AIDS
MASTER'S DEGREE IN ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

كلية العلوم و التقنيات بطنجة
Faculté des Sciences et Techniques de Tanger

RAPPORT DE MINI PROJET

PYTHON

Sujet : Application consiste à analyser un ensemble de données.

14/01/2024

Réalisée par : BEZZAR Nouhaila

NOUIH Omar

Professeur : Mme. KHALI ISSA Sanaa

INTRODUCTION

Le présent projet a été initié dans le dessein de fournir aux étudiants une expérience pratique enrichissante dans le domaine de la programmation et de l'apprentissage automatique. À travers le développement d'une application versatile en langage Python, nous aspirons à offrir un environnement propice à l'exploration et à la mise en œuvre de concepts clés liés à l'analyse de données qualitatives et quantitatives, ainsi qu'à l'application d'algorithmes variés de machine learning.

Cette application polyvalente repose sur un socle robuste permettant d'aborder l'analyse de jeux de données sous différents angles, faisant appel à des algorithmes renommés tels que la régression linéaire, les arbres de décision, le naïf bayésien, le support vector machine (SVM), Random Forest, le k plus proche voisin, K-means, et les réseaux de neurones. Cette diversité d'approches permettra aux utilisateurs, en particulier les étudiants, de se familiariser avec un éventail de techniques d'apprentissage automatique, les préparant ainsi à relever les défis complexes du monde réel.

L'objectif principal de cette application est de démystifier les concepts souvent intimidants de la programmation et de l'apprentissage automatique en les rendant accessibles et applicables à travers une interface conviviale. Nous aspirons à créer un outil pédagogique qui favorise une compréhension approfondie des mécanismes sous-jacents à ces algorithmes, tout en offrant la flexibilité nécessaire pour explorer diverses applications et scénarios d'utilisation.

Au-delà de la simple implémentation d'algorithmes, notre application vise également à sensibiliser les utilisateurs aux bonnes pratiques de gestion de données, à l'importance de la préparation des données et à la prise de décisions éclairées tout au long du processus d'analyse. Ainsi, ce projet se veut être une ressource complète, offrant à la fois une immersion pratique dans le monde de la programmation et de l'apprentissage automatique, ainsi qu'une compréhension approfondie des principes fondamentaux de la gestion de données.

1. Architecture et Modules

L'application est construite en utilisant le langage de programmation Python avec l'interface utilisateur développée à l'aide du module Tkinter. Les modules et bibliothèques clés inclus dans l'application sont les suivants :

Tkinter : Utilisé pour créer une interface utilisateur intuitive permettant aux utilisateurs d'interagir avec l'application de manière conviviale.

Pandas : Employé pour la manipulation et la gestion des données. Facilite le chargement, le nettoyage et la transformation des ensembles de données.

Scikit-learn : Intégré pour la mise en œuvre des algorithmes de machine learning tels que la régression linéaire, les arbres de décision, le SVM, Random Forest, etc.

Matplotlib et Seaborn : Utilisés pour la visualisation graphique des données et des résultats.

PandasTable et sv-ttk : Intégrés pour créer des tables interactives dans l'interface utilisateur, permettant une exploration détaillée des données.

Autres bibliothèques : Des bibliothèques supplémentaires telles que NumPy, os, et platform sont utilisées pour des fonctionnalités spécifiques.

2. Fonctionnalités Principales

Interface Utilisateur Intuitive :

L'interface utilisateur offre une expérience conviviale avec des fonctionnalités pour créer, importer et extraire des ensembles de données de manière efficace. Les widgets Tkinter sont utilisés pour créer des fenêtres, des boutons et des champs de saisie.

Représentation des Données :

La visualisation graphique est réalisée grâce à l'utilisation de Matplotlib et Seaborn. Les graphiques interactifs permettent aux utilisateurs de comprendre visuellement les caractéristiques de leurs données.

Gestion des Données :

Pandas est utilisé pour la manipulation des données, y compris le nettoyage des données, la gestion des valeurs manquantes et la normalisation des caractéristiques. Les fonctionnalités sont intégrées pour assurer la qualité des données utilisées dans les modèles de machine learning.

Algorithmes de Machine Learning :

Scikit-learn est utilisé pour implémenter les algorithmes de machine learning. Les utilisateurs peuvent choisir parmi plusieurs algorithmes tels que la régression linéaire, les arbres de décision, le SVM, Random Forest, etc.

Validation des Modèles :

Des méthodes d'évaluation des modèles sont intégrées pour tester la performance des algorithmes sélectionnés. Cela inclut des métriques telles que l'exactitude, la précision, le rappel, et d'autres selon le type de problème.

Visualisation des Résultats :

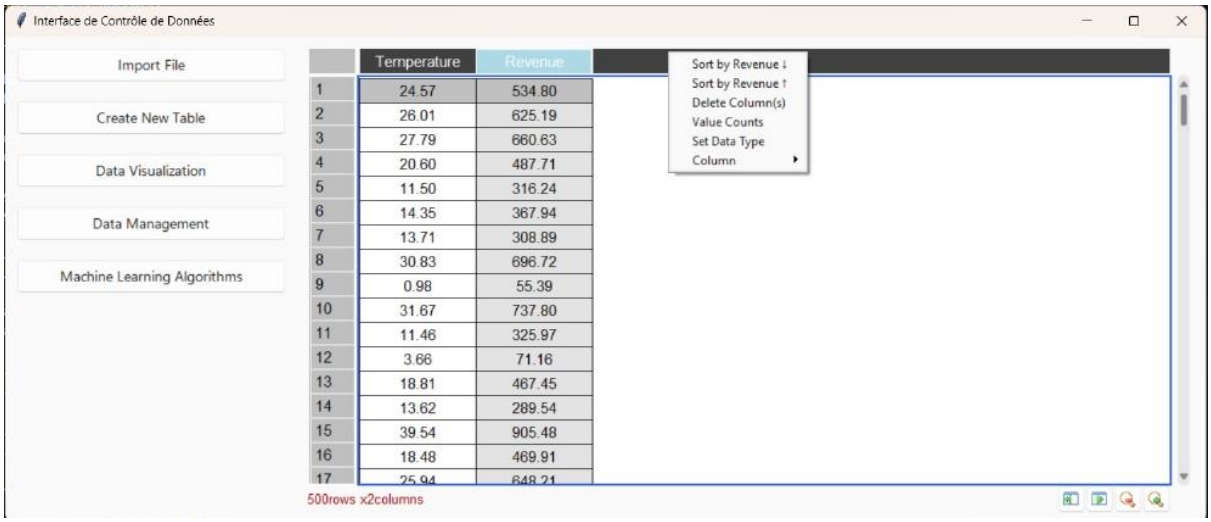
Matplotlib et Seaborn sont utilisés pour créer des visualisations claires des résultats, y compris des graphiques et des tableaux. Cela facilite l'interprétation des performances des algorithmes sur les données.

3. Explication de l'application

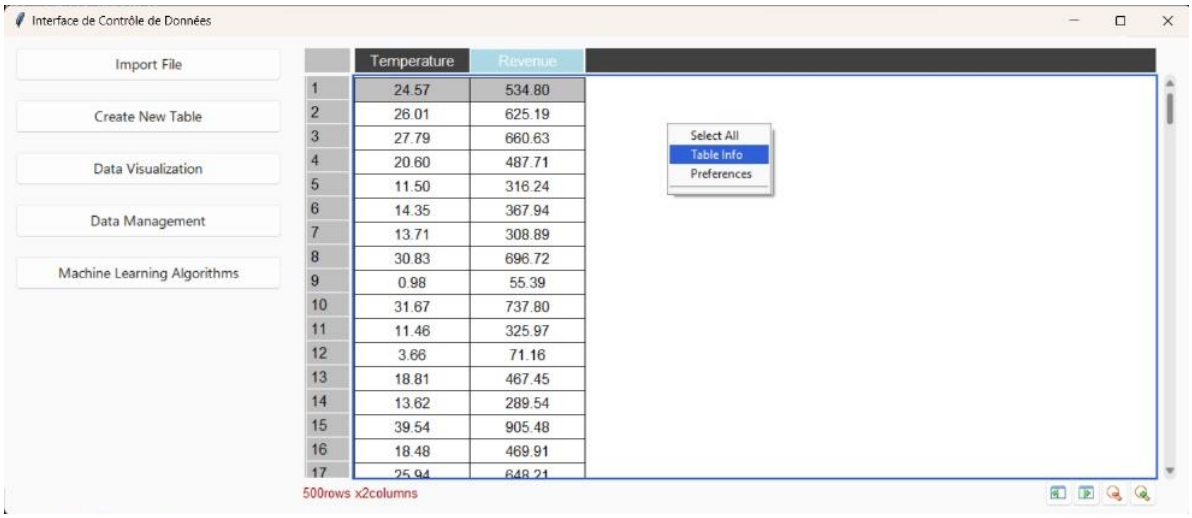
La figure ci-dessous représente l'interface de contrôle de données.

Importation de fichier :

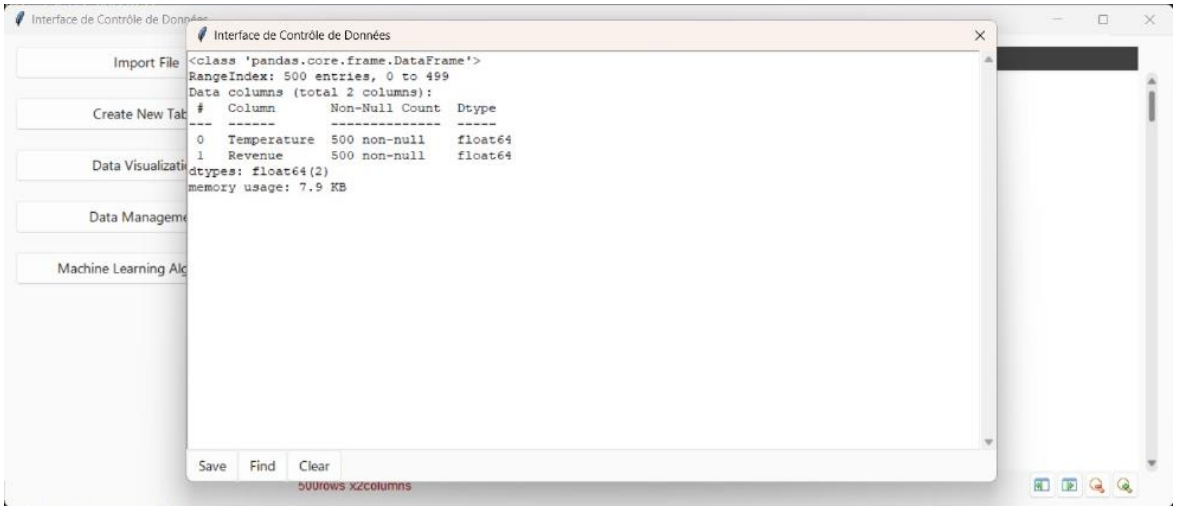
La figure ci-dessous représente la méthode pour gérer les colonnes.



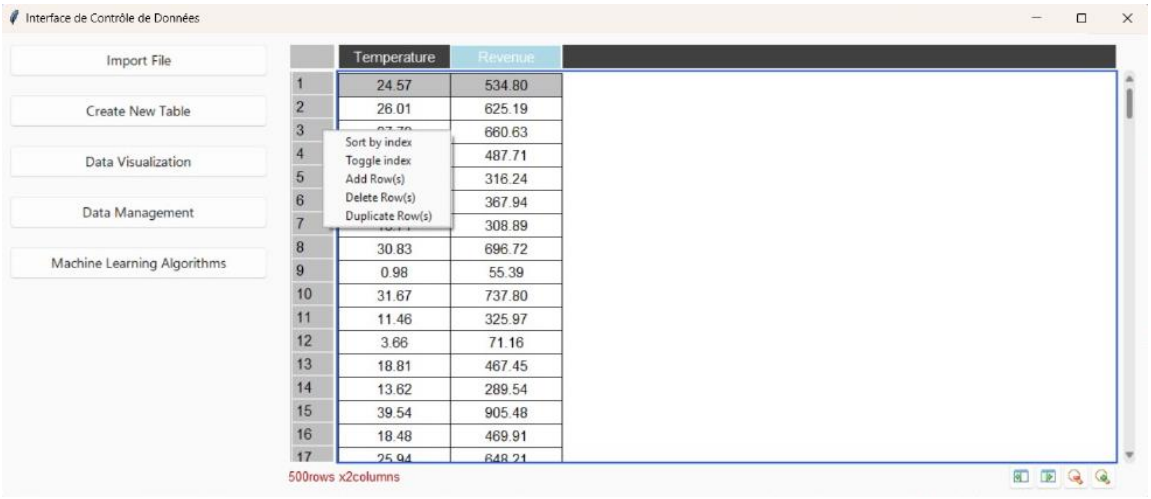
La figure ci-dessous représente la méthode pour afficher les informations du tableau.



La figure ci-dessous représente les informations du tableau.

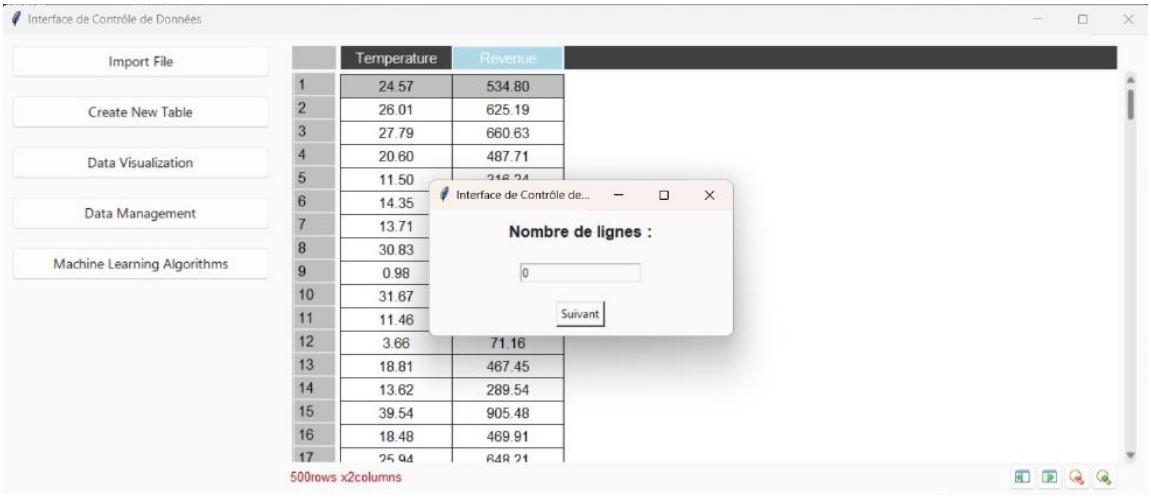


La figure ci-dessous représente la méthode pour gérer les lignes du tableau (la suppression, l'ajout..).

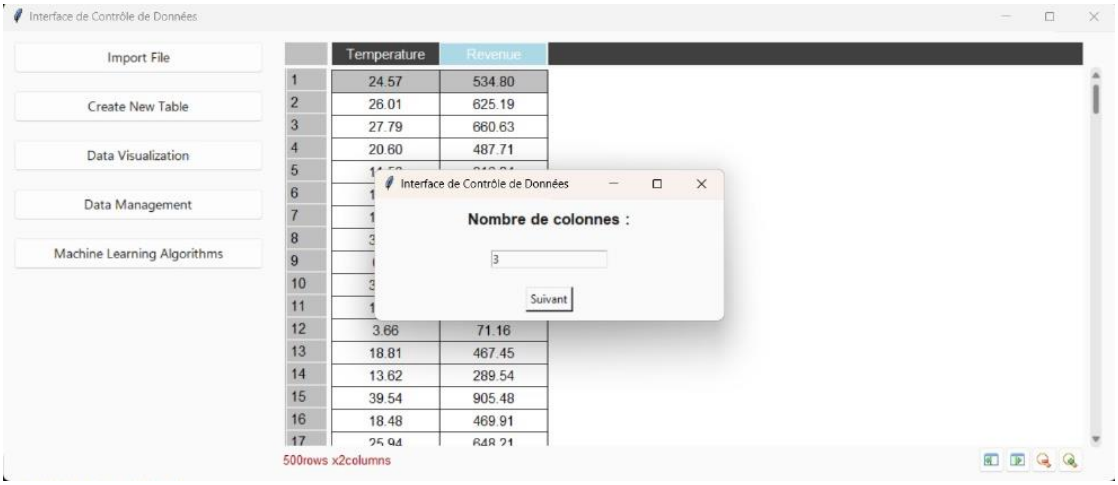


La création de nouveau tableau :

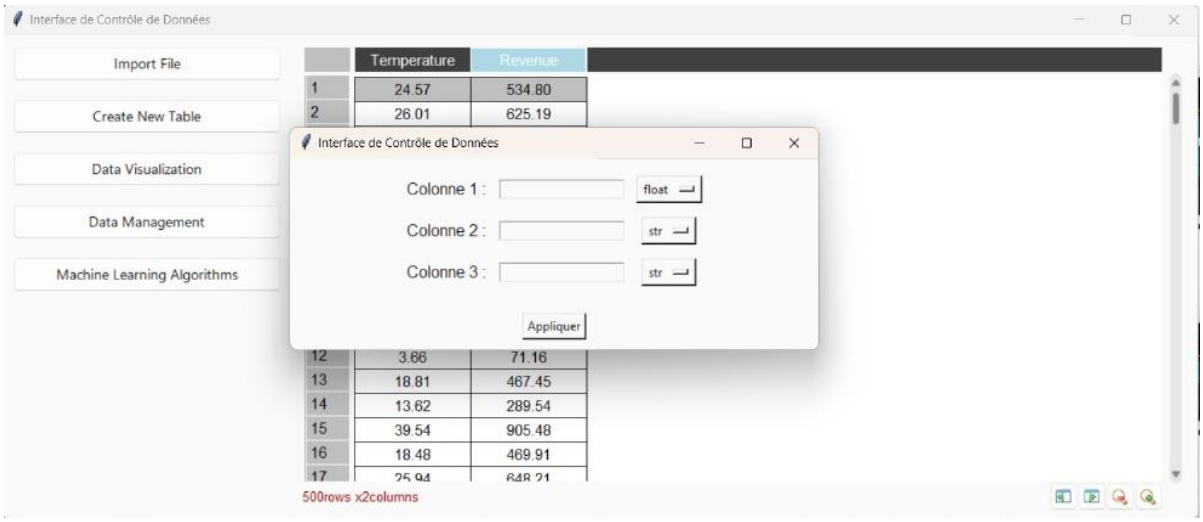
La figure ci-dessous représente la sélection de nombre de lignes du tableau.



La figure ci-dessous représente la sélection de nombre de colonnes du tableau.

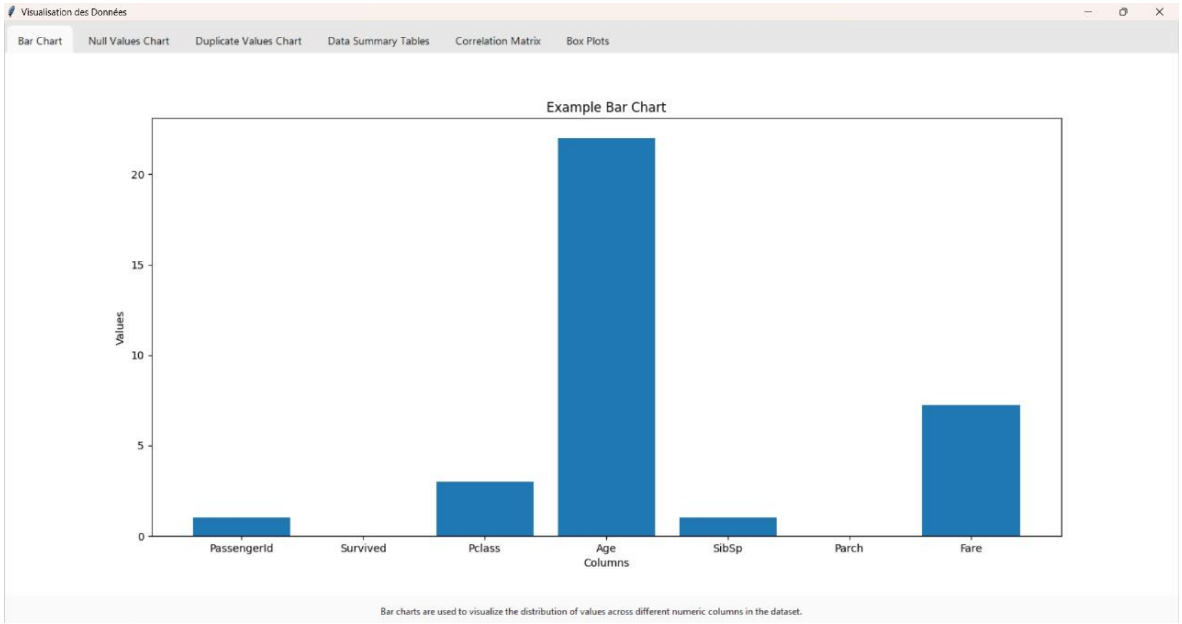


La figure ci-dessous représente le saisi des noms de colonnes et ses types.

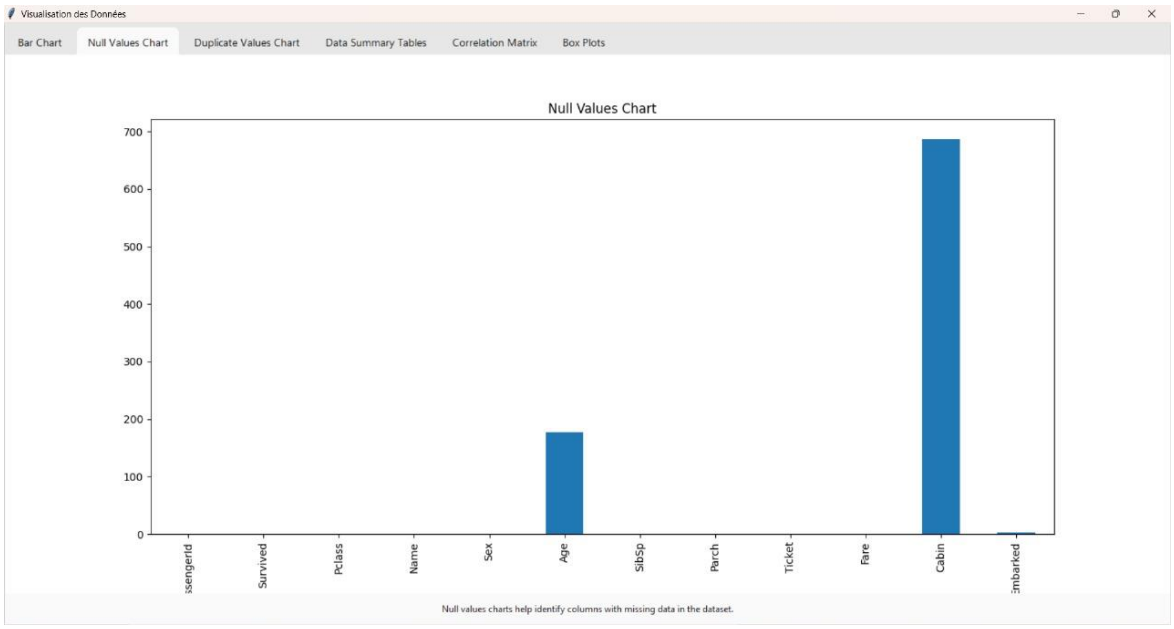


La visualisation des données :

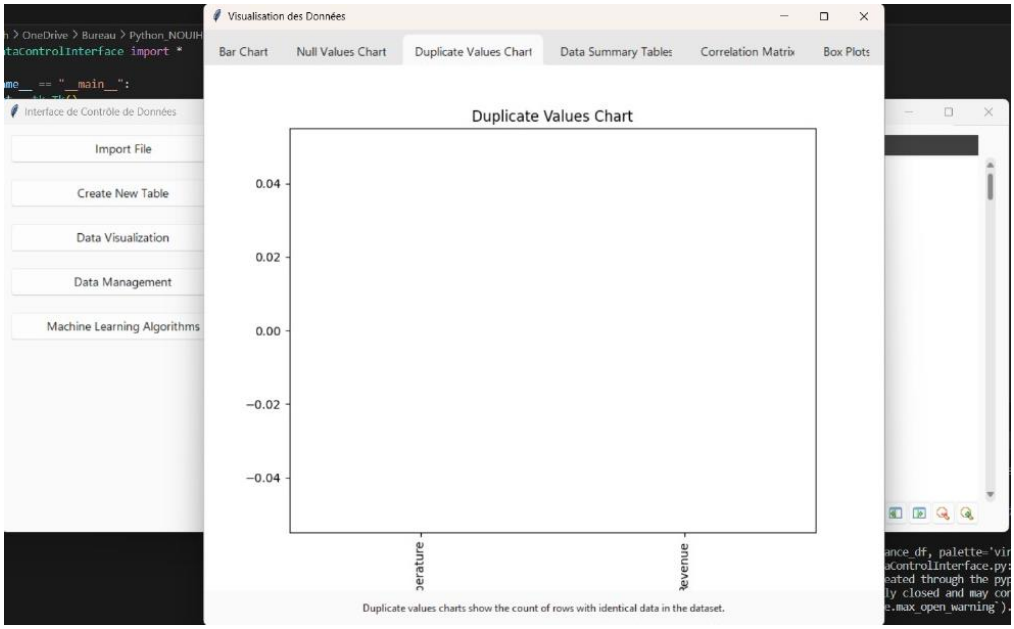
La figure ci-dessous représente les graphiques à barres, ils sont utilisés pour visualiser la distribution des valeurs sur différentes colonnes numériques de l'ensemble de données.



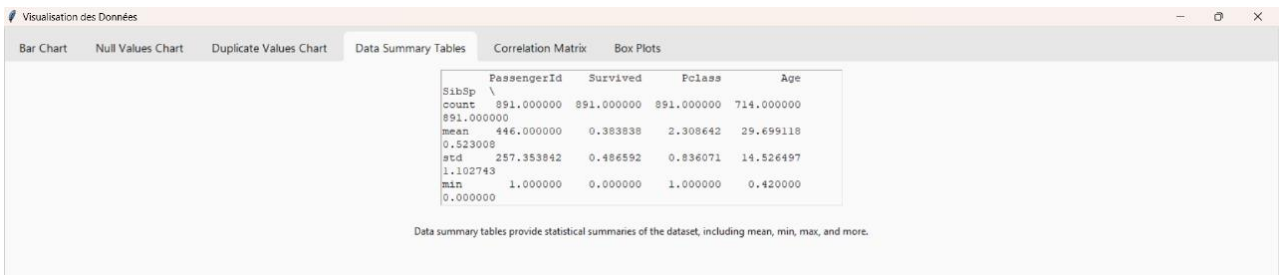
La figure ci-dessous représente les graphiques de valeurs nulles, ils aident à identifier les colonnes contenant des données manquantes dans l'ensemble de données.



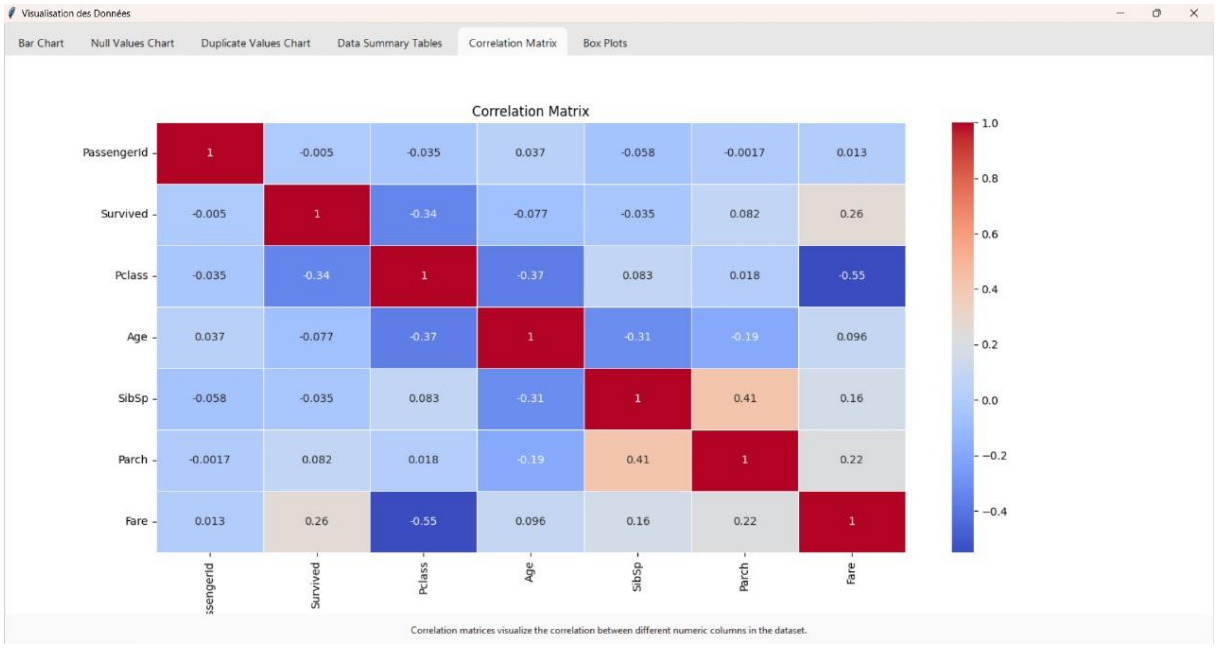
La figure ci-dessous représente les graphiques de valeurs en double, ils affichent le nombre de lignes contenant des données identiques dans l'ensemble de données.



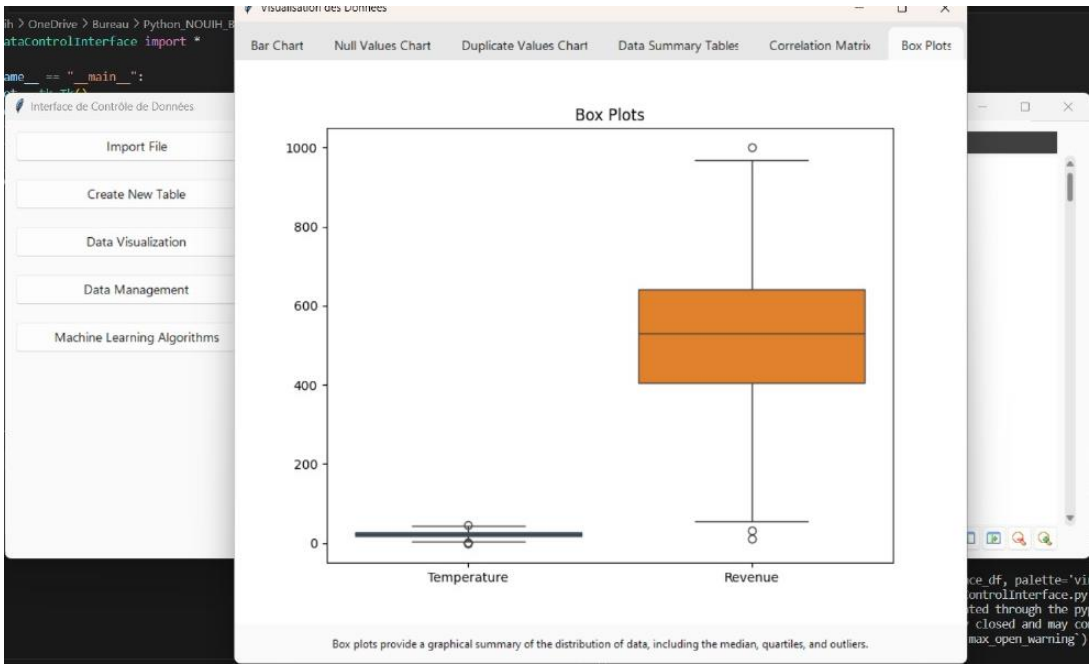
La figure ci-dessous représente les tableaux récapitulatifs des données, ils fournissent des résumés statistiques de l'ensemble de données, y compris la moyenne, le minimum, le maximum, etc.



La figure ci-dessous représente les matrices de corrélation, elles visualisent la corrélation entre les différentes colonnes numériques de l'ensemble de données.

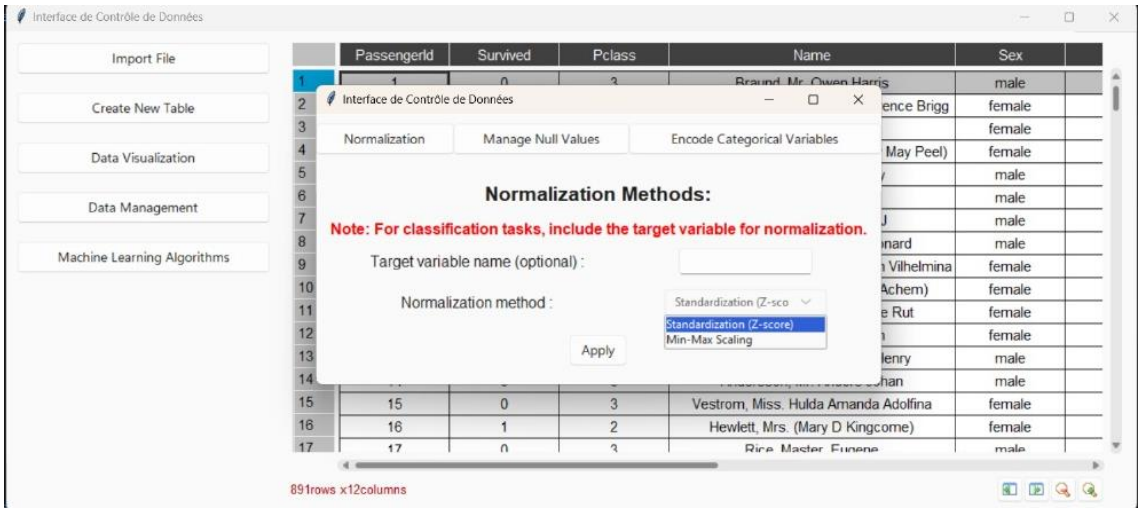


La figure ci-dessous représente les diagrammes en boîte, ils fournissent un résumé graphique de la distribution des données, y compris la médiane, les quartiles et les valeurs aberrantes.

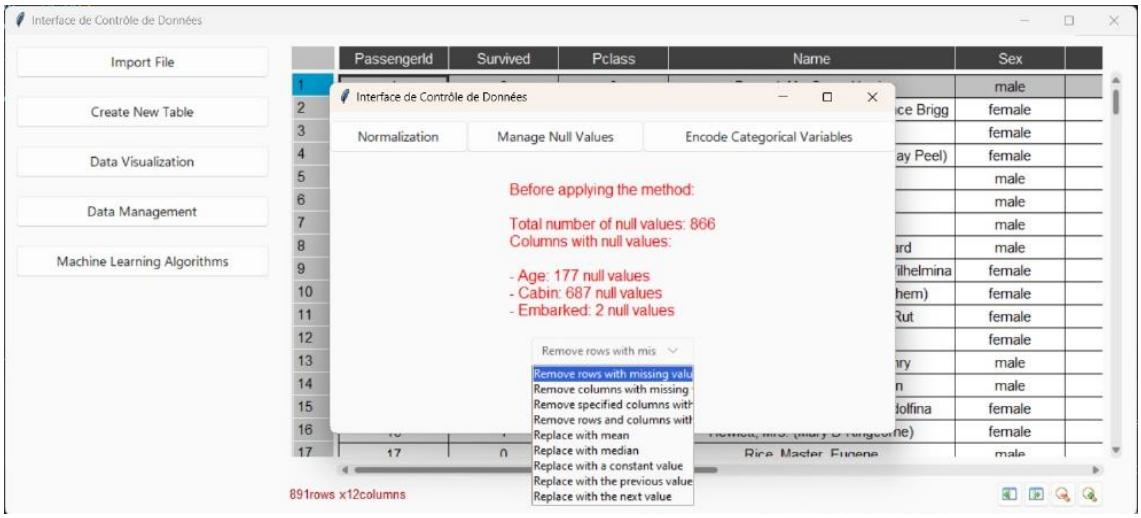


La gestion des données :

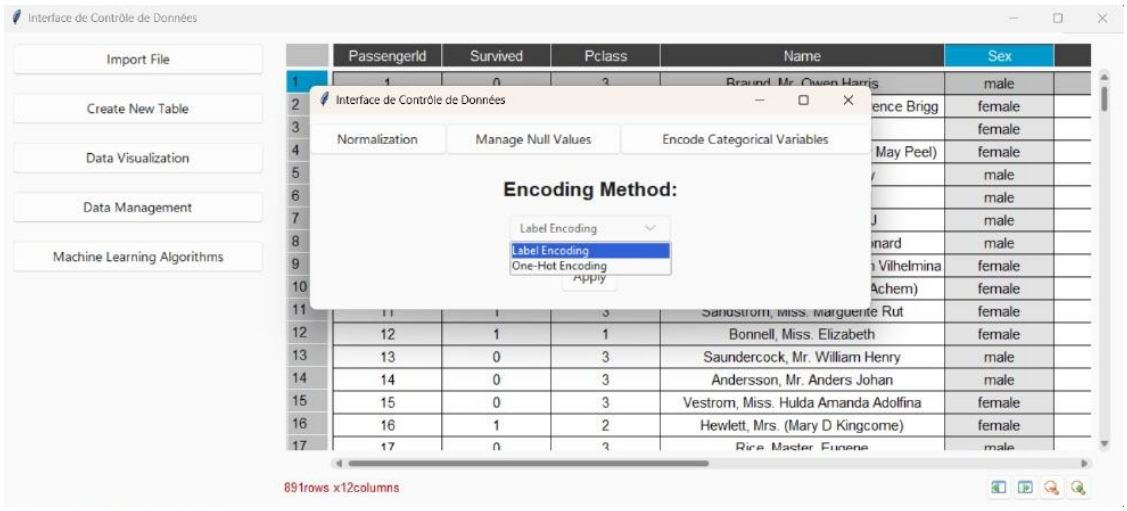
La figure ci-dessous représente la normalisation des caractéristiques.



La figure ci-dessous représente la gestion des valeurs nulles.

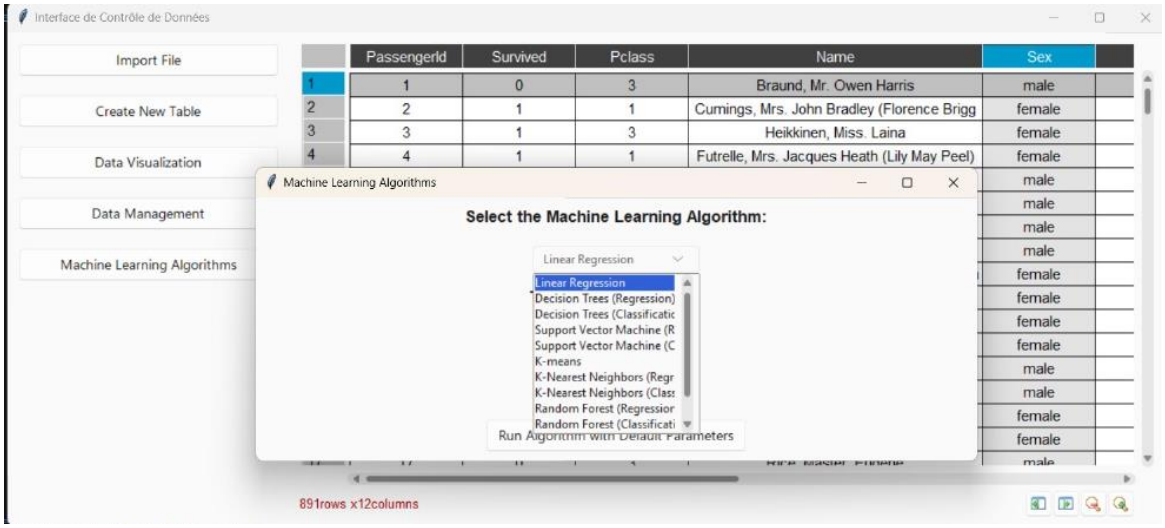


La figure ci-dessous représente la méthode pour encoder les variables catégoriels.



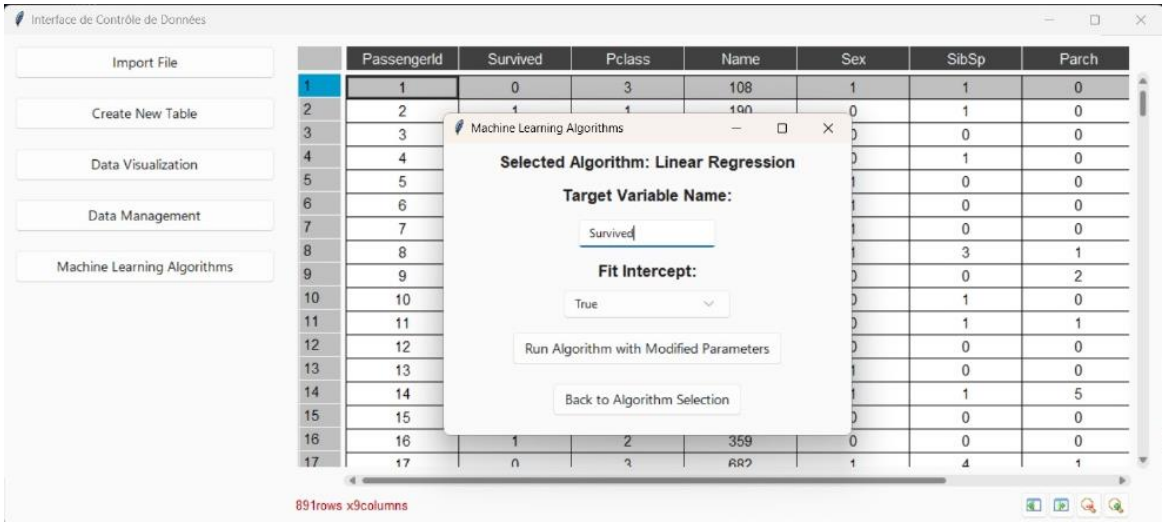
Les algorithmes de Machine Learning :

La figure ci-dessous représente les différentes méthodes de Machine Learning, par exemple l'arbre de décision, SVM, K-means …

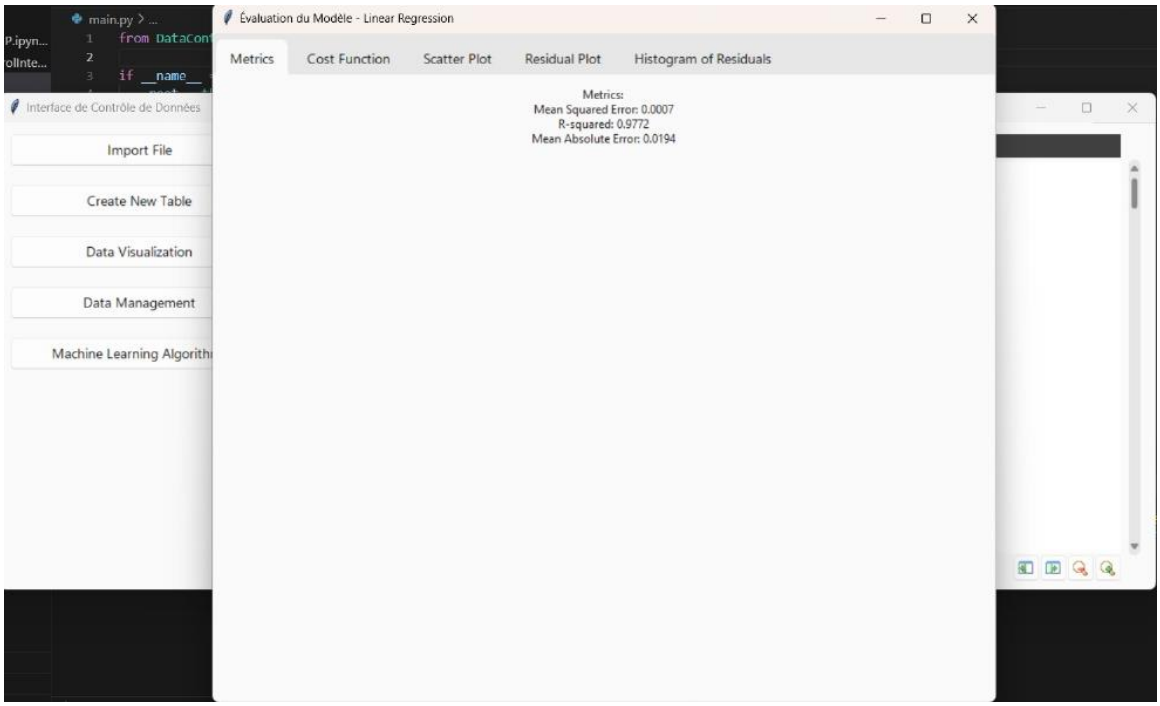


Pour la régression linéaire :

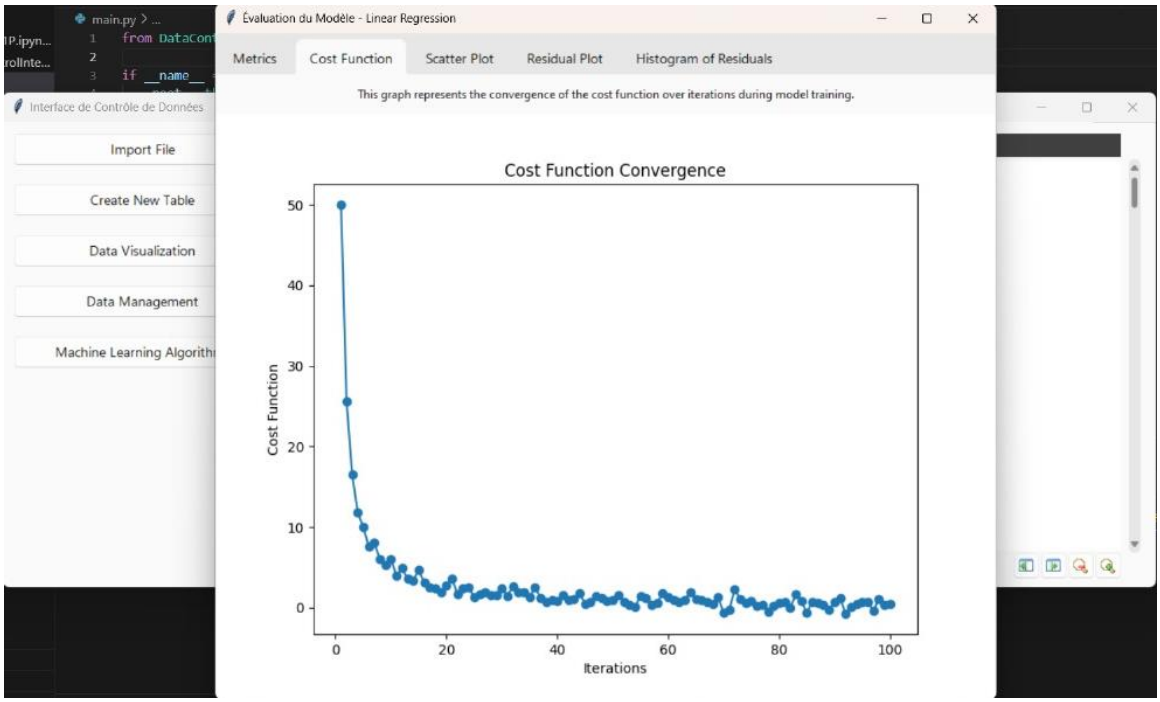
La figure ci-dessous représente la saisie des informations pour créer l'algorithme de la régression linéaire.



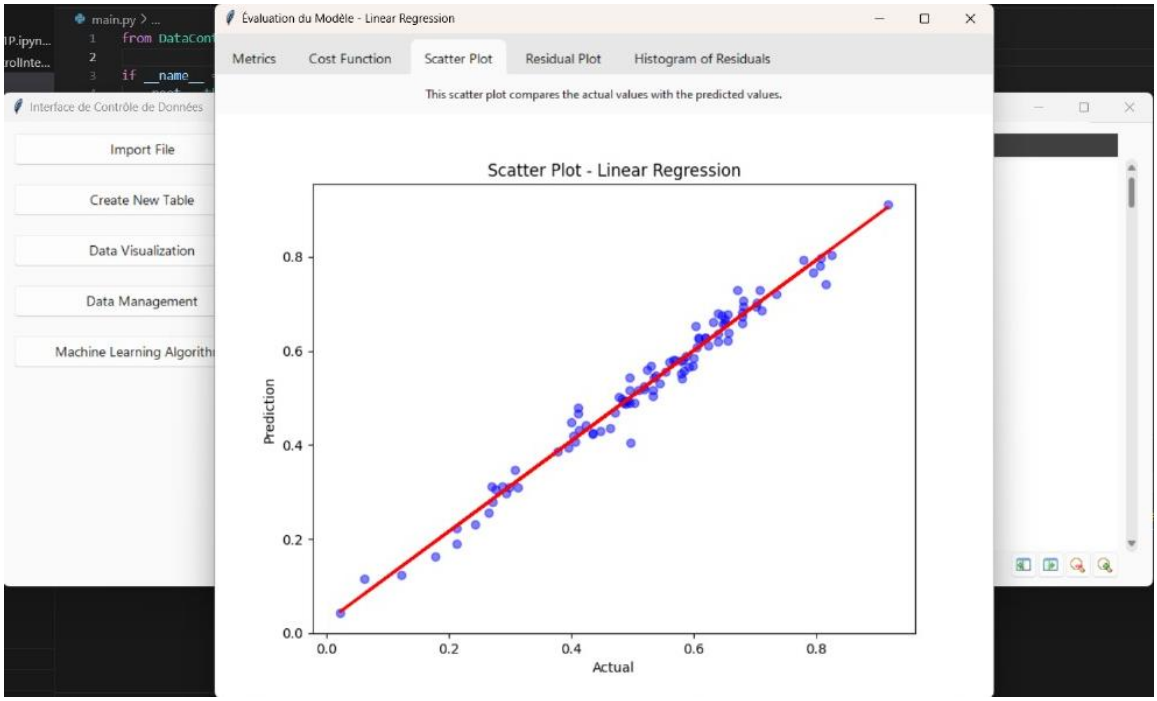
La figure ci-dessous représente la matrice de la régression linéaire.



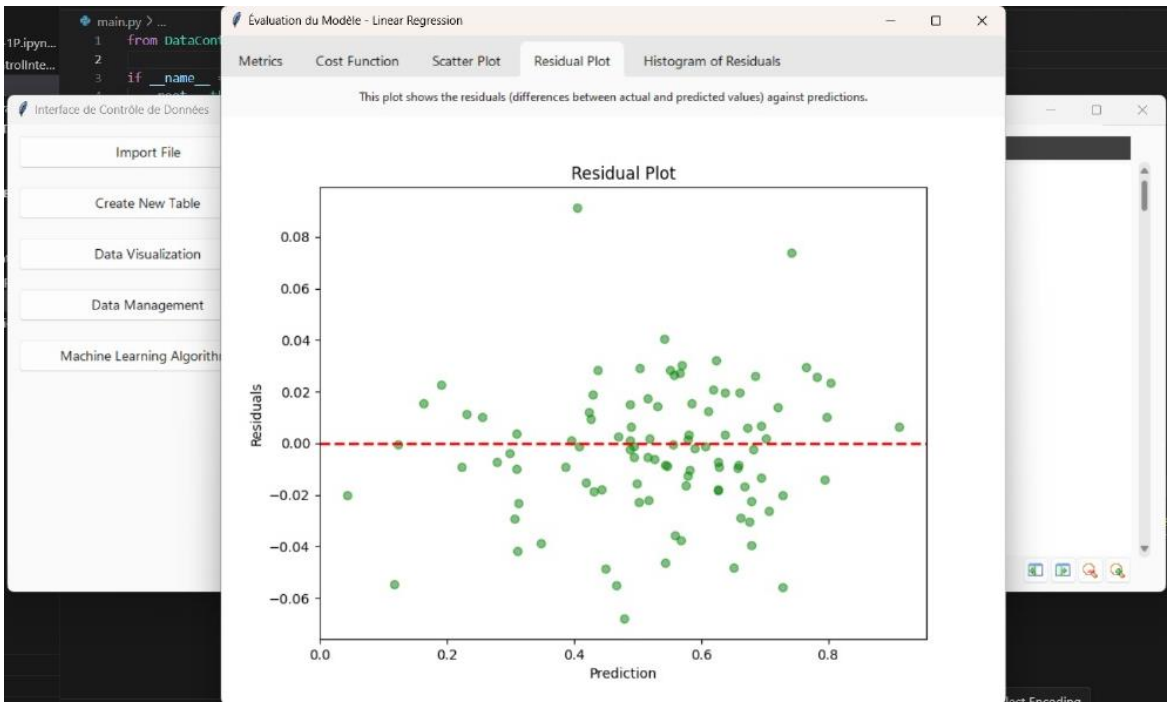
Ce graphique ci-dessous représente la convergence de la fonction de coût au fil des itérations lors de la formation du modèle.



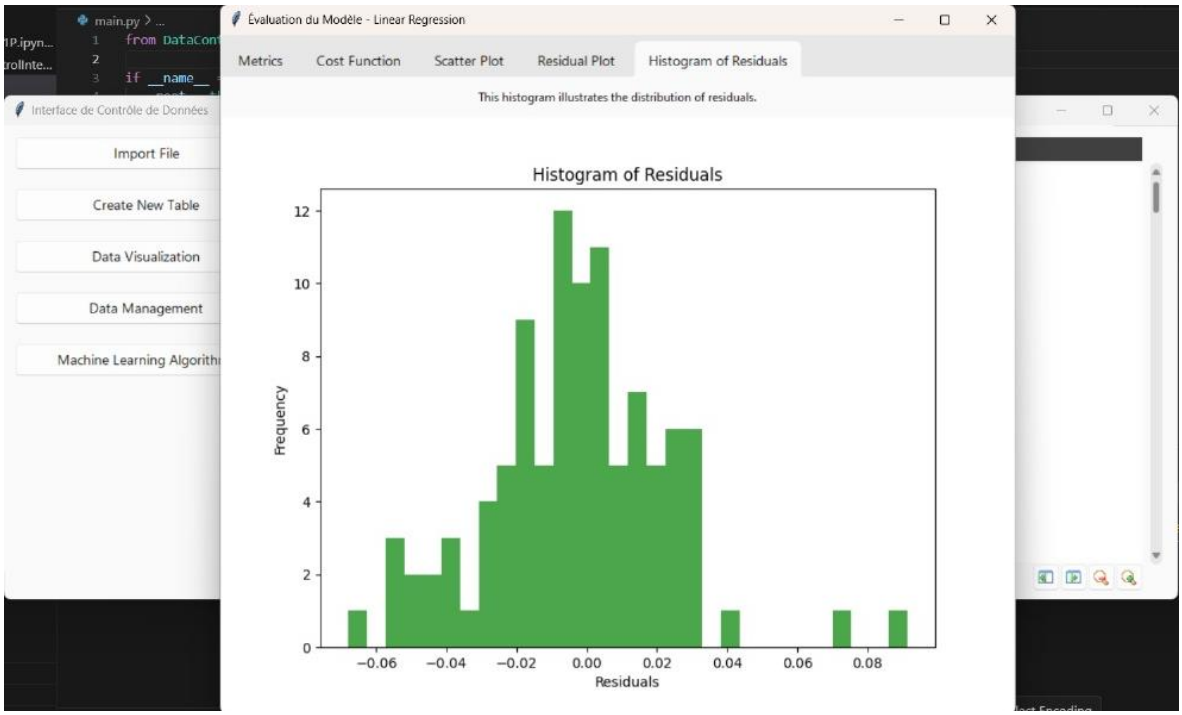
La figure ci-dessous représente le nuage de points, il compare les valeurs réelles avec les valeurs prédites.



Ce graphique ci-dessous montre les résidus (différences entre les valeurs réelles et prédites) par rapport aux prédictions.

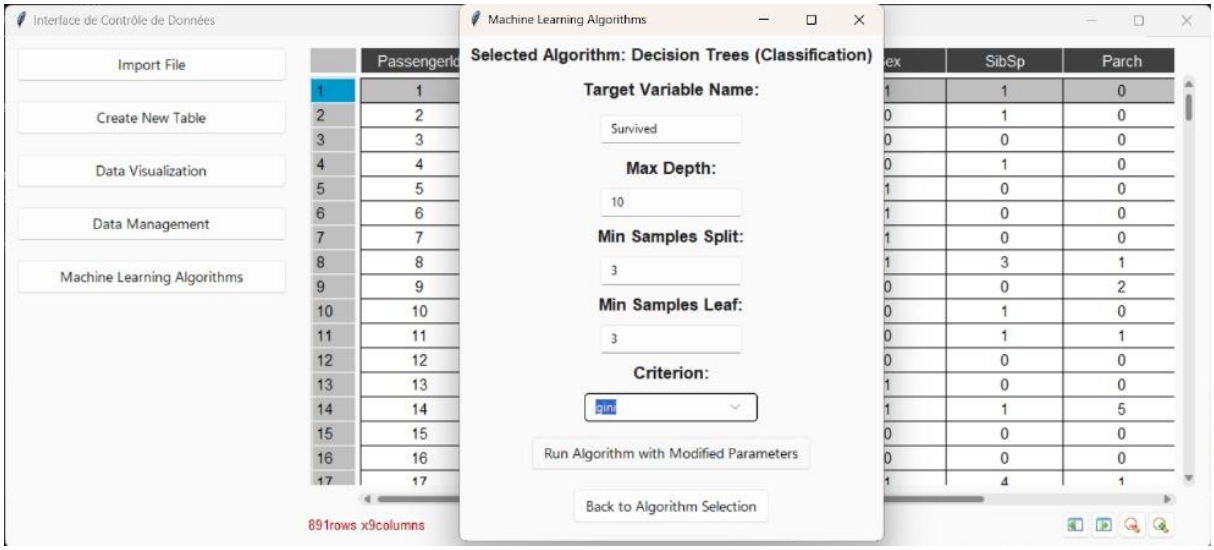


Cet histogramme ci-dessous illustre la répartition des résidus.

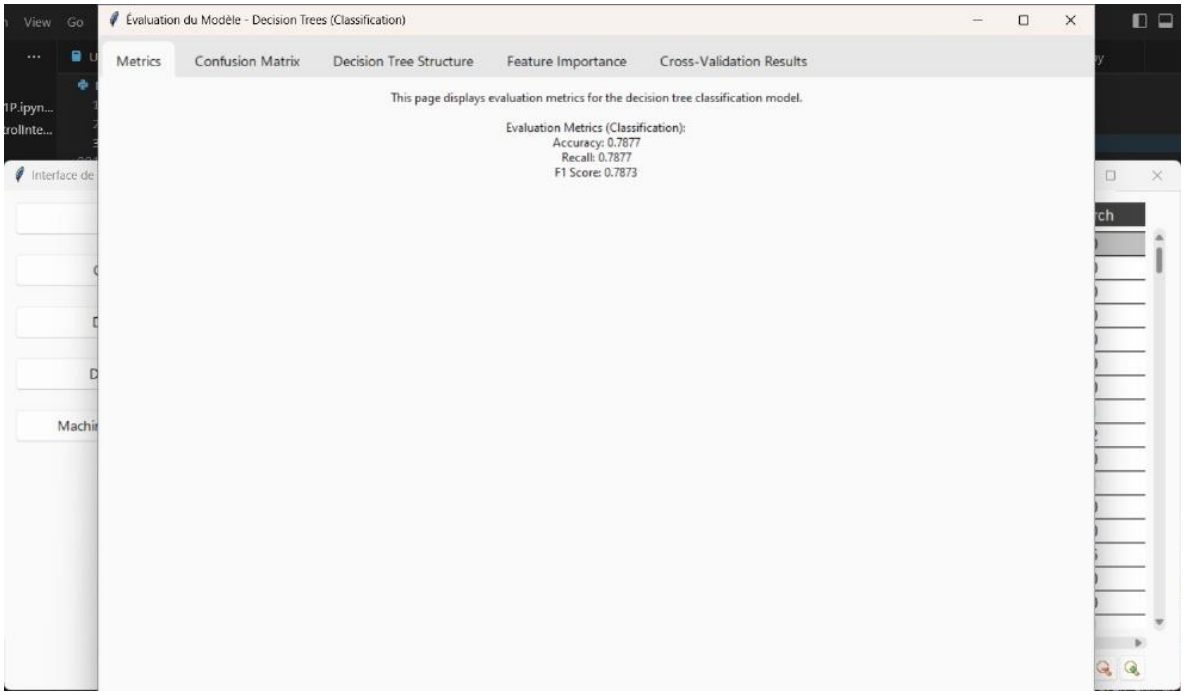


Pour l'arbre de décision pour la classification :

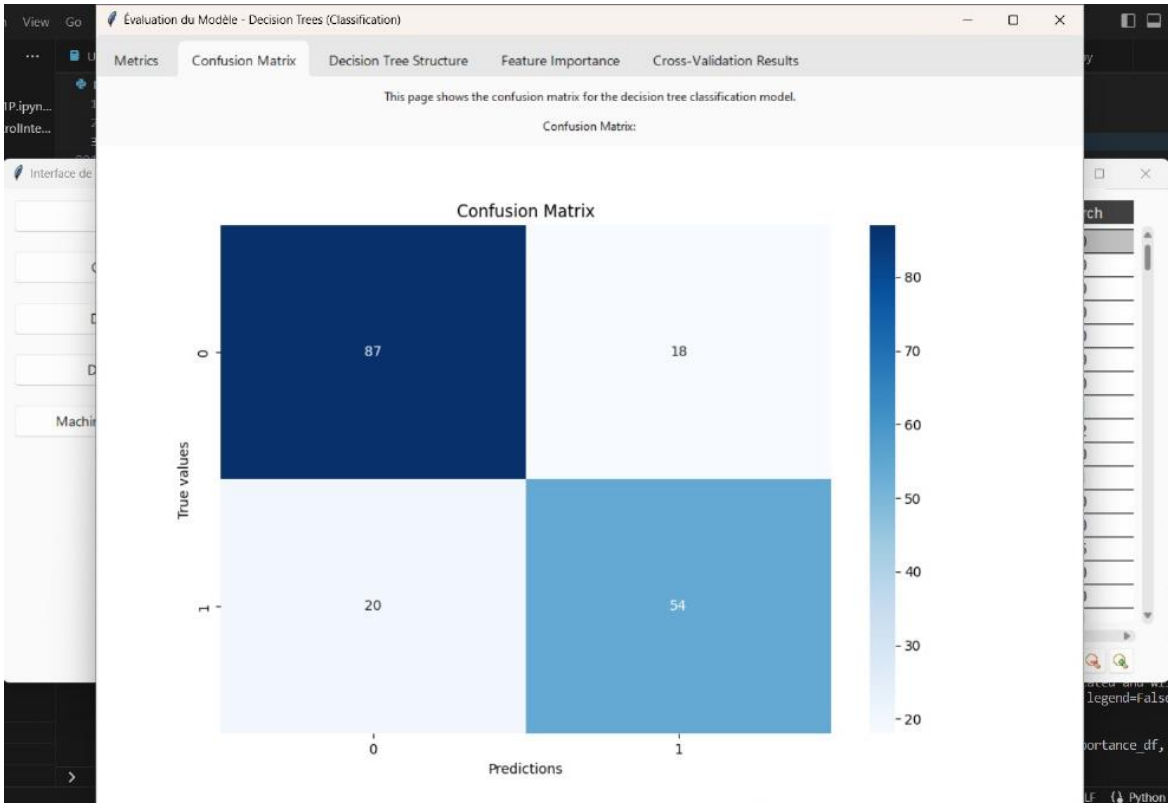
La figure ci-dessous représente la saisie des informations pour créer l'algorithme de l'arbre de décision pour la classification.



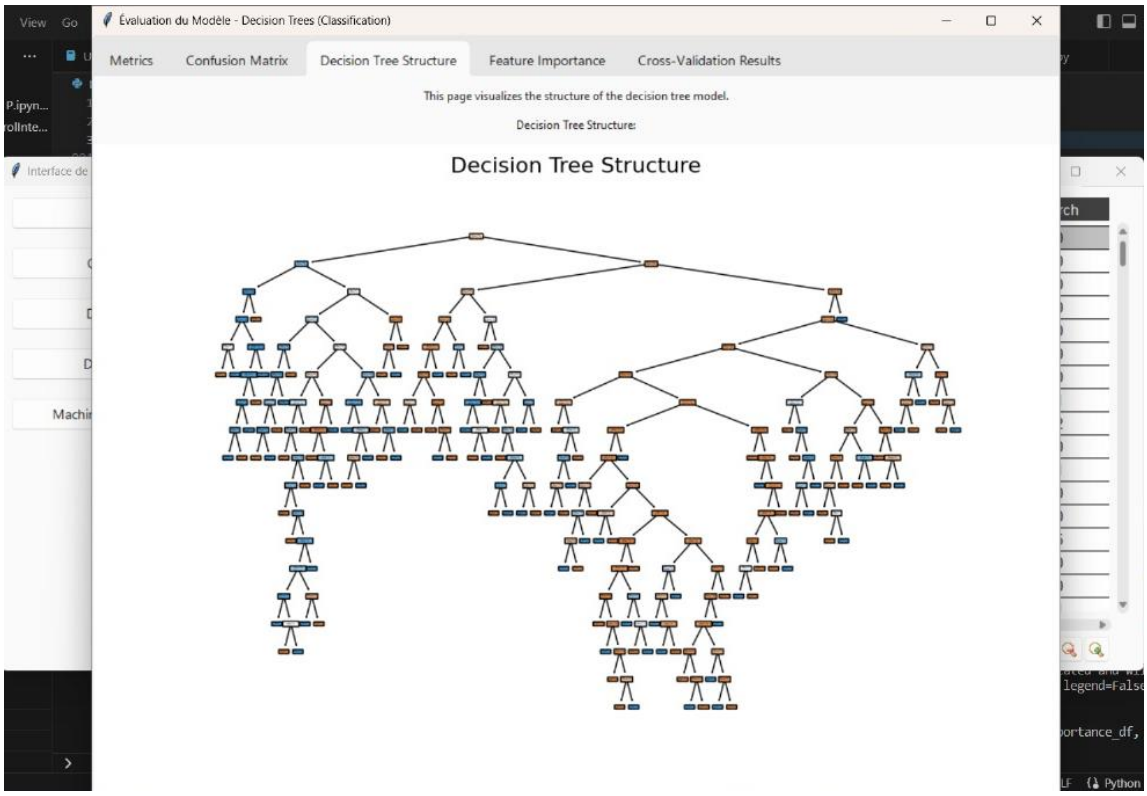
Cette page ci-dessous affiche les mesures d'évaluation du modèle de classification d'arbre de décision.



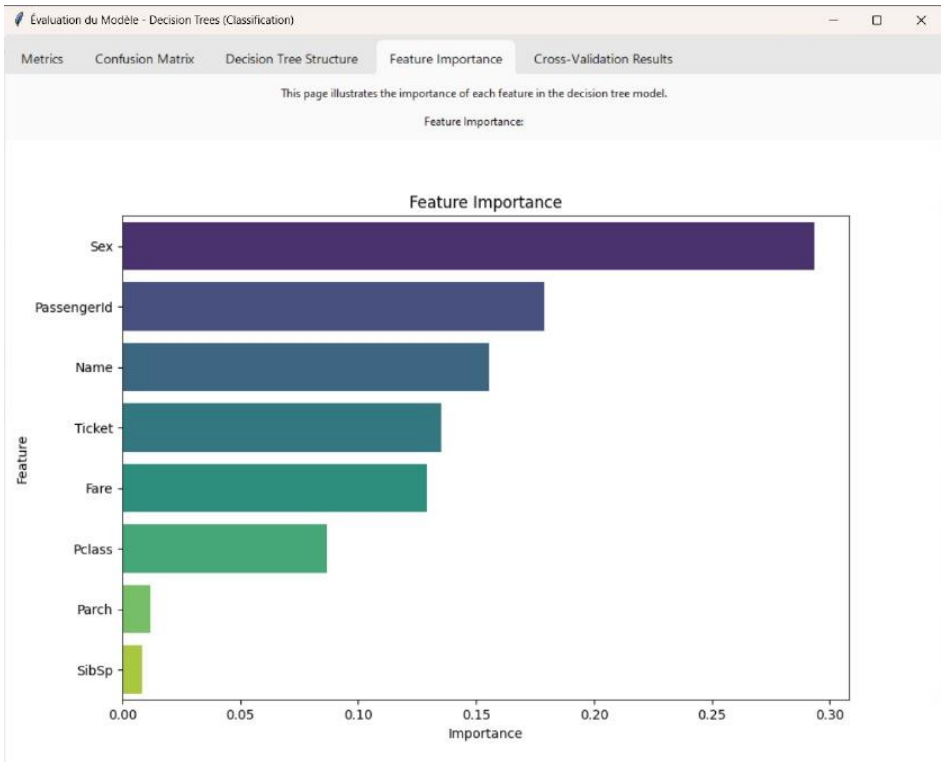
Cette page ci-dessous montre la matrice de confusion pour le modèle de classification d'arbre de décision.



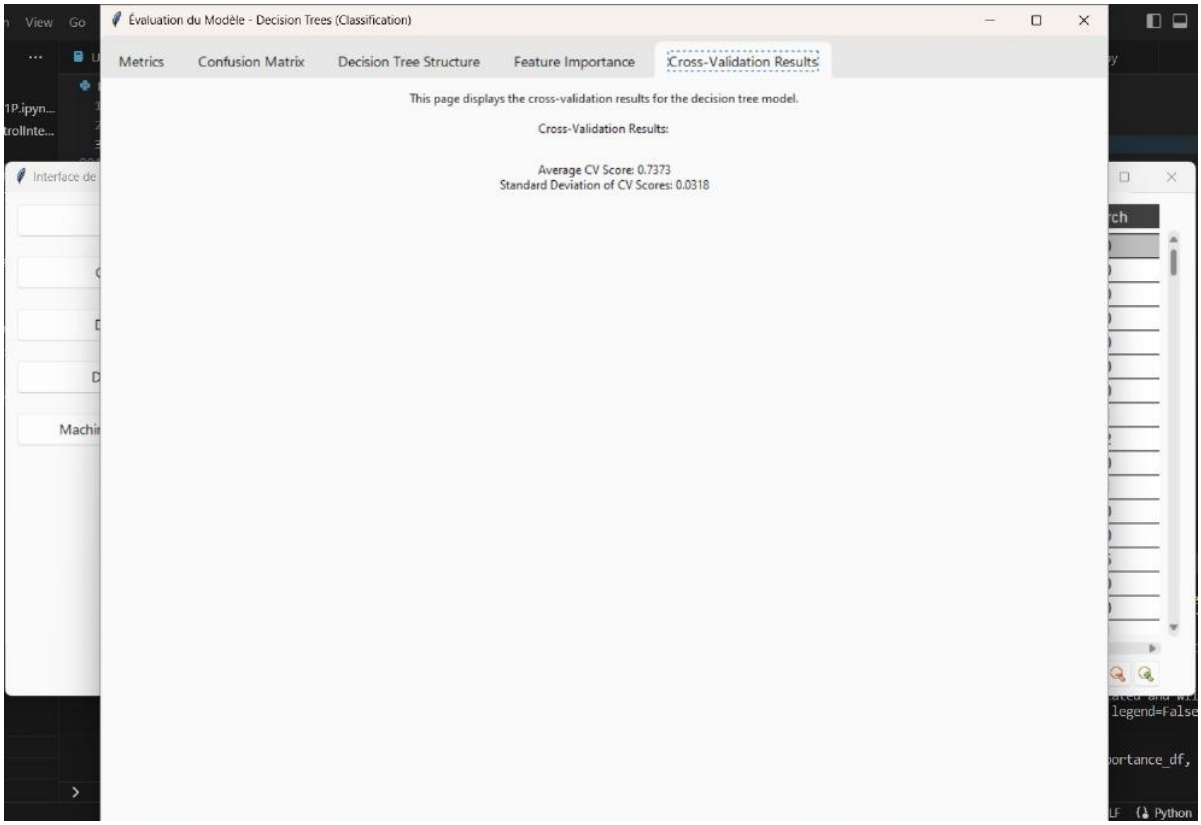
Cette page ci-dessous visualise la structure du modèle d'arbre de décision.



Cette page illustre l'importance de chaque fonctionnalité dans le modèle d'arbre de décision.



Cette page affiche les résultats de la validation croisée pour le modèle d'arbre de décision.



CONCLUSION

L'application offre une solution complète pour l'analyse de données et l'apprentissage automatique, avec une interface utilisateur conviviale, des fonctionnalités de gestion des données, une variété d'algorithmes de machine learning, une validation de modèles robuste et des outils de visualisation des résultats. La documentation complète et les tutoriels garantissent que l'application peut être utilisée de manière efficace par les étudiants et autres utilisateurs. Ce projet a permis aux étudiants de développer des compétences pratiques en programmation et en machine learning tout en travaillant sur des cas d'utilisation réels.

WIBIOGRAPHIE

- ✓ **scikit-learn**: https://scikit-learn.org/stable/
- ✓ **tkinter** : https://docs.python.org/fr/3/library/tkinter.html
- ✓ **pandastable** : https://pandastable.readthedocs.io/en/latest/description.html