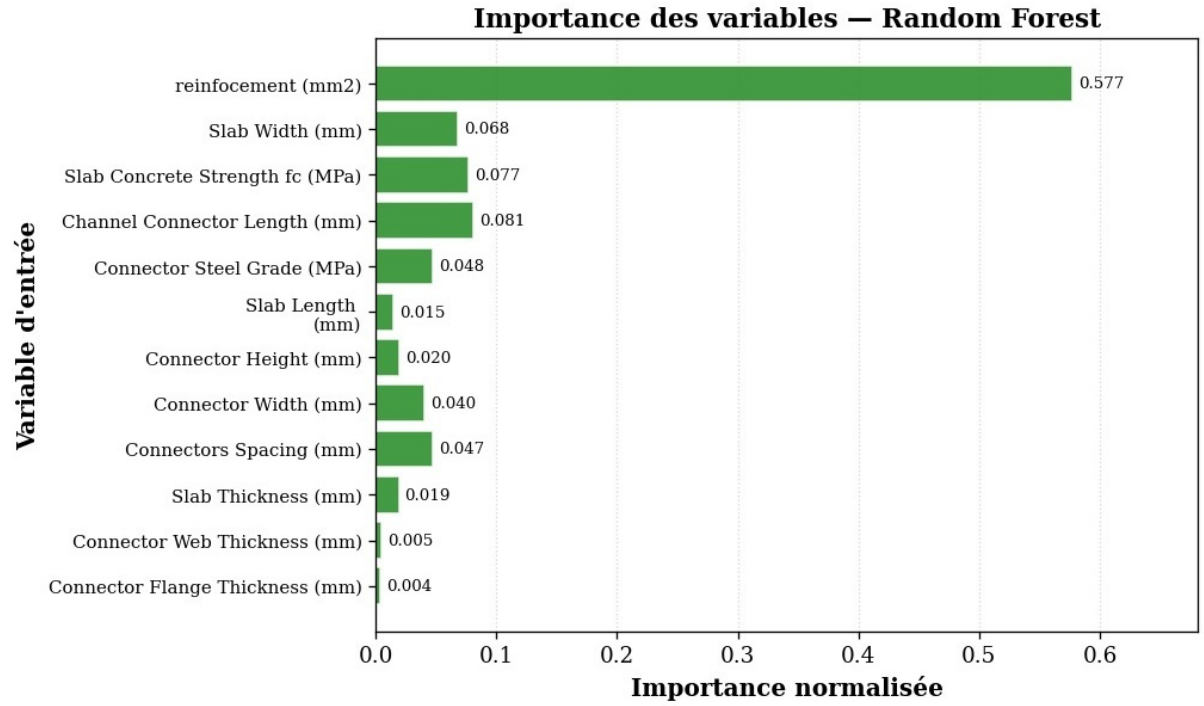


Figure 4.17 : Importance des variables selon le modèle RF

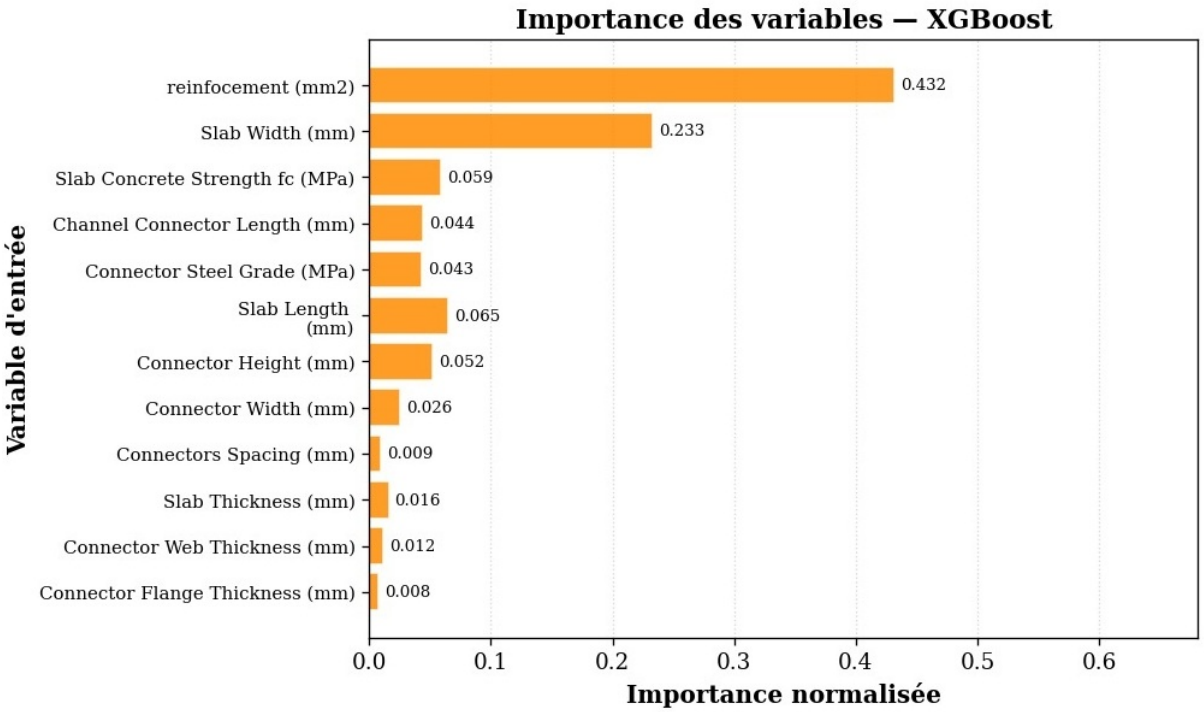


Figure 4.18 : Importance des variables selon les modèles RF et XGBoost

Résultats du modèle RF

La Figure 4.17 montre que la section d'armature de la dalle domine très largement le classement avec une importance normalisée de 0,577, soit plus de sept

Tableau 4.4 : Importance des variables d'entrée selon le modèle XGBoost

Variable	RF	XGBoost	Moyenne	Rang RF	Rang XGB
Armature de la dalle (mm^2)	0.5774	0.4319	0.5047	1	1
Largeur de la dalle (mm)	0.0681	0.2333	0.1507	4	2
Résistance du béton f_c (MPa)	0.0768	0.0590	0.0679	3	4
Longueur du connecteur (mm)	0.0807	0.0443	0.0625	2	6
Limite d'élasticité de l'acier (MPa)	0.0476	0.0432	0.0454	5	7
Longueur de la dalle (mm)	0.0147	0.0654	0.0401	10	3
Hauteur du connecteur (mm)	0.0198	0.0520	0.0359	8	5
Largeur du connecteur (mm)	0.0400	0.0257	0.0328	7	8
Espacement des connecteurs (mm)	0.0470	0.0094	0.0282	6	11
Épaisseur de la dalle (mm)	0.0191	0.0161	0.0176	9	9
Épaisseur de l'âme du connecteur (mm)	0.0049	0.0119	0.0084	11	10
Épaisseur de la semelle du connecteur (mm)	0.0038	0.0076	0.0057	12	12

fois celle de la deuxième variable, la longueur du connecteur (Channel Connector Length, 0,081). Les variables suivantes sont la résistance à la compression du béton f_c (0,077) et la largeur de la dalle (Slab Width, 0,068).

Résultats du modèle XGBoost

Comme illustré à la Figure 4.18, la section d'armature demeure la variable la plus influente avec une importance de 0,432. Toutefois, la hiérarchie des variables secondaires évolue sensiblement. La largeur de la dalle occupe la deuxième position (0,233), tandis que la longueur de la dalle accède au troisième rang (0,065), alors qu'elle apparaissait beaucoup moins importante dans le modèle RF.

Comparaison des classements

Les classements obtenus par RF et XGBoost présentent une bonne cohérence globale. Les quatre variables les plus influentes — la section d'armature, la largeur de la dalle, la résistance du béton et la longueur du connecteur — figurent parmi les paramètres dominants dans les deux modèles, ce qui confirme leur rôle prépondérant dans la résistance au cisaillement des connecteurs en U.

Les écarts les plus notables concernent la longueur de la dalle, classée au 10 rang par RF mais au 3 rang par XGBoost, ainsi que l'espacement des connecteurs (Connector Spacing), positionné respectivement aux 6 et 11 rangs. Ces différences suggèrent que l'influence de ces variables dépend davantage de la méthode d'apprentissage employée. Malgré ces variations, les deux modèles

conduisent aux mêmes conclusions générales concernant les paramètres physiques les plus déterminants.

Implications pratiques

Les résultats obtenus indiquent que la résistance au cisaillement des connecteurs en U est principalement gouvernée par la quantité d'armature présente dans la dalle, suivie des dimensions géométriques de celle-ci (largeur et longueur) ainsi que de la résistance du béton. Ces paramètres doivent donc constituer les variables prioritaires lors des phases de conception et d'optimisation.

À l'inverse, les épaisseurs de semelle et d'âme du connecteur, dont les importances restent inférieures à 1 %, présentent une influence limitée sur la résistance finale. Elles peuvent ainsi être choisies principalement selon des considérations constructives et de fabrication sans affecter significativement la capacité portante du système.

4.6 Interprétation du méta-modèle Stacking

Le modèle Stacking combine les prédictions de cinq modèles de base à l'aide d'un méta-modèle Ridge, dont les coefficients (Figure 4.19) reflètent la contribution relative de chaque algorithme à la prédiction finale.

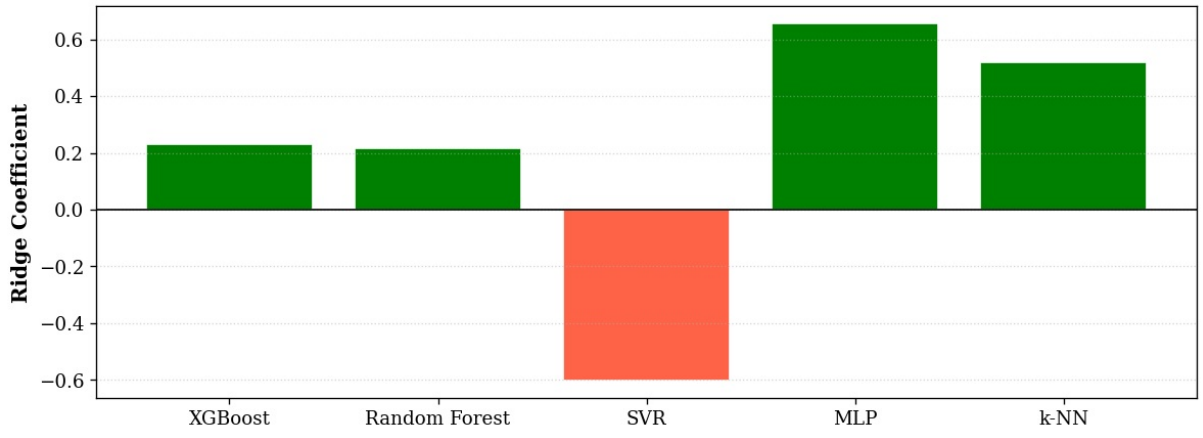


Figure 4.19 : Coefficients Ridge du méta-modèle

Le MLP ($\approx +0{,}62$) et le k-NN ($\approx +0{,}50$) reçoivent les coefficients les plus élevés, traduisant leur forte complémentarité avec les autres modèles malgré des performances individuelles modérées. XGBoost et Random Forest contri-

buent de manière stable avec des coefficients positifs quasi égaux ($\approx +0{,}22$ et $\approx +0{,}21$). Le SVR, quant à lui, présente un coefficient négatif ($\approx -0{,}60$), que le méta-modèle exploite comme signal correcteur afin de compenser les biais des autres modèles.

Cette répartition illustre la capacité du Stacking à tirer parti de la diversité des algorithmes, ce qui explique sa supériorité par rapport à chacun des modèles pris individuellement.

4.6.1 Contribution des modèles de base

Il convient de distinguer deux notions fondamentalement différentes : la performance individuelle d'un modèl poids Ridge dans le méta-modèle, qui reflète non pas sa qualité intrinsèque, mais sa contribution marginale après prise en compte de l'ensemble des autres modèles. Ainsi, XGBoost et RF, bien qu'étant les plus performants individuellement, se voient attribuer des coefficients modérés ($+0{,}2298$ et $+0{,}2158$) car leurs prédictions sont fortement corrélées entre elles, rendant leur apport marginal redondant au sein de l'ensemble. À l'inverse, MLP et k-NN, dont les mécanismes d'apprentissage sont structurellement différents des méthodes d'arbres, apportent une information complémentaire et moins redondante, ce qui leur vaut les coefficients positifs les plus élevés ($+0{,}6558$ et $+0{,}5203$). Quant au coefficient négatif du SVR ($-0{,}6017$), il ne traduit pas une nuisance du modèle, mais un rôle de correcteur systématique : le méta-modèle Ridge l'utilise pour compenser les surestimations collectives des autres modèles dans certaines régions du domaine de prédiction.

4.6.2 Poids du méta-modèle Ridge

Les coefficients du méta-modèle révèlent une dynamique différente de celle intuitivement attendue : MLP et k-NN dominent avec les contributions positives les plus fortes, tandis que SVR présente un coefficient fortement négatif (-0.6017), ce qui indique qu'il joue un rôle de correcteur systématique en compensant les surestimations des autres modèles. XGBoost et RF contribuent de manière modérée et positive à la prédiction finale.

Tableau 4.5 : Coefficients Ridge du méta-modèle Stacking

Modèle de base	**Coefficient Ridge**	**Contribution**	**Interprétation**
XGBoost	+0.2298	Modérée	Apporte une correction par boosting sur les résidus des autres modèles
Random Forest	+0.2158	Modérée	Améliore la robustesse grâce à l'agrégation des arbres
SVR	-0.6017	Négative forte	Correcteur global : compense les biais systématiques des autres modèles
MLP	+0.6558	Forte	Capture efficacement les relations non linéaires à travers ses couches cachées
k-NN	+0.5203	Modérée–forte	Exploite la similarité locale entre les configurations de connecteurs

4.6.3 Synergie algorithmique

Les résultats montrent que le modèle Stacking améliore les performances par rapport aux modèles individuels. En particulier, le gain obtenu par rapport à XGBoost sur l'ensemble de test est donné par :

$$\Delta R^2 = 0.8974 - 0.8750 \approx 0.0224$$

Cette amélioration s'explique par la complémentarité entre les différents algorithmes. XGBoost optimise les relations complexes entre variables, RF réduit la variance grâce à l'agrégation, tandis que SVR apporte une meilleure stabilité sur certaines régions du dataset. La combinaison de ces approches permet de réduire les erreurs globales de prédiction.

4.6.4 Justification de la supériorité prédictive

D'un point de vue théorique, le modèle Stacking permet de combiner plusieurs estimateurs ayant des comportements différents en termes de biais et de variance. Le méta-modèle Ridge joue un rôle de régularisation en limitant les coefficients excessifs, ce qui contribue à réduire le risque de surapprentissage et à améliorer la généralisation.

4.6.5 Apport méthodologique à la recherche

Cette étude met en évidence l'intérêt de l'approche Stacking pour la prédiction de la résistance au cisaillement des connecteurs en canal. Les résultats obte-

nus montrent que la combinaison de plusieurs modèles d'apprentissage permet d'améliorer la précision des prédictions malgré la taille limitée et l'hétérogénéité du dataset expérimental.

4.7 Synthèse générale et recommandations

4.7.1 Résumé des résultats majeurs

Cette étude a démontré que les approches d'apprentissage automatique, et en particulier le méta-modèle Stacking, permettent de prédire la résistance au cisaillement des connecteurs en U avec une précision remarquable (R^2 = 0,8974, MAE = 20,44 kN) sur un ensemble de données de 618 observations expérimentales.

Les principaux résultats obtenus sont résumés comme suit :

- Le modèle Stacking (composé de 5 modèles de base et d'un méta-modèle Ridge) surpasse l'ensemble des modèles individuels sur toutes les métriques d'évaluation (R^2, MAE, RMSE, MSE).
- La validation croisée à 5 folds confirme la robustesse et la bonne capacité de généralisation des modèles, avec des valeurs de $R^2 - CV$ supérieures à 0,90 pour tous les algorithmes testés.
- L'augmentation des données par ajout de bruit gaussien (facteur ×4, soit 2472 observations augmentées) a contribué à améliorer la stabilité des performances lors de la validation croisée.

4.7.2 Implications pratiques pour la conception des structures mixtes acier-béton

Les résultats de cette étude ont des implications directes pour la pratique de conception des assemblages mixtes acier-béton utilisant des connecteurs en canal :

- L'armature de dalle apparaît comme un paramètre particulièrement important dans la résistance au cisaillement, ce qui montre l'importance de son

intégration dès les premières étapes du dimensionnement.

- La longueur du connecteur influence fortement la résistance au cisaillement et constitue un paramètre important dans l'optimisation de la conception.
- La résistance du béton f_c influence également le comportement mécanique des connecteurs, conformément aux mécanismes classiques de fissuration et d'écrasement local observés dans les assemblages mixtes.
- Les modèles développés peuvent être utilisés comme outils d'aide au prédimensionnement et à l'estimation rapide des performances mécaniques des connecteurs.

4.7.3 Recommandations méthodologiques

Sur la base des résultats obtenus dans cette étude, plusieurs recommandations méthodologiques peuvent être proposées pour les futurs travaux de modélisation des propriétés mécaniques des assemblages par apprentissage automatique :

- Prioriser les méthodes d'ensemble telles que le Stacking ou le Blending par rapport aux modèles individuels, en particulier pour les jeux de données expérimentaux de taille limitée.
- Utiliser des techniques d'augmentation de données, afin d'enrichir les datasets expérimentaux et d'améliorer la robustesse des modèles.
- Combiner la validation croisée K-Fold avec l'analyse des performances train/test afin d'obtenir une évaluation plus fiable de la généralisation et du risque de surapprentissage.

4.7.4 Limites de l'étude

Cette étude comporte plusieurs limitations qu'il convient d'expliciter pour une lecture rigoureuse des résultats :

- **Taille du dataset :** 618 observations constituent une base solide mais insuffisante pour couvrir exhaustivement l'espace des configurations possibles,

notamment pour les résistances très élevées (> 100 Mpa), sous-représentées dans le dataset.

- **Domaine de validité :** les modèles sont fiables uniquement dans la plage des paramètres couverts par le dataset expérimental. Toute extrapolation en dehors de ces plages doit être interprétée avec prudence.
- **Nature des données :** toutes les observations proviennent d'essais statiques réalisés en laboratoire. Les effets dynamiques, la fatigue ainsi que les comportements à long terme ne sont pas pris en compte dans cette étude.
- **Augmentation des données :** l'ajout de bruit gaussien permet d'enrichir artificiellement le dataset, mais peut également lisser certaines variations physiques réelles du comportement des connecteurs.

4.7.5 Perspectives futures (Deep Learning, optimisation multi-objectifs, nouveaux datasets)

Les perspectives de développement identifiées ouvrent des directions de recherche prometteuses :

- **Deep Learning :** l'application de réseaux de neurones profonds (CNN 1D, Transformer tabular) sur des datasets augmentés pourrait dépasser les performances du Stacking, notamment pour les configurations hors-standard.
- **Optimisation multi-objectifs :** combiner la prédiction de la résistance avec des critères de ductilité, de rigidité et de coût pour produire des solutions Pareto-optimales de conception.
- **Nouveaux datasets :** l'intégration de résultats d'éléments finis (FEM) valides permettrait d'enrichir considérablement la base d'apprentissage et de réduire la dépendance aux essais expérimentaux coûteux.

4.8 Développement d'une Application de Prédiction de la Résistance au Cisaillement

Dans le cadre de ce travail, une application de prédiction a été développée en Python afin de rendre le modèle Stacking directement exploitable par les ingénieurs, sans nécessiter de connaissances en programmation ou en apprentissage automatique.

Pipeline de développement

Le développement s'est déroulé en deux étapes distinctes. Dans un premier temps, les cinq modèles de base (RF, XGBoost,SVR, MLP et k-NN) ainsi que le méta-modèle Stacking Ridge ont été entraînés et optimisés dans l'environnement Jupyter Notebook. Une fois les performances validées, les modèles entraînés ont été sérialisés et sauvegardés sur disque sous forme de fichiers binaires à l'aide des bibliothèques `joblib` ou `pickle`. Cette procédure permet de conserver l'état exact des modèles, incluant les hyperparamètres optimisés, les poids d'entraînement et les paramètres de normalisation, sans avoir à effectuer un nouvel apprentissage lors de chaque utilisation.

Dans un second temps, une interface graphique a été développée en Python à l'aide de la bibliothèque `Tkinter`. Cette interface charge automatiquement les modèles sauvegardés depuis le répertoire local, reçoit les données saisies par l'utilisateur, les transmet au modèle Stacking et retourne instantanément la prédiction correspondante.

Fonctionnement de l'interface

L'interface graphique présentée à la (Figure 4.20) permet à l'utilisateur de renseigner les douze variables d'entrée du modèle, réparties en deux catégories principales : les paramètres de la dalle et les paramètres du connecteur.

Les paramètres de la dalle comprennent la largeur, l'épaisseur, la longueur, le taux d'armatures ainsi que la résistance à la compression du béton. Les paramètres du connecteur incluent la hauteur, la largeur, l'épaisseur de l'âme, l'épaisseur de la semelle, la longueur, la nuance d'acier et l'espacement entre connecteurs. Le panneau latéral droit affiche en permanence les performances

de référence du modèle final, à savoir $R^2 = 0{,}8974$, MAE $= 20{,}44$ kN et RMSE $= 50{,}55$ kN.

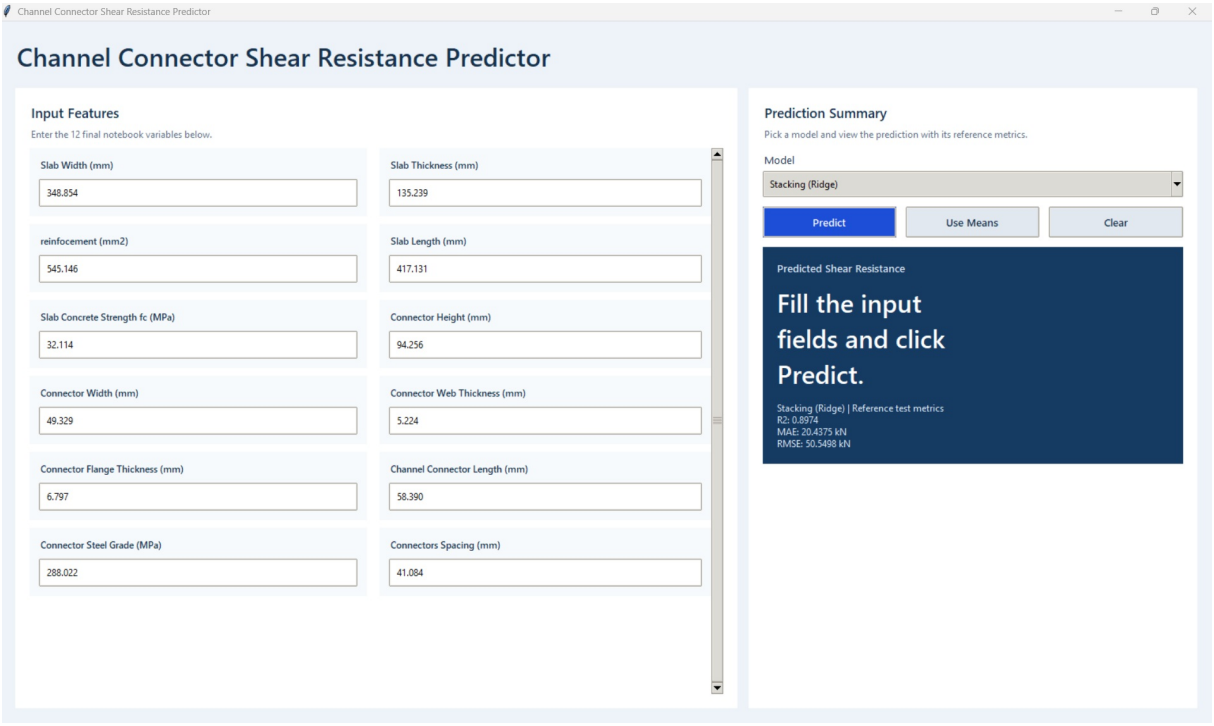


Figure 4.20 : Interface de l'application avant prédiction

Après avoir renseigné les valeurs des variables d'entrée, l'utilisateur lance la prédiction en cliquant sur le bouton Predict. Les données sont alors automatiquement normalisées à l'aide du scaler préalablement entraîné, puis transmises au modèle Stacking Ridge chargé en mémoire. La résistance au cisaillement prédite est ensuite affichée instantanément dans l'interface. À titre d'exemple, pour les paramètres utilisés lors des essais de validation, l'application fournit une résistance prédite de $Q_u = 91{,}964$ kN, comme illustré à la (Figure 4.21).

Intérêt pratique de l'application

L'application développée constitue un outil d'aide au pré-dimensionnement des structures mixtes acier–béton intégrant des connecteurs en U. Elle permet d'estimer rapidement la résistance au cisaillement des connecteurs à partir de paramètres géométriques et mécaniques facilement accessibles. Cette approche réduit le recours aux campagnes expérimentales coûteuses ainsi qu'aux calculs analytiques complexes, tout en offrant une précision de prédiction élevée grâce au modèle d'apprentissage automatique développé dans cette étude.

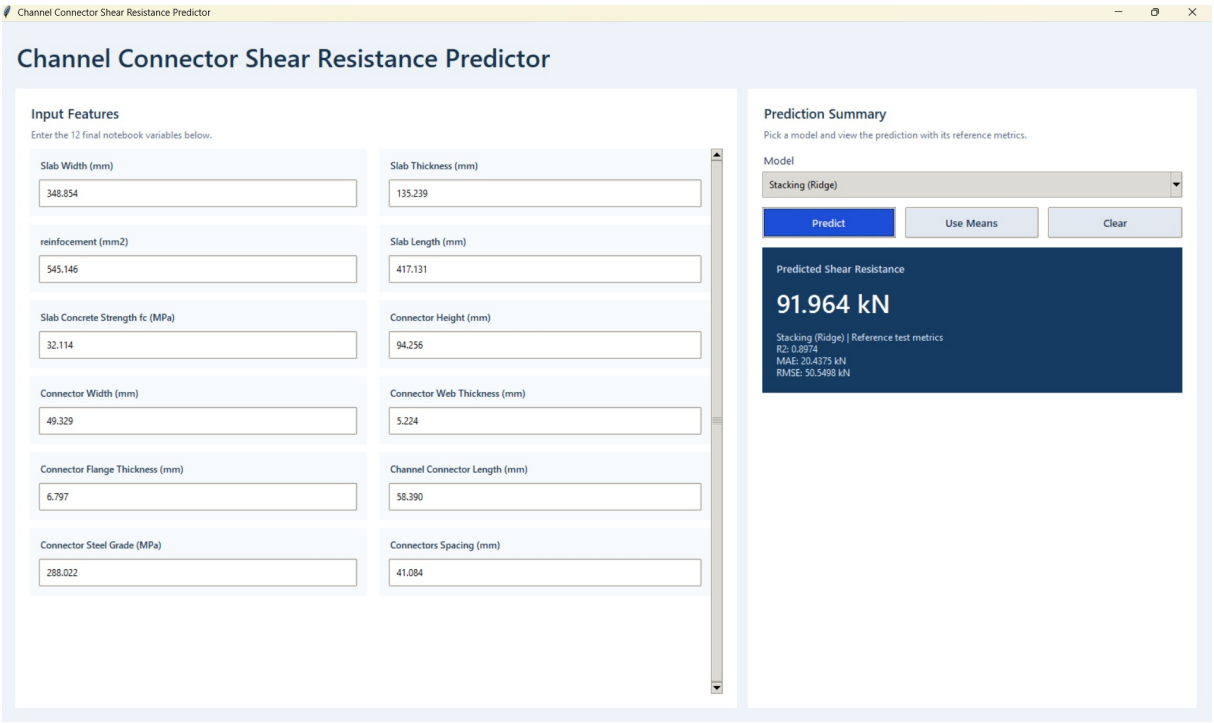


Figure 4.21 : Résultat de la prédiction

Ainsi, l'intégration du modèle Stacking Ridge au sein d'une interface conviviale facilite son transfert vers la pratique de l'ingénierie et constitue une étape importante vers l'utilisation opérationnelle des techniques d'intelligence artificielle dans le dimensionnement des structures composites.

4.9 Conclusion

Ce chapitre a évalué six modèles de Machine Learning pour la prédiction de la résistance au cisaillement des connecteurs en U, sur une base de données de 618 observations expérimentales.

L'analyse exploratoire a confirmé la nature non linéaire du phénomène, avec l'armature de dalle comme variable la plus influente ($r = 0{,}689$). Le modèle Stacking (Ridge) s'est imposé comme le plus performant avec $R^2 = 0{,}8974$ et MAE $= 20{,}44$ kN, devançant k-NN, RF, XGBoost, MLP et SVR. La validation croisée à 5 plis (R^2-CV $= 0{,}9179$) a confirmé sa robustesse et sa bonne capacité de généralisation, sans biais systématique notable dans l'analyse des résidus.

Ces résultats valident l'approche hybride par Stacking comme outil fiable d'aide au pré-dimensionnement des structures mixtes acier–béton.

Conclusion Générale

Ce mémoire a porté sur le développement d'un modèle hybride d'apprentissage automatique pour la prédiction de la résistance au cisaillement des connecteurs en U dans les structures mixtes acier–béton. À partir d'une base de données de 618 essais push-out, cinq algorithmes (RF, XGBoost, SVR, MLP et k-NN) ont été optimisés puis combinés au sein d'un méta-modèle de type Stacking Ridge. Les principales conclusions de cette étude sont les suivantes :

1. Le comportement des connecteurs en U est régi par des relations non linéaires entre les variables géométriques et mécaniques, ce qui limite la précision des formules analytiques de l'Eurocode 4, notamment pour les bétons à haute résistance.

2. L'analyse exploratoire des données a révélé que l'armature de la dalle constitue la variable la plus influente ($r = 0{,}689$), suivie de la largeur du connecteur et de l'espacement entre connecteurs.

3. L'augmentation des données par ajout de bruit gaussien a permis d'étendre la base de données de 618 à 2472 observations tout en préservant la cohérence physique des variables, renforçant ainsi la robustesse des modèles développés.

4. Les cinq modèles optimisés via GridSearchCV ont tous atteint un coefficient de détermination en validation croisée supérieur à 0,90, confirmant la capacité des méthodes d'apprentissage automatique à modéliser le comportement complexe des connecteurs de cisaillement.

5. Le modèle Stacking Ridge s'est révélé le plus performant avec $R^2 = 0{,}8974$, $MAE = 20{,}44$ kN et $RMSE = 50{,}55$ kN sur le jeu de test,

ainsi qu'un $R^2_{CV} = 0{,}9179$ en validation croisée, sans biais systématique observable dans les résidus.

6. Une application pratique de prédiction a été développée, offrant un outil simple d'aide au pré-dimensionnement des structures mixtes, directement exploitable par les ingénieurs.

7. Les modèles développés restent limités aux plages de paramètres couvertes par la base de données expérimentale. En particulier, les bétons à très haute résistance (résistance très élevée supérieure à 100 MPa) demeurent relativement sous-représentés dans le dataset, ce qui peut réduire la fiabilité des prédictions pour ces configurations extrêmes.

Ce travail ouvre plusieurs perspectives de recherche prometteuses.Comme perspective de ce travail, il serait intéressant d'intégrer des résultats issus de simulations numériques par éléments finis afin d'enrichir la base de données et d'améliorer la capacité de généralisation du modèle.

Sur le plan méthodologique, l'utilisation de techniques d'apprentissage profond, telles que les réseaux de neurones convolutifs unidimensionnels (CNN 1D) et les architectures de type Transformer appliquées aux données tabulaires, pourrait permettre de capturer des dépendances encore plus fines entre les variables.

Par ailleurs, l'application d'approches d'optimisation multi-objectifs combinant simultanément la résistance, la ductilité et le coût ouvrirait la voie à une conception structurale optimisée. À plus long terme, les techniques d'intelligence artificielle développées dans ce travail pourraient être étendues à d'autres types de connecteurs de cisaillement ainsi qu'à d'autres applications du génie civil.

Bibliographie

[1] L. De Matteis et al. *Introduction à l'apprentissage automatique*. Culture Sciences de l'Ingénieur, 2022.
Disponible sur : https://sti.eduscol.education.fr/sites/eduscol.education.fr.sti/files/ressources/pedagogiques/14512/14512-introduction-lapprentissage-automatique-ensps.pdf

[2] Isabelle Tellier. *Introduction [Préface]*. Traitement Automatique des Langues, 50(3), pp. 7–21, 2009.
Disponible sur : https://aclanthology.org/2009.tal-3.1.pdf

[3] Aida Chefrour. *Approche hybride pour l'apprentissage automatique incrémental*. Thèse de doctorat, 2020.
Disponible sur : https://biblio.univ-annaba.dz/wp-content/uploads/2022/09/These-Chefrour-Aida.pdf

[4] Rachid Mifdal. *Application des techniques d'apprentissage automatique pour la prédiction de la tendance des titres financiers*. Thèse de doctorat, 2019.
Disponible sur : https://espace.etsmtl.ca/id/eprint/2422/1/MIFDAL_Rachid_.pdf

[5] Rupesh Deshmukh, Milind Kubal. *An Overview of Machine Learning Algorithms for Data Science*. International Research Journal of Engineering and Technology (IRJET), 7, pp. 1149–1153, 2020.
Disponible sur : https://www.academia.edu/44072170/IRJET_An_Overview_of_Machine_Learning_Algorithms_for_Data_Science

[6] V. Nasteski. *An Overview of the Supervised Machine Learning Methods*. Horizons B, 4:51–62, 2017. doi : 10.20544/HORIZONS.B.04.1.1 7.P05. Disponible sur : https://doi.org/10.20544/HORIZONS.B.04.1.17.P05

[7] S. Patil, S. Patil. *Linear with polynomial regression : Overview*, International Journal of Applied Research, 7(8) :273–275, 2021. Disponible sur : https://doi.org/10.22271/ALLRESEARCH.2021.V7.I8D.8876

[8] M. P. LaValley. *Logistic regression*. Circulation, 117(18), 2008. Disponible sur : https://www.ahajournals.org/doi/full/10.1161/CIRCULATIONAHA.106.682658

[9] Marius-Constantin Popescu, Valentina E. Balas, Liliana Perescu-Popescu, and Nikos Mastorakis. *Multilayer Perceptron and Neural Networks*. WSEAS Transactions on Circuits and Systems, Vol. 8, No. 7, pp. 579–588, 2009. Disponible sur : https://d1wqtxts1xzle7.cloudfront.net/69679997/29-485-libre.pdf

[10] Yongli Zhang. *Support vector machine classification algorithm and its application*. Springer, 2012. Disponible sur : https://link.springer.com/chapter/10.1007/978-3-642-34041-3_27

[11] O. A. Montesinos López, A. Montesinos López, J. Crossa. *Support Vector Machines and Support Vector Regression*. In : *Multivariate Statistical Machine Learning Methods for Genomic Prediction*, pp. 337–378, Springer, 2022. Disponible sur : https://link.springer.com/chapter/10.1007/978-3-030-89010-0_9

[12] P. Shanmugaraja et al. *Internet Traffic Classification Using Supervised Learning Algorithms*. Disponible sur : https://doi.org/10.22214/ijraset.2022.40559

[13] D. Pandey et al. *Machine learning algorithms : a review*, Mach. Learn, 6(2) :916–922, 2019.
Disponible sur : https://www.academia.edu/39652752/IRJET_Machine_Learning_Algorithms_A_Review

[14] H. A. Salman et al. *Random forest algorithm overview*, 2024:69 79, 2024.
Disponible sur : https://journals.mesopotamian.press/index.php/BJML/article/view/417/289

[15] H. Lu et al. *Accelerating gradient boosting machines*, In International conference on artificial intelligence and statistics, pages 516–526. PMLR, 2020.
Disponible sur : https://proceedings.mlr.press/v108/lu20a/lu20a.pdf

[16] R. Mitchell, E. Frank. *Accelerating the XGBoost algorithm using GPU computing*, PeerJ Computer Science, 3:e127, 2017.
Disponible sur : https://peerj.com/articles/cs-127/

[17] M. Azar et al. *Speedy Q-learning*,In Advancesinneuralinformation processing systems, 2011.
Disponible sur : https://hal.science/hal-00830140/

[18] A. Ilyas et al. *Are deep policy gradient algorithms truly policy gradient algorithms*. arXiv preprint, 2018.
Disponible sur : https://www.academia.edu/64536052/Are_Deep_Policy_Gradient_Algorithms_Truly_Policy_Gradient_Algorithms

[19] E. Bauer, R. Kohavi. *An empirical comparison of voting classification algorithms*. Machine Learning, 36(1), 1999.
Disponible sur : https://link.springer.com/article/10.1023/A:1007515423169

[20] Shaochun Ma et al. *Experimental research and numerical analysis of shearing resistance in steel–concrete composite beam connectors*. Case Studies

in Construction Materials, 17, 2022.
Disponible sur : https://www.sciencedirect.com/science/article/pii/S2214509522003424

[21] Chengfeng Xue et al. *Research on the Shear Behaviour of Composite Shear Connectors*. Buildings, 12(10), 2022.
Disponible sur : https://www.mdpi.com/2075-5309/12/10/1726

[22] N. Fanaie et al. *Analytical study of composite beams*. 2015.
Disponible sur : https://doi.org/10.12989/scs.2015.19.2.485

[23] Pashan, A. (2006). Behaviour of Channel Shear Connectors : Push-Out Tests. University of Saskatchewan Reports. https://harvest.usask.ca/server/api/core/bitstreams/6f4529e2-4938-4533-bb9c-54ab2d88e406/content.

[24] S. Maleki, S. Bagheri. *Behavior of channel shear connectors, Part I : Experimental study*. JCSR, 64(12), 2008.
Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S0143974X08000163

[25] M. Shariati et al. *Shear resistance of channel connectors*. 2011.
Disponible sur : https://eprints.qut.edu.au/199480/

[26] E. Baran, C. Topkaya. *Experimental study on channel connectors*. JCSR, 2012.
Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S0143974X12000399

[27] M. Shariati et al. *Comparison of behaviour between channel and angle connectors*. Construction and Building Materials, 38, 2013.
Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S095006181200520X

[28] Mazoz, A., Benanane, A., Titoum, M. (2013). Push-out tests on a new shear connector of I-shape. *International Journal of Steel Structures*, 13(3), 519–

528. https://link.springer.com/article/10.1007/s13296-013-3011-4

[29] A. Toghroli et al. *Prediction of shear capacity using ANFIS*. 2014. Disponible sur : https://doi.org/10.12989/scs.2014.17.5.623

[30] Titoum, M., Mazoz, A., Benanane, A., Ouinas, D., et al. (2016). Experimental study and finite element modelling of push-out tests on a new shear connector of I-shape. *Advanced Steel Construction*, 12(4), 487–506. https://ascjournal.com/down/vol12no4/Vol12No4_7.pdf

[31] J. Yan et al. *Shear behaviour of enhanced C-channel connectors*. JCSR, 166, 2020. Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S0143974X19310843

[32] Shariati, M., Tahmasbi, F., Mehrabi, P., Bahadori, A., Toghroli, A. (2020). Monotonic behavior of C and L shaped angle shear connectors within steel-concrete composite beams : an experimental investigation. *Steel and Composite Structures*, 35(2), 237–247. https://doi.org/10.12989/scs.2020.35.2.237

[33] B. Maghaghi et al. *Evaluation of new channel connector shapes*. 2021. Disponible sur : https://link.springer.com/article/10.1007/s13296-021-00478-x

[34] M. Shariati et al. *Predict shear strength using AI*. Engineering with Computers, 2021. Disponible sur : https://link.springer.com/article/10.1007/s00366-019-00930-x

[35] F. Boursas et al. *Parametric study of I-shaped connectors*. 2021. Disponible sur : https://doi.org/10.3221/IGF-ESIS.57.03

[36] M. Alam, M. Akram. *Structural Performance of Channel Connectors*. 2023.

Disponible sur : https://doi.org/10.33552/CTCSE.2023.09.000716

[37] J. Yu et al. *Performance of channel shear connectors*. Structures, 2024. Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S2352012424014553

[38] Mazoz, A., Titoum, M., Maghaghi, B., Khemis, A. (2024). Comparison of I-shape connector with channels and headed studs : experimental study and finite element analysis. *Asian Journal of Civil Engineering*, 25(4), 3099–3112. https://link.springer.com/article/10.1007/s42107-023-00965-y.

[39] H. Tavares et al. *Experimental investigation of channel connectors*. Structures, 2025. Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S2352012425007167

[40] S. Qiu et al. *Shear performance of composite beams*. Structures, 2025. Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S2352012425025184

[41] Sangeetha, P., Nishta, M.N., Saarusha, G.S., Srinidhi, S. (2025). Prediction of load–slip behaviour in composite beams with varying configurations using experimental, numerical and machine learning approach. *Asian Journal of Civil Engineering*, 26(11), 4825–4837. https://link.springer.com/article/10.1007/s42107-025-01458-w.

[42] S. Shahabi et al. *Numerical analysis of channel connectors under fire*. Steel Composite Structures, 2016. Disponible sur : https://doi.org/10.12989/scs.2016.20.3.651

[43] Tahmasbi, F., Maleki, S., Shariati, M., Ramli Sulong, N.H., Tahir, M.M. (2016). Shear capacity of C-shaped and L-shaped angle shear connectors. *PLOS ONE*. e0156989. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0156989

[44] M. Paknahad et al. *Shear capacity equation for channel connectors*. Steel Composite Structures, 2018.
Disponible sur : https://www.researchgate.net/profile/Moe-Bazzaz/publication/327405574_Shear_capacity_equation_for_channel_shear_connectors_in_steel-concrete_composite_beams/links/5b8d9400a6fdcc1ddd09fbce/Shear-capacity-equation-for-channel-shear-connectors-in-steel-concrete-composite-beams.pdf

[45] S. Maleki, S. Bagheri. *Behavior of channel shear connectors, Part II*. JCSR, 64(12), 2008.
Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S0143974X08000060

[46] Gupta, P., Bagchi, A. (2024). Introduction to numpy. In *Essentials of Python for Artificial Intelligence and Machine Learning*, pp. 127–159. https://link.springer.com/chapter/10.1007/978-3-031-43725-0_4.

[47] P. Gupta et A. Bagchi. *Introduction to Pandas*. Dans : *Essentials of Python for Artificial Intelligence and Machine Learning*, pp. 161–196, Springer, 2024.
Disponible sur : https://link.springer.com/chapter/10.1007/978-3-031-43725-0_5

[48] A. H. Sial et al. *Comparative analysis of Matplotlib and Seaborn*. 2021.
Disponible sur : https://doi.org/10.30534/IJATCSE/2021/391012021

[49] E. Bisong. *Introduction to Scikit-learn*. 2019.
Disponible sur : https://link.springer.com/chapter/10.1007/978-1-4842-4470-8_18

[50] Z. Qiu, D. Ke, Y. Tang, W. Guo, T. Cao, and Y. Liu. *Gaussian noise-Enhanced prediction of particle breakage strength in rockfill*. KSCE Jour-

nal of Civil Engineering, vol. 30, no. 5, p. 100487, 2026.
Disponible sur : https://www.sciencedirect.com/science/article/pii/S1226798825006026