Prédiction de qualité de soudure

- Apprentissage Automatique -

Lien GitLab

-Rapport de projet -

CentraleSupélec - Université Paris-Saclay, Gif-Sur-Yvette

30 octobre 2024

Groupe 3

BOUAITA Rayane - DAVID Erwan - EL ANATI Pierre - FAYNOT Guillaume - TRIER Gabriel

rayane.bouaita@student-cs.fr

erwan.david@student-cs.fr

pierre.el-anati@student-cs.fr

guillaume.faynot@student-cs.fr

gabriel.trier@student-cs.fr

CentraleSupélec

université PARIS-SACLAY

Résumé du projet

Dans le contexte industriel actuel, la qualité des soudures joue un rôle fondamental pour la tenue des structures. C'est le cas dans des secteurs critiques comme l'aérospatiale, la construction navale ou l'énergie nucléaire. Ces industries sont fortement dépendantes de l'expertise des soudeurs et à la vérification de la qualité des soudures, principalement basée sur l'expérience. Or, avec le développement des techniques d'intelligence artificielle et de l'accès croissant aux données, une opportunité se présente pour capturer, standardiser et enrichir ce contrôle qualité.

Table des matières

I. Introduction

a. Problématique du projet

La problématique centrale du projet réside dans la prédiction de la qualité des soudures, qui jusqu'à présent repose principalement sur un contrôle par expérience des soudeurs. La question est donc de savoir comment, à partir de données mesurées, on peut prédire de façon fiable cette qualité. Et dans un autre sens, déterminer quels éléments empiriques liés au soudage expliquent quantitativement la qualité intrinsèque d'une soudure. Ainsi, ce défi soulève plusieurs questions :

- Quelles sont les variables les plus pertinentes pour évaluer la qualité de la soudure ?
- Comment prétraiter ces données pour fournir par la suite une prédiction précise, y compris dans un contexte où certaines données ne sont pas entièrement étiquetées ?
- Quelles méthodes de Machine Learning peuvent permettre d'automatiser et d'améliorer ce processus ?

b. Traduction en un problème de Machine Learning (ML)

La problématique de prédiction de la qualité des soudures peut être traduite en un problème de ML, ici de régression ou de classification, selon la nature des variables cibles (continue ou discrète) qui associe les caractéristiques mesurables des soudures (température, composition chimique des aciers, grandeurs électriques, etc.) à une qualité prédite. Les différentes étapes de résolution de ce problème sont :

- Analyse statistiques descriptives des données.
- Nettoyages et transformation de certaines données pour être exploitables par les futurs modèles de ML.
- Imputation des données manquantes dans le dataset.
- Réduction de la dimensionnalité.
- Modélisations prédictives seront mises en place. Plusieurs algorithmes de ML seront appliqués pour prédire la qualité des soudures.
- Comparaison des prédictions suivants différentes méthodes.

c. Notions de soudage

Le soudage désigne un procédé d'assemblage de matériaux, principalement des métaux, par fusion. La qualité d'une soudure est évaluée sur plusieurs critères, et plus traditionnellement aux moyens de test de propriétés mécaniques. C'est la région du cordon de soudure qui est testée par ces méthodes. Le test de traction en est l'exemple le plus fréquemment utilisé. Il consiste en un allongement d'une éprouvette par application d'un effort de traction sur une machine jusqu'à déformation (*Yield Strength*), rupture (*Ultimate Yield Strength*). On note également l'élongation (*Elongation*) et la réduction de surface (*Reduction of area*). Cet essai mécanique permet de s'assurer que la soudure est à la fois fonctionnelle et durable, il est couteux en temps et financièrement, car il nécessite du matériel de précision.

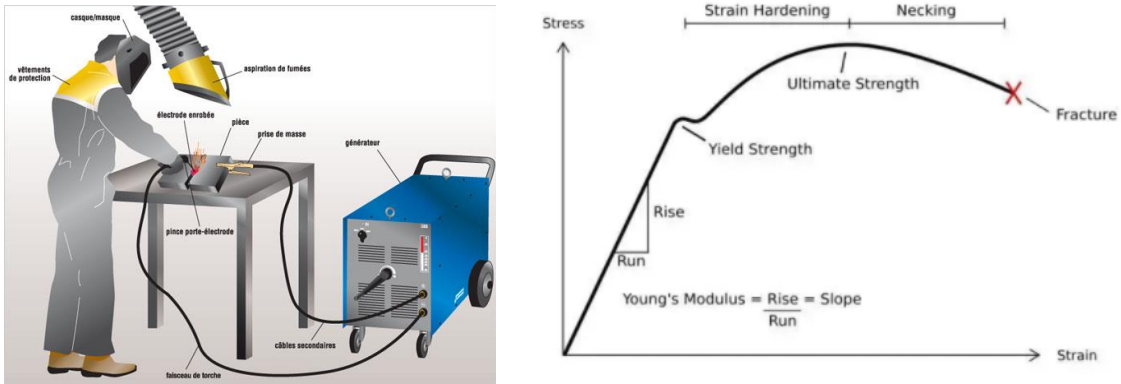

Figure 1 : Schéma de principe du soudage et Allure d'une courbe de traction

II. Stratégie

Afin de prédire la qualité de la soudure nous utiliserons les résultats d'un test de traction. Ce test contient la plupart des informations nécessaires pour prédire la qualité d'une éprouvette. Cependant il peut être long et couteux de le faire de manière fréquente pour les industriels. Nous allons donc économiser ce test aux industriels en en prédisant une variable principale. Nous nous intéressons à la prédiction de la variable limite élastique (*Yield Strength*). Cette variable quantifie la contrainte nécessaire pour commencer à déformer plastiquement notre matériau un fois soudé : plus elle est élevée, plus le matériau est dur.

III. Prétraitement

a. Description de la base de données et tri primaire des features

Le dataset est composé de 44 features et de 1652 échantillons. Beaucoup de ces features n'interviennent pas dans la qualité de la soudure. Notons qu'il y a 46% de valeurs manquantes, donc la méthode d'imputation est déterminante dans la prédiction. Dans la première étape de prétraitement, nous sélectionnons uniquement les features avec un taux de valeurs manquantes acceptable, soit inférieur à 65%, et supprimant les autres. Ce seuil est défini en fonction de la répartition des valeurs manquantes pour faciliter par la suite l'imputation. Supprimer certaines features permet de limiter l'overfitting. De plus, cela peut être important car une imputation excessive peut introduire du bruit ou des biais nuisant à la qualité de la prédiction finale.

b. Imputation précise

Une imputation précise peut être effectué pour certaine valeurs de *Yield strengh* et *Charpy impact toughness*. Cette imputation est possible avec les variables très fortement corrélés entre elles. Nous avons alors identifié deux paires : pour *Yield strength*, en utilisant les valeurs de *UTS*, et pour *Charpy impact toughness*, en utilisant *Reduction of Area*.
Pour chaque cas, nous procédons de la manière suivante :

1. Nous identifions les données complètes pour les deux variables afin de créer un ensemble d'entraînement.
2. Nous entraînons un modèle de régression linéaire où la variable corrélée (ex: *UTS*) est utilisée pour prédire la variable manquante (ex: *Yield strength*).
3. Les prédictions du modèle sont ensuite utilisées pour combler les valeurs manquantes.

c. Etude de la corrélation

Nous allons continuer notre réduction du nombre de features en ne gardant qu'une feature parmi celles corrélées ensemble à plus de 82% (celle avec le moins de valeurs manquantes). Ce chiffre a été choisi empiriquement. Cela permet de réduire la redondance dans le dataset, pour possiblement améliorer les performances des modèles de prédiction et éviter le surapprentissage. Une matrice de corrélation est réalisée (Fig.2) après suppression des features corrélées (mais avant suppression des features issues du test de traction). Nous remarquons que tous les résultats issus du test de traction sont fortement corrélés entre eux.

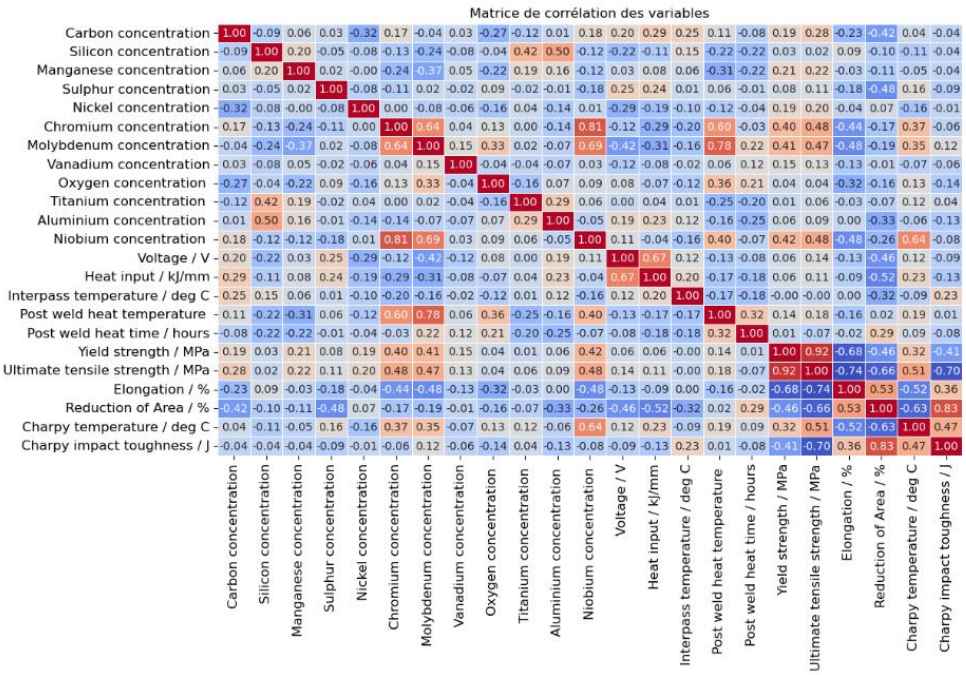

Figure 2 : Matrice de corrélation

d. Faible variance & WeldID

Finalement, nous supprimons la feature catégorielle WeldID qui comprend trop de catégories différentes et ne semble pas pertinent (variable d'identification). Nous supprimons aussi les quelques features dont la variance est inférieure à 0.01. Cela permet de réduire la complexité du modèle en supprimant les variables non informatives.

IV. Imputation

a. Imputation en masse

Notre première approche consiste à imputer toutes les features restantes après le pré-traitement en utilisant différentes méthodes : 20 features restantes.

Features catégoriques : Imputation par la mode (valeur la plus fréquente), assurant ainsi la cohérence avec la majorité des données tout en minimisant l'introduction de biais.

Features numériques : Imputation par KNN, régression (Iterative Imputer), MICE (modélisation itérative tenant compte des relations entre variables), médiane et moyenne. Chaque méthode a été appliquée de manière systématique sur toutes les colonnes numériques générant ainsi plusieurs dataset différents que nous utiliserons pour évaluer les performances avec différents algorithmes. Cette approche globale a l'avantage de simplifier le processus d'imputation, mais elle peut introduire des biais pouvant affecter la qualité des prédictions.

b. Imputation selon distribution des données

Une seconde approche plus précise a été réalisée en fonction de la distribution des colonnes (outliers, écart type, distribution symétrique ou asymétrique) et de leur corrélation avec d'autres variables :

Imputation par la médiane : Cette méthode a été choisie pour les colonnes présentant une distribution asymétrique avec de nombreux outliers. Étant une mesure robuste, elle n'est pas influencée par les valeurs extrêmes, ce qui permet d'éviter que ces dernières ne biaisent les résultats. Cette approche a été appliquée pour des colonnes comme la concentration en *Nickel* et *Vanadium*, où la distribution est fortement asymétrique, permettant ainsi d'obtenir une imputation plus fiable.

Imputation par la moyenne : La moyenne a été appliquée aux features avec une distribution symétrique, peu d'outliers, et peu de valeurs manquantes, comme *Post weld heat tresatment temperature* et *Post weld heat treatment time* afin de limiter le risque.

Imputation par KNN (K-Nearest Neighbors) : Le KNN a été privilégié pour les colonnes avec un taux élevé de valeurs manquantes, en particulier lorsque les colonnes montraient des similitudes avec d'autres variables. KNN, robuste aux outliers, a été choisi pour des features comme *Charpy temperature* et *Charpy impact toughness.*

Imputation par régression : Finalement, nous avons choisi l'imputation par régression pour les features avec un fort taux de valeurs manquantes et une corrélation élevée avec d'autres variables. Cette méthode permet de mieux capturer les relations linéaires entre les variables, comme pour la concentration en *Chromium* et *Molybdenium*, qui montrent de fortes corrélations avec d'autres features.

V. Réduction de dimension

Dans cette partie, nous avons appliqué une méthode de réduction de dimensions, à savoir la PLS. L'objectif principal de cette étape est de simplifier l'ensemble de caractéristiques tout en maximisant la variance expliquée dans la variable cible « *Yield strength / MPa* ».
Avant d'appliquer la PLS, les données ont été préparées de la manière suivante :

- Encodage des variables catégorielles : Les variables ont été encodées via la fonction pd.get_dummies en utilisant le paramètre *DropFirst = True (*diminuer le nombre de nouvelles features), afin de transformer ces variables en indicatrices.
- Normalisation des caractéristiques : La méthode PLS est sensible aux échelles des différentes variables. Par conséquent, un StandardScaler a été appliqué pour centrer et réduire les variables de sorte qu'elles aient une moyenne de 0 et un écart type de 1.

Une fois les données prétraitées, nous avons tracer la courbe de la variance expliquée cumulée en fonction du nombre de composantes principales conservées.

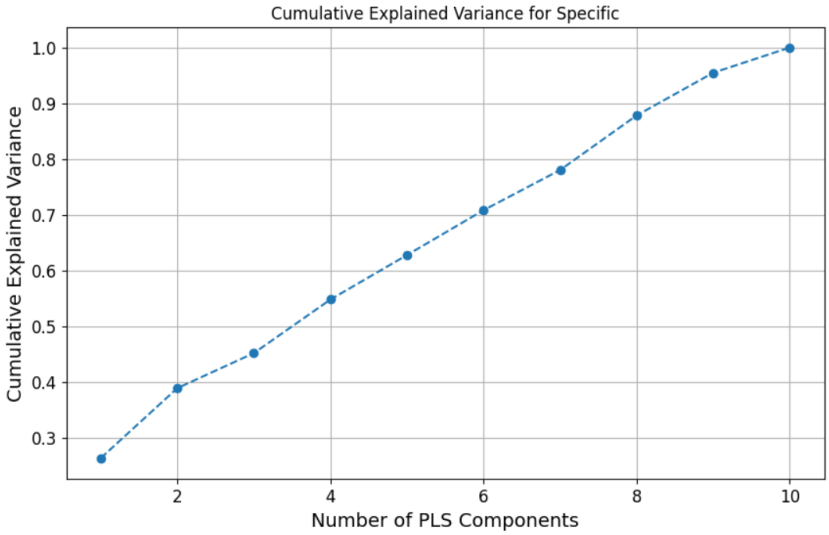

Figure 3 : Variance expliquée cumulée en fonction du nombre de composantes principales

On observe que la variance expliquée cumulée atteint le seuil 0.8 à partir de 8 composantes. Par conséquent nous avons fait le choix de ne garder que les 8 premières composantes principales calculées.

VI. Méthodes de ML

a. Methode 1 – Supervisé - Régression

Dans cette approche, nous utilisons plusieurs modèles de régression supervisée afin de prédire la qualité de la soudure, représentée par la résistance à la traction (*Yield strength*). Nous testons ces différents modèles avec les data frames obtenus via l'analyse PLS pour chaque type d'imputation. Cette approche nous permet d'évaluer plusieurs méthodes de régression afin de choisir celle qui convient le mieux à nos données. Les données de test et d'entrainement ont été séparé par une validation croisée « K-Fold » en 5 plis.

Nous avons choisi des modèles simples comme la **régression linéaire** et **KNN**, permettant d'évaluer s'il existe une relation linéaire entre les variables prédictives et la qualité de la soudure, ou encore pour capturer des **similarités entre groupes d'observations** dans des configurations de soudure spécifiques.
En complément, nous avons opté pour des modèles d'ensemble plus complexes tels que le ***Random Forest Regressor***, qui exploite plusieurs arbres de décision pour modéliser des relations non linéaires tout en étant robuste aux données bruitées.

Finalement nous avons explorés des améliorations du *Random Forest* avec des modèles de ***Boosting***, comme *XGBoost* et *AdaBoost*, permettent de corriger les erreurs à chaque itération en se concentrant sur les exemples difficiles à prédire, rendant ces algorithmes performants et adaptés aux tâches complexes.

b. Méthode 2 – Semi supervisé – Self Training

i. Classification

Le dataset initial présente un nombre significatif de données manquantes pour notre variable cible, la « Yield Strength ». Par conséquent, une approche efficace consiste à utiliser des techniques semi-supervisées pour prédire cette variable.

Une première étape dans notre démarche d'apprentissage semi-supervisé est de classer la variable à prédire. Pour cela, nous allons discrétiser les valeurs de « Yield Strength » en deux catégories en fonction de la moyenne de l'ensemble des données :

- 0 : si la valeur est inférieure à la moyenne.
- 1 : si la valeur est supérieure ou égale à la moyenne.

Cette discrétisation nous permet de traiter le problème comme un problème de classification binaire, avec l'objectif de classer chaque instance de notre jeu de données dans l'une des deux catégories.

Pour mettre en œuvre cette approche semi-supervisée, nous utilisons une régression logistique. Cet algorithme est adapté pour notre problème car il modélise la probabilité qu'une observation appartienne à l'une des deux classes (0 ou 1).

Avec une approche de self-training, on entraîne d'abord le modèle sur les données étiquetées. Ensuite, on utilise ses prédictions pour étiqueter les données non étiquetées, mais seulement si la confiance du modèle dépasse un certain seuil. Ces nouvelles étiquettes sont ensuite ajoutées au jeu d'entraînement, et le processus est répété, jusqu'à que toutes les données soient étiquetées et que le modèle final soit complètement entraîné.

ii. Régression

Contrairement à la méthode précédente, cette apprentissage semi-supervisé permet d'améliorer les performances d'un modèle de régression. Nous avons implémenté une boucle de self-training en utilisant un modèle de **régression Ridge** (avec un paramètre alpha fixé à 0.00001 déterminé avec la méthode GridSearch) car comparé aux autres méthodes, elle permet de réduire le surapprentissage.

Ainsi, nous sélectionnons les prédictions dont la confiance (mesurée par un seuil basé sur l'écart-type des étiquettes d'origine) est suffisamment élevée. Ces prédictions sont ensuite ajoutées au jeu de

données étiquetées pour les itérations suivantes, augmentant ainsi la quantité de données d'entraînement disponibles pour le modèle.

VII. Résultats numériques

a. Résultats de la méthode 1

Nous comparons ici les performances des différentes méthodes d'imputations en fonction de plusieurs algorithmes de prédiction. Plusieurs métriques d'évaluations sont utilisées : MSE, R^2 ajusté et R^2.

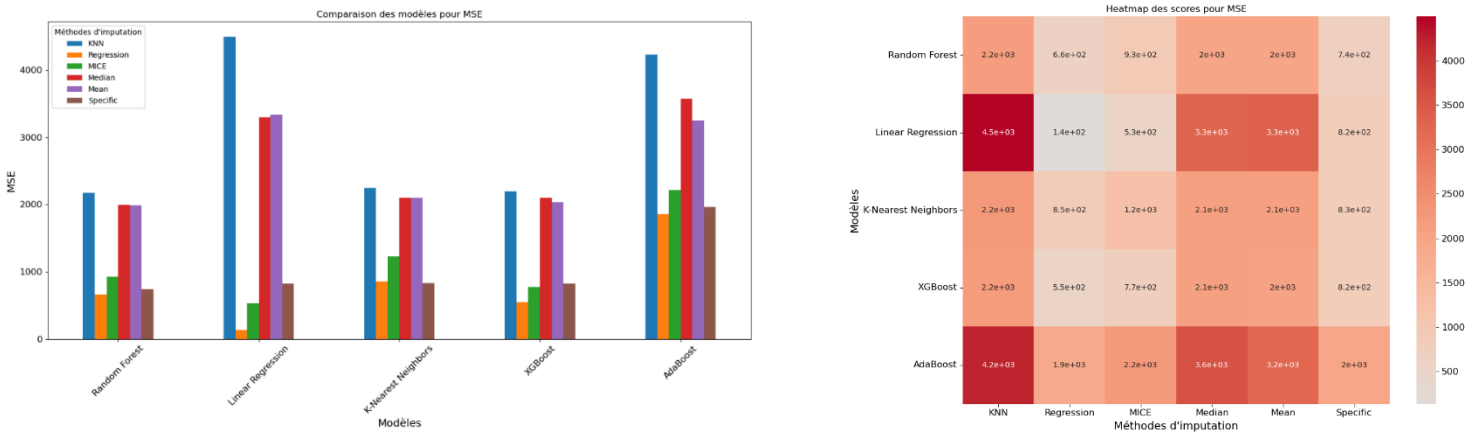

Figure 4 : Résultat MSE

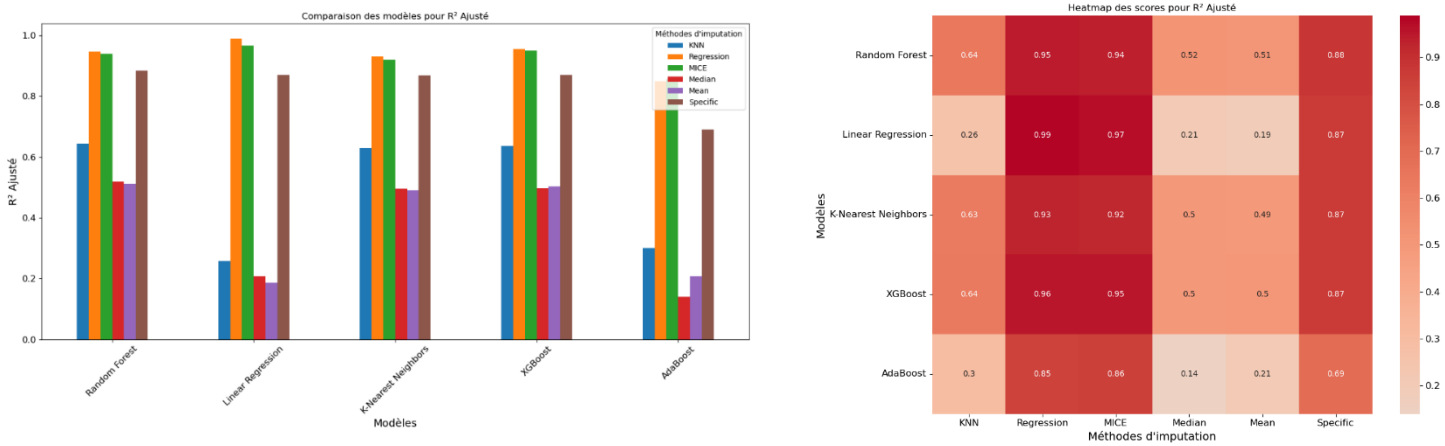

Figure 5 : Résultat R2 ajusté

L'analyse du MSE montre que les performances varient selon le modèle et la méthode d'imputation. En effet, les résultats montrent que Random Forest et XGBoost se démarquent en termes de précision, avec des MSE relativement bas, notamment lorsqu'ils sont associés aux méthodes d'imputation MICE et Specific. Ces combinaisons permettent de maintenir le MSE en dessous de 1000, montrant une bonne généralisation du modèle, malgré la complexité des données et les éventuelles valeurs manquantes. En revanche, Linear Regression et KNN obtiennent des MSE plus élevés, surtout avec des imputations comme Mean et Median.

L'évaluation à travers le R^2 ajusté renforce cette hiérarchie. Random Forest et XGBoost atteignent des scores proches de 1, surtout avec MICE et Specific, confirmant leur capacité à bien expliquer la variance des données. Linear Regression, bien que moins performante en termes de MSE, montre un R^2 ajusté élevé (proche de 0,99 dans certains cas), ce qui peut être trompeur en raison de la manière dont les données

manquantes sont complétées. Les résultats sont anormalement bons. KNN se positionne juste derrière, avec des R^2 plus modérés, et AdaBoost reste largement en retrait avec un R^2 ajusté faible, souvent inférieur à 0,4.

On remarque également, à travers ces métriques, que des méthodes d'imputation plus sophistiquées aident à préserver les relations entre les variables dans le dataset. En effet, des méthodes d'imputation comme KNN, MICE et « Specific » tiennent compte des corrélations entre les variables pour estimer les valeurs manquantes. Ainsi, les imputations sont plus réalistes, améliorant la cohérence des données et, par conséquent, la performance des modèles.

Finalement, XGBoost et Random Forest, avec des méthodes d'imputation avancées, apparaissent comme les modèles les plus performants pour prédire la valeur du Yield Strength.

b. Résultats de la méthode 2

Les résultats de notre premier modèle basé sur une approche semi-supervisé avec une régression logistique, se résument dans la matrice de confusion suivante :

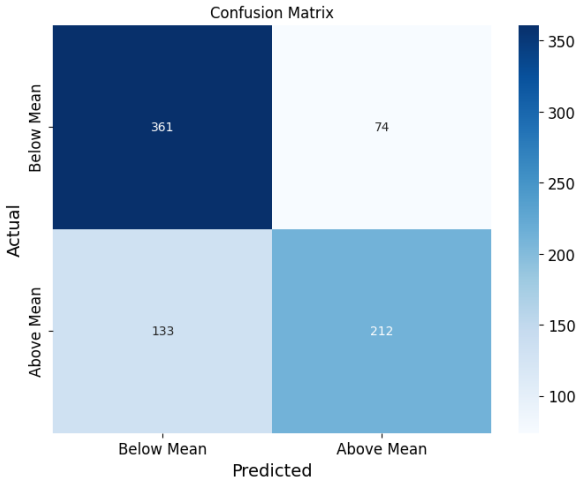

Figure 6 : Matrice de confusion - Régression logistique

À partir de ces données, nous pouvons calculer plusieurs métriques clés : la précision est de 74,0 %, ce qui signifie qu'une majorité des prédictions positives sont correctes, tandis que le rappel, à 61,5 %, montre que le modèle détecte environ 61,5 % des véritables instances positives. Ces résultats suggèrent que, bien que le modèle ait une bonne capacité à identifier les classes négatives, il pourrait encore être amélioré pour mieux détecter les classes positives.

De cette base, nous revoyons notre approche semi supervisé dans le but d'effectuer une tache de régression. Pour le nouveau entrainement, avant chaque itération, nous avons évalué les performances du modèle en utilisant une **validation croisée à 5 plis** pour mesurer le coefficient de détermination R^2, l'erreur absolue moyenne (MAE) et l'erreur quadratique moyenne (MSE).

En affichant les métriques au fur et à mesure des itérations, on observe que la MAE, MSE et R^2 s'améliorent indiquant que l'ajout progressif des données non étiquetées avec une prédiction fiable a permis de mieux généraliser les prédictions.

Après avoir finalisé le modèle, nous avons effectué une validation croisée finale et obtenu des scores moyens pour les métriques clés sur l'ensemble des données. Les résultats montrent une amélioration progressive et des performances supérieures aux méthodes d'apprentissage supervisé, confirmant l'efficacité de l'approche semi-supervisée.

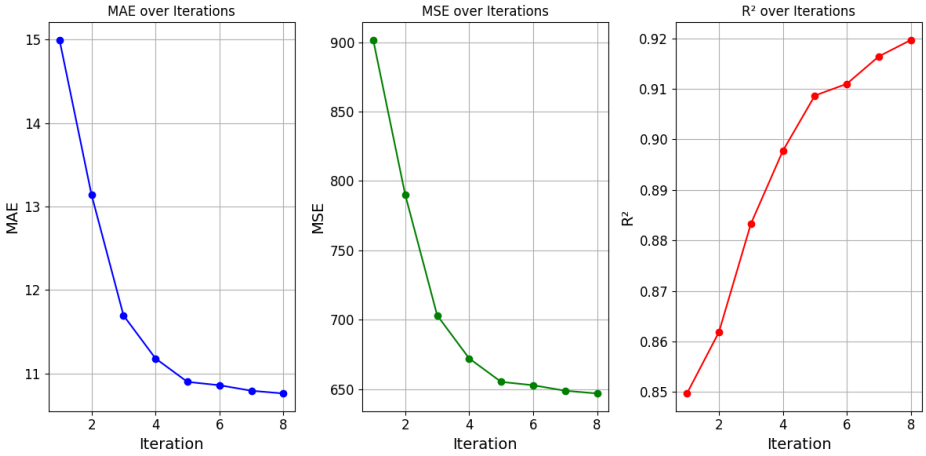

Figure 7 : Métriques de performance au fil des itérations de l'apprentissage semi-supervisé

A partir de ces résultats en appliquant la méthode du coude, on observe que le nombre d'itérations optimal est n_iter = 5.

c. Etude comparative des performances

Une façon de comparer des modèles de machine learning est d'explorer les aspects de robustesse, d'interprétabilité, de complexité ou encore de fiabilité des prédictions de chaque modèle. Le Random Forest Regressor est très robuste et gère bien les données bruitées, offrant des prédictions fiables même dans des environnements complexes. Cependant, il est moins interprétable en raison de sa structure en multiples arbres. XGBoost, quant à lui, se distingue par sa précision et sa capacité à améliorer ses prédictions, mais il est très complexe et exige des ressources computationnelles importantes, ce qui peut le rendre difficile à ajuster et à comprendre. Ici, le nombre de données n'est pas suffisamment élevé pour que la demande en ressources élevée ne soit pleinement justifiée.

En revanche, le modèle semi-supervisé avec régression Ridge combine données étiquetées et non étiquetées pour améliorer la robustesse et la fiabilité des prédictions. Cette approche permet de tirer parti de données supplémentaires sans étiquettes, ce qui est avantageux lorsque les données étiquetées sont limitées. Ce modèle maintient une bonne interprétabilité grâce à sa simplicité relative par rapport aux modèles comme XGBoost, bien qu'il puisse nécessiter des ajustements minutieux pour éviter le surapprentissage. Cependant, sa performance dépend fortement de la qualité des données non étiquetées, ce qui peut introduire des incertitudes.

VIII. Conclusion

a. Discussion des résultats obtenus

Ce projet a montré que l'apprentissage automatique peut prédire efficacement la qualité des soudures, offrant une alternative aux tests physiques. Les méthodes d'imputation choisies comme MICE et les modèles comme XGBoost et Random Forest ont produit des résultats prometteurs, bien que certains biais soient présents dans les méthodes plus simples comme la régression linéaire sur les données imputées par regression.

En effet, chaque modèle et méthode d'imputation présente des atouts et des limites. Les modèles complexes comme XGBoost et Random Forest sont performants et robustes, mais nécessitent des ajustements minutieux et leur explicabilité est parfois complexe. Les approches plus simples, comme la régression Ridge semi-supervisée, offrent un bon équilibre entre interprétabilité et efficacité avec des données étiquetées limitées.

Un dernier point porte sur l'évaluation de la qualité d'une soudure, prédire la limite élastique (*Yield strength*) est performant, mais dépendant du contexte où des caractéristiques mécaniques particulières qui

sont demandées pour la pièce soudée. Ainsi, prédire un score plus complexe peut être envisager dépendant de l'industrie considérée.

IX. Annexe

a. Figures complémentaires

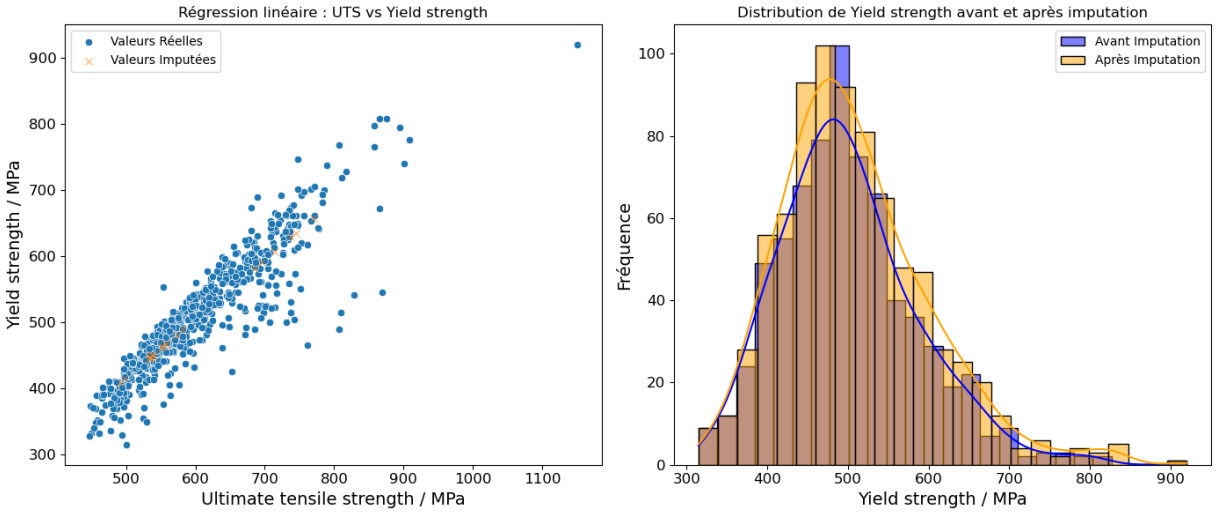

Figure 8 : Imputation précise Yield avec UTS

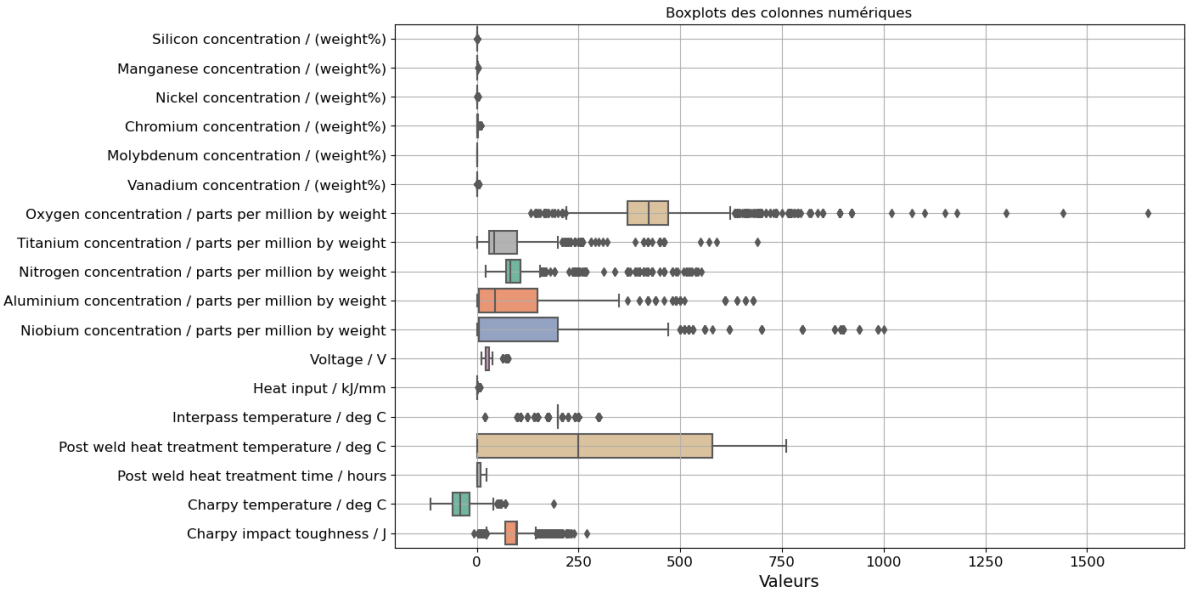

Figure 9 : Distribution des features

b. Diagramme de Gantt

Nous avons utilisé un diagramme de GANNT contenant le partage en tâches, le temps que chaque tâche a pris et quel membre de l'équipe s'en est chargée. Il nous a permis de suivre l'avancement du projet et l'organisation générale de celui-ci.

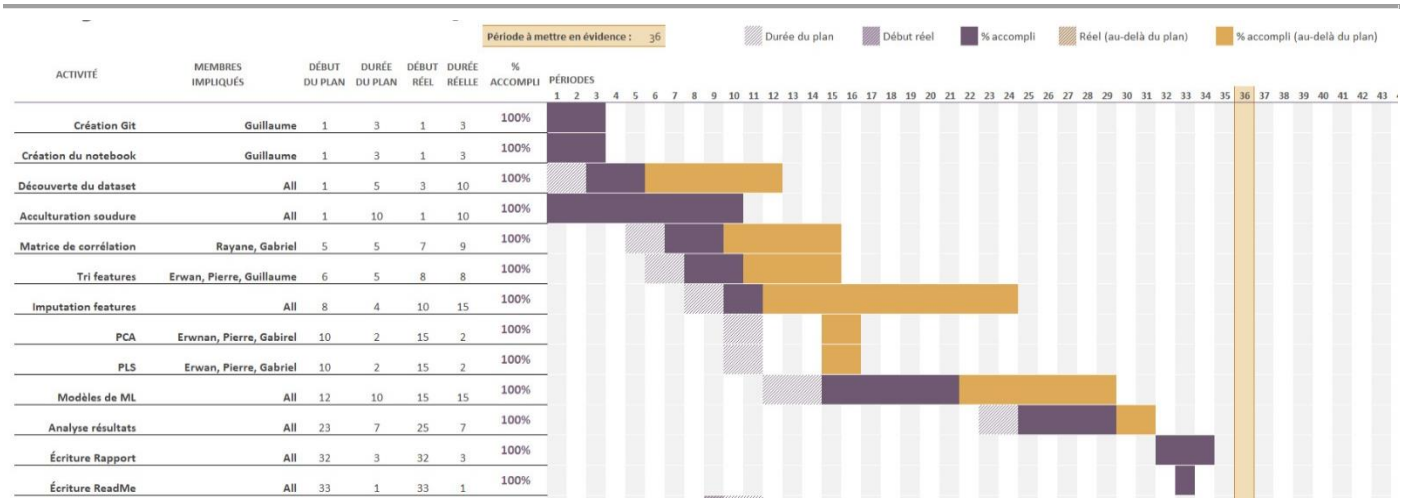

Figure 10 :Diagramme de Gantt

Table des figures