2022

BAGUIAN HAROUNA DIASSANA FATOUMATA dite SOUKOURA

17/06/2022



Rapport projet Métier : Application de machine learning

Table des matières

**Aucune entrée de table des matières n'a été trouvée.**

Introduction

Le machine Learning est en plein essor, de nos jours les algorithmes sont devenu de plus en plus efficace et donne de bon résultats. les algorithme de machine Learning il y’a de toute sorte allant de la prédiction a la classification. Cet essor est favorisé par le boum de la masse de donnée. Ces données doivent être traité afin de les utilisé dans l’algorithme de machine Learning c’est le data prepocessing. Le traitement des données fait référence à la conversion de données brutes en informations significatives, et ces données sont également lisibles par machine.  Ainsi, le traitement des données implique la suppression des valeurs aberrantes, les outliers et les normalisé les données. Après le traitement des données une visualisation est nécessaire afin de mieux comprendre nos données, c’est le data visualisation. La data visualisation (data viz ou représentation graphique de données ; elle s'écrit également data visualization) consiste à structurer visuellement des données recueillies et stockées. Ainsi, l'exploitation des données se fait plus facilement. Ensuite nous pouvons choisir le modèle adéquat pour entrainer avec nos données. Faire tout ceci est réservé a un public aguerri.

Mais si on pouvait faire du machine Learning sans coder ?

C’est au vu de répondre a cette problématique que nous avons créez data cleaner.

Mots clés : python, data prepocessing, data visualisation, streamlit, prediction, classification

Le machine Learning passe par plusieurs étapes : data collection and prepocessing, choix du modèle, training du modèle, évaluation et la prédiction

1. **Accueil**



1. **Preprocessing**

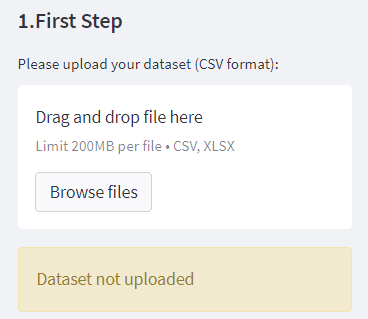
Le preprocessing est la deuxième étape du machine Learning juste après la collecte des données.

Le prétraitement fait référence aux transformations appliquées à nos données avant de les alimenter à l’algorithme. Le prétraitement des données est une technique utilisée pour convertir les données brutes en un ensemble de données propres.

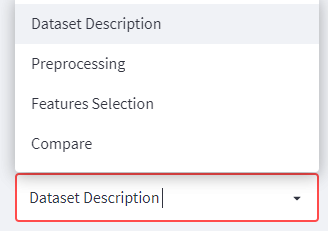
Pour le prétraitement de nos données on est passé par plusieurs étapes.

1. **Uploader les données**

Dans cette partie les données brutes avec lesquels on désire entrainer les algorithmes sont uploader sous format csv.



1. **Dataset description**



Comprendre le dataset est très primordial pour faire du machine Learning.

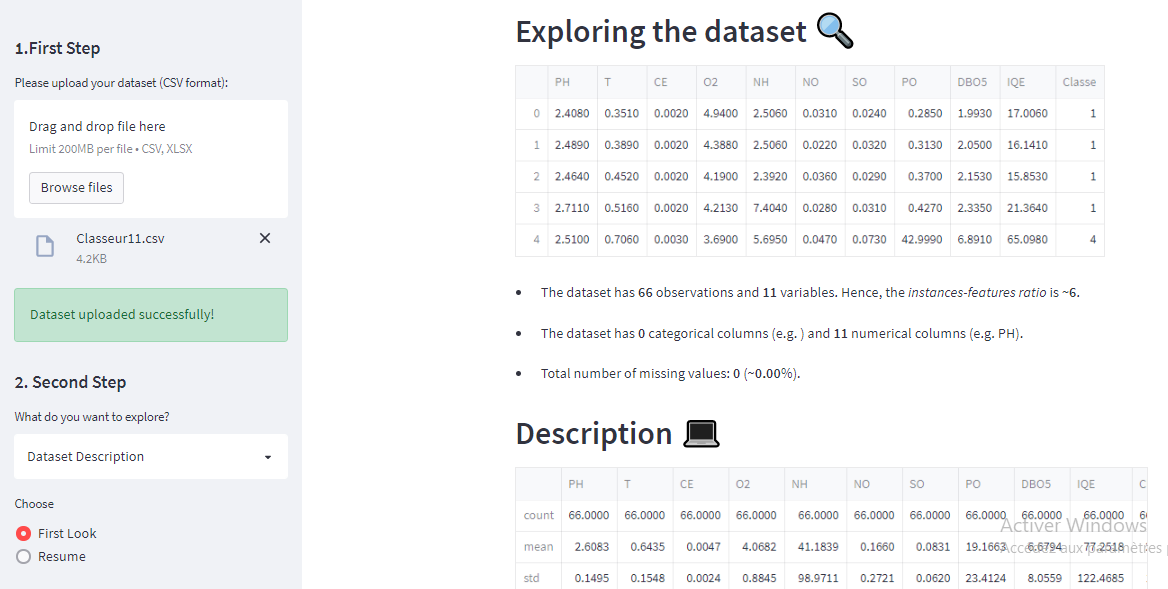
Plusieurs informations très utiles peuvent émaner d’un dataset. Connaitre ses informations est très crucial. Parmi ces informations on peut citer le nombre de ligne et de colonne du dataset, le nombre de lignes, de colonnes du dataset, les différents types des variables (features), le nombre de valeurs manquantes (qui peuvent provoqué des erreurs d’entrainement pour des modèles de machine Learning qui sont sensible au valeurs aberrantes).

Après avoir uploadé le bon dataset, sélectionné dans le select box juste en bas du message « dataset uploaded successfully » l’option dataset description.

**First look**

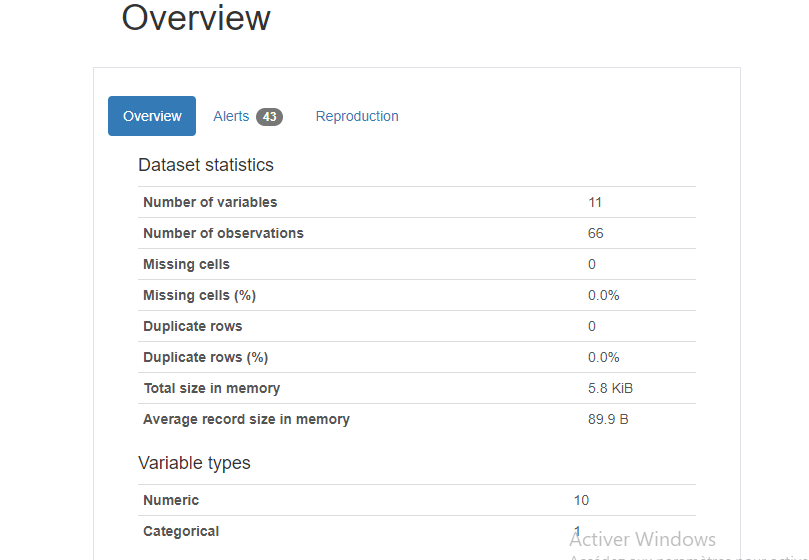
Le radio bouton first look nous donne un aperçu très global du dataset que vous venez t’uploader.

Ces informations sont le nombre et lignes et de colonnes, le nombre de variables catégorielle et numérique, le nombre de valeurs manquantes et leurs pourcentages, on peut également avoir les statistiques de base comme la moyenne, l’écart type, la valeur minimale et maximale de chaque variable.



**Resume**

Comme le radio bouton first look, résume donne en plus d’une description basique il donne une description très poussé et intégrale du dataset, on peut observer les différentes corrélations entres les variables. Naviguez jusqu’en bas de la page pour bien analyser votre dataset



1. **Preprocessing**

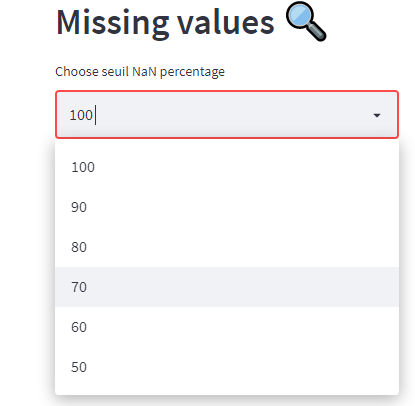
Comme définit plus haut le preproccessing ou le traitement des données fait référence aux transformations appliquées à nos données avant de les alimenter à nos algorithmes.

03 grandes notions sont abordées dans cette partie, la suppression des valeurs aberrantes (missing values), imputation statistique par la moyenne, la suppression des outliers.

**La suppression des missing values**

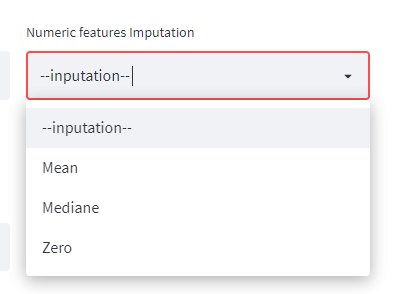
Le problème de la valeur manquante est assez courant dans de nombreux ensembles de données réels. Une valeur manquante peut fausser les résultats des modèles d'apprentissage automatique et/ou réduire la précision du modèle.

La suppression se fait par colonne, nous définissons des seuils( de 50 à 100) c’est à dire le pourcentage maximal de valeurs manquante par colonne et nous supprimons la colonne si le pourcentage dépasse ce seuil.



**Imputation statistique par la moyenne**

Après avoir supprimer les valeurs colonnes qui répondaient a notre critère de seuil il faut maintenant remplacer les valeurs manquante entre les lignes du dataset par la moyenne, ça s’appelle l’imputation statistique par la moyenne, il y’a plusieurs types d’imputation a l’instar de celle de la moyenne, on a l’imputation par la médiane (ce n’est pas une bonne solution car la médiane peut être une valeur manquante) il y’a aussi l’imputation par zéro ( ces zéros risque d’être considéré comme des outliers)



**Visualisation et suppression des outliers**

Plusieurs algorithmes de Machine Learning sont sensibles aux données d’entrainement ainsi qu’à leurs distributions. Avoir des O*utliers dans Training Set*d’un algorithme de Machine Learning peut rendre la phase d’entrainement plus longue. Sans mentionner que l’apprentissage sera biaisé. Par conséquent, le modèle prédictif produit ne sera pas performant, ou du moins, loin d’être optimal.

Pour la visualisation on a utilisé des box plots.

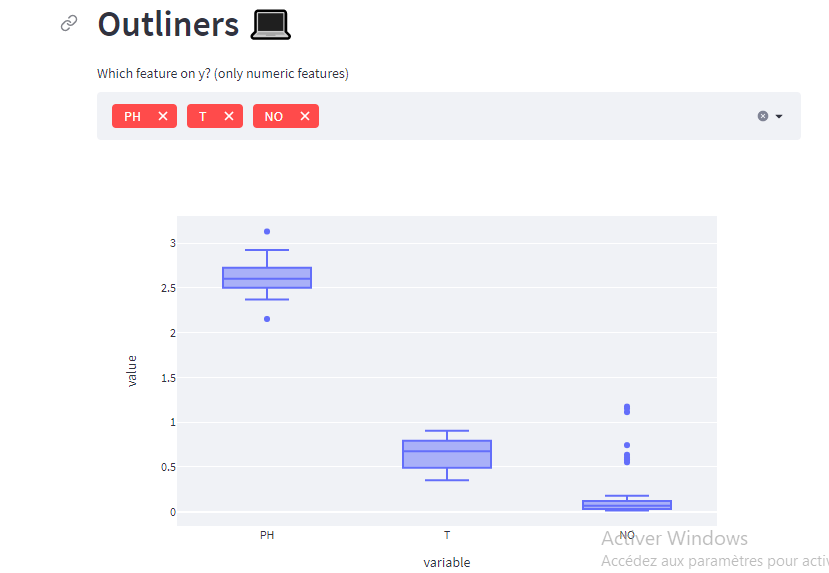
La suppression s’est faites par la méthodes de l’intervalle interquartile(IQR)

Chaque ensemble de données peut être divisé en quartiles. Le premier quartile indique que 25 % des points de données sont inférieurs à cette valeur, tandis que le deuxième quartile est considéré comme le point médian de l'ensemble de données. La méthode interquartile trouve les valeurs aberrantes sur les ensembles de données numériques en suivant la procédure ci-dessous

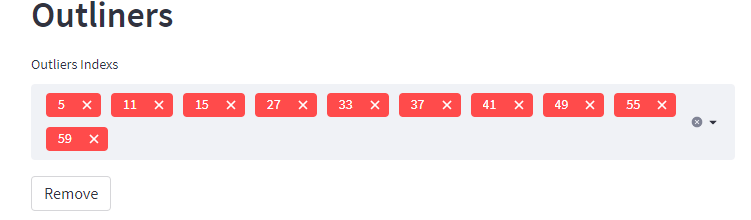
* Trouvez le premier quartile, Q1.
* Trouvez le troisième quartile, Q3.
* Calculez l'IQR. IQR= Q3-Q1.
* Définissez la plage de données normale avec la limite inférieure comme Q1–1.5\*IQR et la limite supérieure comme Q3+1.5\*IQR.
* Tout point de données en dehors de cette plage est considéré comme aberrant et doit être supprimé pour une analyse plus approfondie.

Le concept de quartiles et d'IQR peut être mieux visualisé à partir de la boîte à moustaches. Il a le point minimum et maximum défini comme Q1–1.5\*IQR et Q3+1.5\*IQR respectivement. Tout point en dehors de cette plage est aberrant.

Choisir les données a représenté par des box plots pour visualiser les outliers afin de les supprimer.



Après cette visualisation juste en bas de la page, les lignes qui sont des outliers sont affichés et il y’a un bouton « Remove » cliqué dessus si vous voulez supprimer les outliers.



1. **Features selection**

Features selection est le processus de réduction du nombre de variables d'entrée lors du développement d'un modèle prédictif.

Il est souhaitable de réduire le nombre de variables d'entrée à la fois pour réduire le coût de calcul de la modélisation et, dans certains cas, pour améliorer les performances du modèle.

Les méthodes de sélection de variables basées sur les statistiques impliquent l'évaluation de la relation entre chaque variable d'entrée et la variable cible(corrélation) à l'aide de statistiques et la sélection des variables d'entrée qui ont la relation la plus forte avec la variable cible. Ces méthodes peuvent être rapides et efficaces, bien que le choix des mesures statistiques dépende du type de données des variables d'entrée et de sortie.

Pour notre cas on a utilisé principale 2 approches, la première consiste à supprimer les variables qui n’ont pas une forte corrélation avec le Target, la deuxième consiste à évaluer l’apport de chaque feature lors de l’entrainement afin de garder les features qui ont une forte contribution.

**Encodage**

Les modèles d'apprentissage automatique exigent que toutes les variables d'entrée et de sortie soient numériques. Cela signifie que si vos données contiennent des données catégorielles, vous devez les coder en nombres avant de pouvoir ajuster et évaluer un modèle.

Les deux techniques les plus populaires sont un codage ordinal et un codage One-Hot.

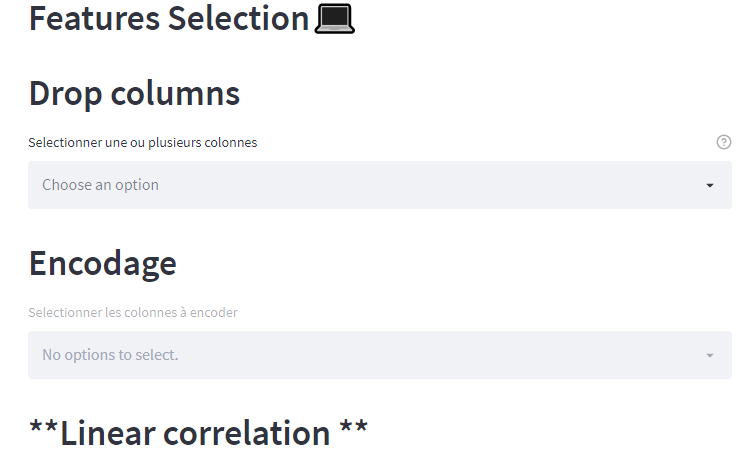
Pour notre application seule l’encodage ordinal est utilisé.

Dans l’encodage ordinal, chaque valeur de catégorie unique se voit attribuer une valeur entière.

Par exemple, "rouge" est 1, "vert" est 2 et "bleu" est 3.

C'est ce qu'on appelle un codage ordinal ou un codage entier et est facilement réversible. Souvent, des valeurs entières commençant à zéro sont utilisées.

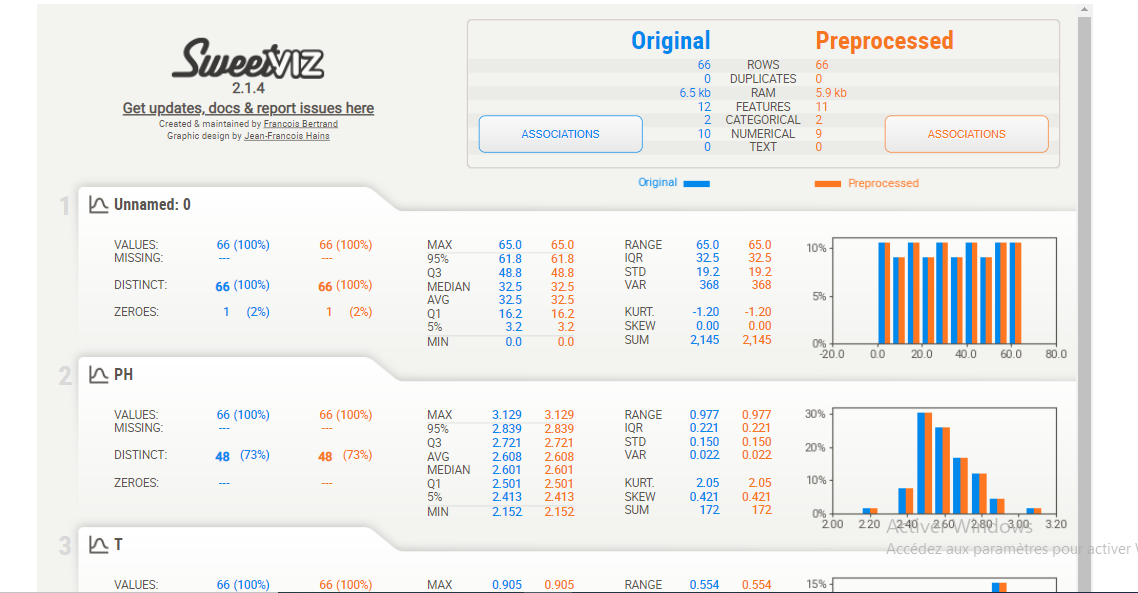
Pour certaines variables, un encodage ordinal peut suffire. Les valeurs entières ont une relation ordonnée naturelle entre elles et les algorithmes d'apprentissage automatique peuvent être en mesure de comprendre et d'exploiter cette relation.



1. **Compare**

Jusque-là nous avons fait une description du dataset, on l’a traité en supprimant les valeurs manquantes et outliers, on a du features selection et si maintenant on comparait notre dataset origine avec le dataset prétraité.

L’objectif est de faire une comparaison détaillé entre original et celui prétraité, on pourra bien voir toutes les modification qu’on a eu a apporté sur le dataset, si son est satisfait des résultats on pourra continuer sinon on revient faire d’autre traitement



**La normalisation des données**

Normalisation : La normalisation est une méthode de prétraitement des données qui permet de réduire la complexité des modèles.

Elle permet de donner la même importance à tous les features

Il existe plusieurs méthodes à l’instar de Standard scaling.

**Standard scaling**

La normalisation standardise la moyenne et l’écart-type de tout type de distribution de données, ce qui permet de simplifier le problème d’apprentissage en s’affranchissant de ces deux paramètres.

Pour effectuer cette transformation, on soustrait aux données leur moyenne empirique *m*et on les divise par leur écart-type *σ.*

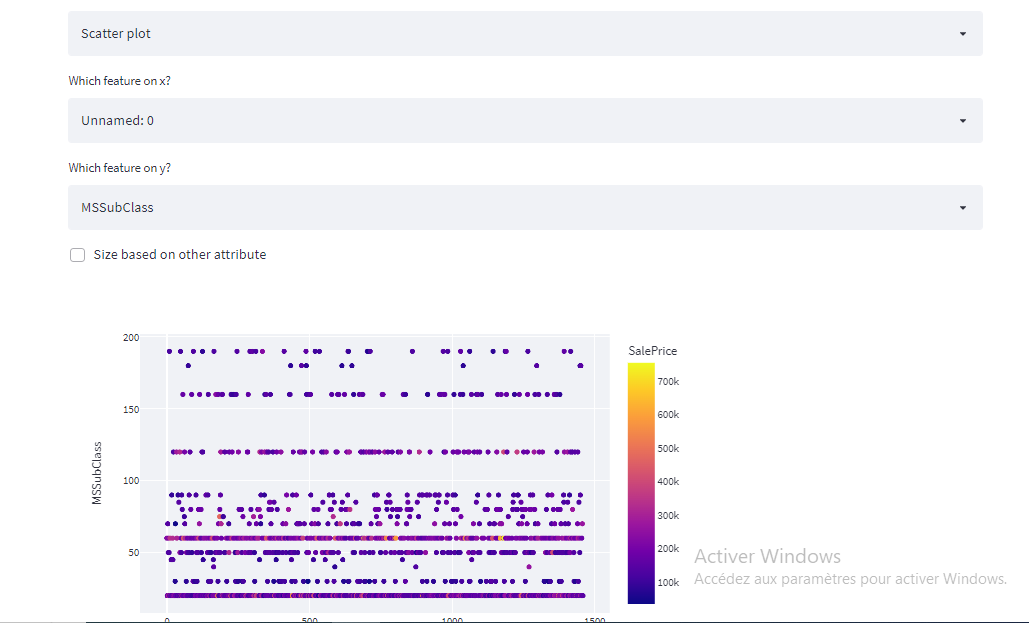
1. **Data visualisation**

La data visualisation est l’art et la manière de transformer la donnée en un formidable outil d’analyse. En montrant l’invisible, la data visualisation facilite et accélère la prise de décision.

La data visualisation simplifie la diffusion de l’information. Elle apporte des points de comparaison et d'analyse sur les tendances. Elle affine alors les prédictions sur les tendances à venir.

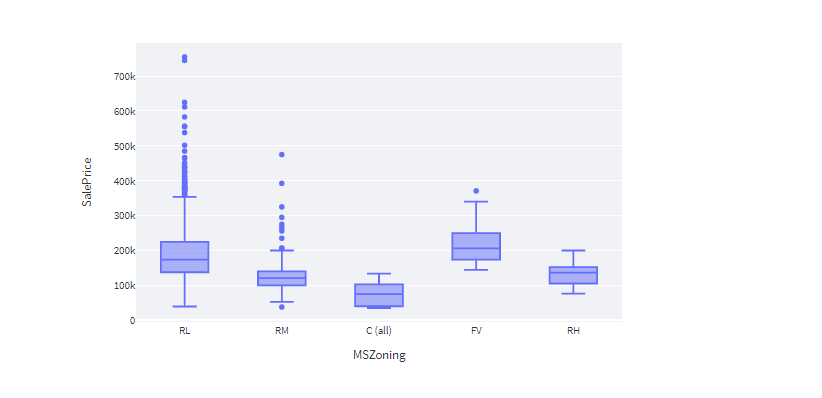
1. **Scatter plot**

Un nuage de points (également appelé  nuage de points ou diagramme de dispersion )  est un type de tracé ou de diagramme utilisant des coordonnées généralement deux variables  pour un ensemble de données.



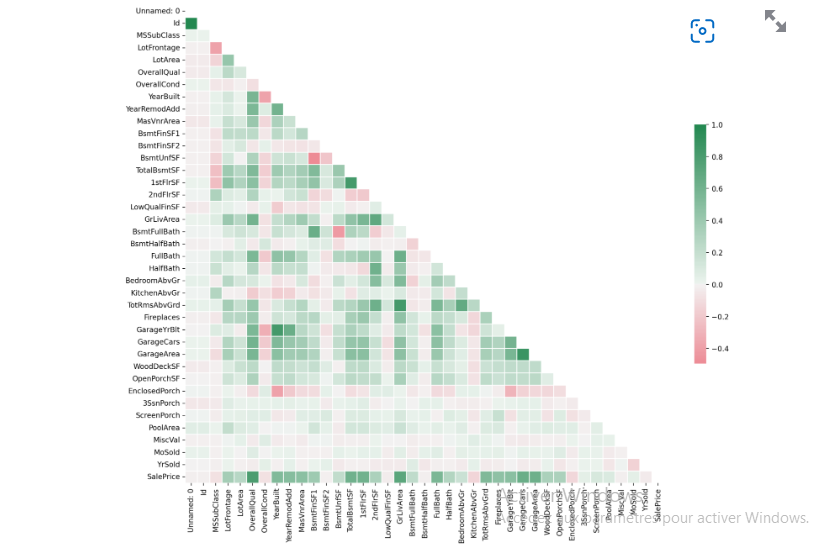
1. **Box plot**

Un box-plot est un graphique simple composé d'un rectangle duquel deux droites sortent afin de représenter certains éléments des données. La valeur centrale du graphique est la médiane (il existe autant de valeur supérieure qu'inférieures à cette valeur dans l'échantillon).



1. **Corrélation matrix**

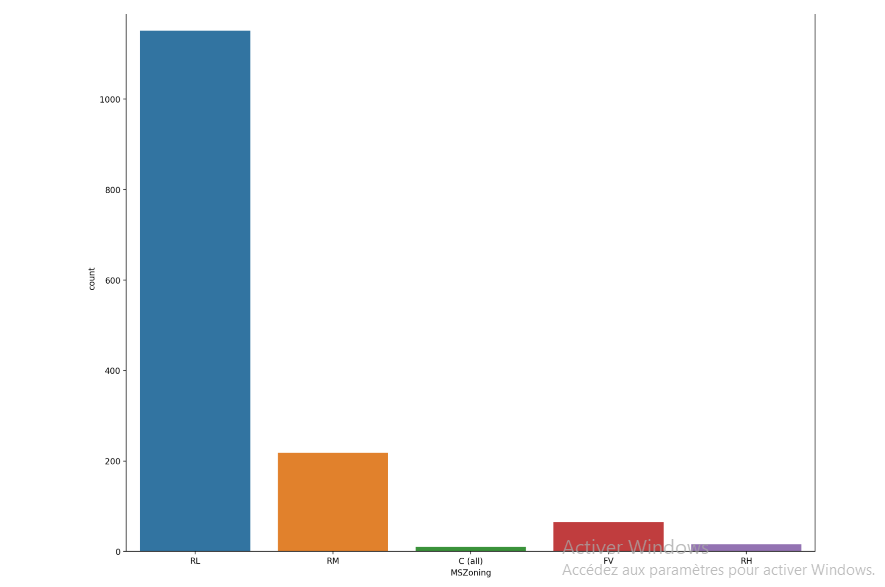
Une matrice de corrélation est simplement un tableau qui affiche les coefficients de corrélation pour différentes variables. La matrice représente la corrélation entre toutes les paires de valeurs possibles dans un tableau. C'est un outil puissant pour résumer un grand ensemble de données et pour identifier et visualiser des modèles dans les données données.



1. **Count plot**

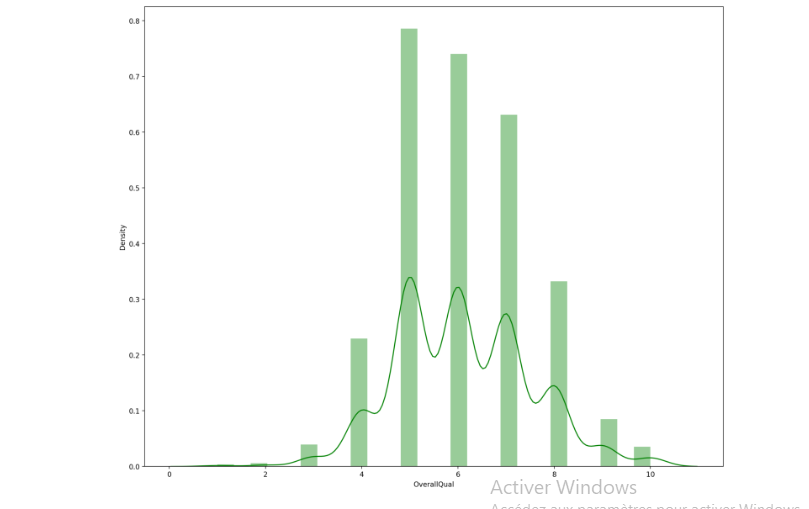
Le count plot est utilisé pour représenter l'occurrence (les comptes) de l'observation présente dans la variable catégorielle.

Il utilise le concept d'un graphique à barres pour la représentation visuelle.



1. **Distribution plot**

Les diagrammes de distribution sont d'une importance cruciale pour l'analyse exploratoire des données. Ils nous aident à détecter les valeurs aberrantes et l'asymétrie, ou à obtenir un aperçu des mesures de la tendance centrale (moyenne, médiane et mode).



1. **Machine Learning**

Apres avoir traité nos données à travers le preprocesing (prétraitement des données), les visualisées dans la partie <visualisation>, on va procéder au machine Learning autrement appelée apprentissage automatique.

Le machine Learning est une technique de programmation informatique qui utilise des probabilités statistiques pour donner aux ordinateurs la capacité d’apprendre par eux-mêmes sans programmation explicite.

Cette partie se divisera principalement en deux volets à savoir :

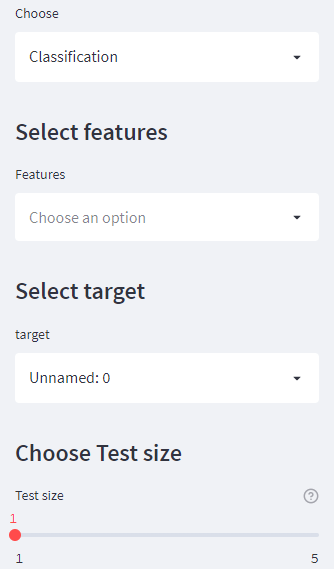
* La classification
* La régression

1. **La classification**

La classification est une partie du machine Learning où la variable de sortie est catégorie.

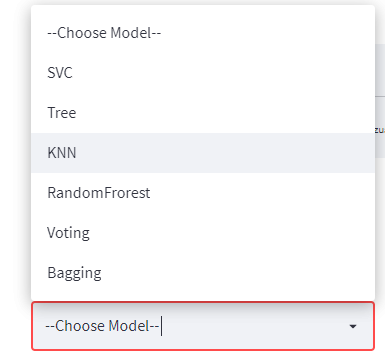
Dans cette partie on mit en place des modèles, qui une fois bien entrainés, essayeront de développer une fonction qui classifiera avec précision la sortie à partir des variables d’entrées.

Après avoir choisi la classification il faut choisir les features (variables d’entrées du modèle) et le target, ensuite on choisi le test size, le test size représente le pourcentage du donné de test (test set) et par complémentarité celui du donné d’entrainement appelé le train set.



**Choix du modèle**

Après avoir choisi les features, le target et le test size choisi le model qui vous convient entre le SVC (support vector Machine), le Tree, RandomClassifier,le KNN(k-Neighbors).



**SVC**

**Définition**

Les machines à vecteurs de support ou séparateurs à vaste marge (en anglais support-vector machine, SVM) sont un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de classification et de régression. Les SVM sont une généralisation des classifier linéaires.

Les avantages des machines à vecteurs de support sont :

* Efficace dans les espaces de grande dimension.
* Toujours efficace dans les cas où le nombre de dimensions est supérieur au nombre d'échantillons.
* Utilise un sous-ensemble de points d'apprentissage dans la fonction de décision (appelés vecteurs de support), il est donc également efficace en mémoire.
* Polyvalent : différentes fonctions du noyau peuvent être spécifiées pour la fonction de décision. Des noyaux communs sont fournis, mais il est également possible de spécifier des noyaux personnalisés.

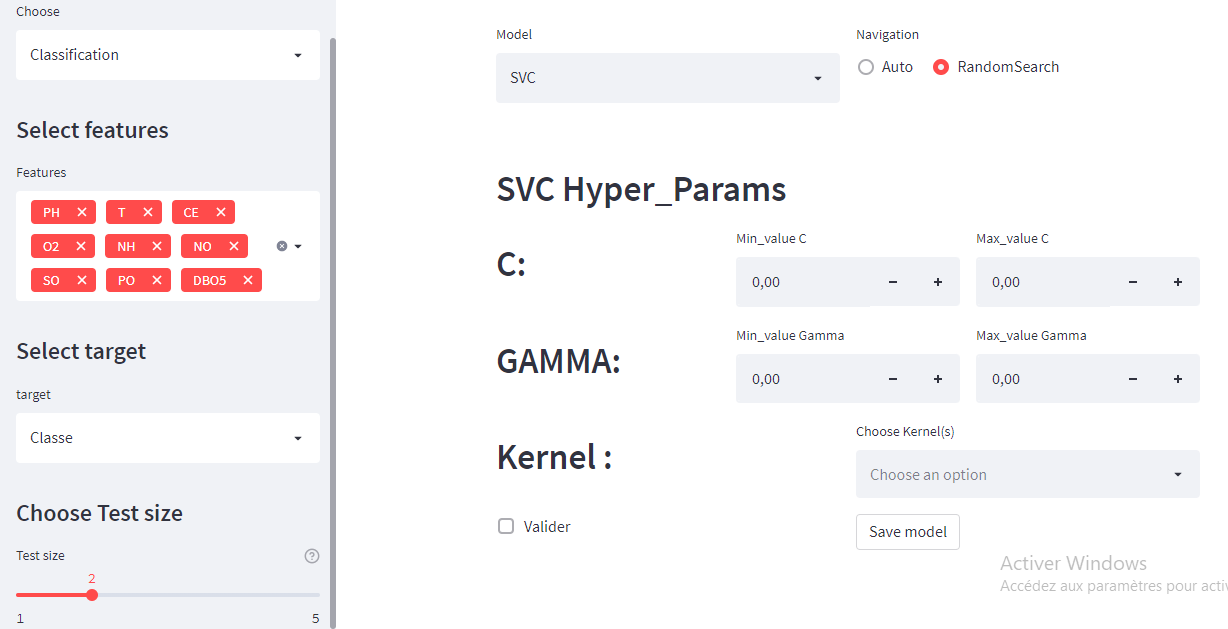
Les inconvénients des machines à vecteurs de support incluent :

* Si le nombre de caractéristiques est bien supérieur au nombre d'échantillons, évitez le sur-ajustement dans le choix des fonctions du noyau et le terme de régularisation est crucial.
* Les SVM ne fournissent pas directement des estimations de probabilité, celles-ci sont calculées à l'aide d'une validation croisée quintuple coûteuse.

Après le choix du modèle l’utilisateur devra procéder au paramétrage des hyperparametres ainsi que l’entrainement.

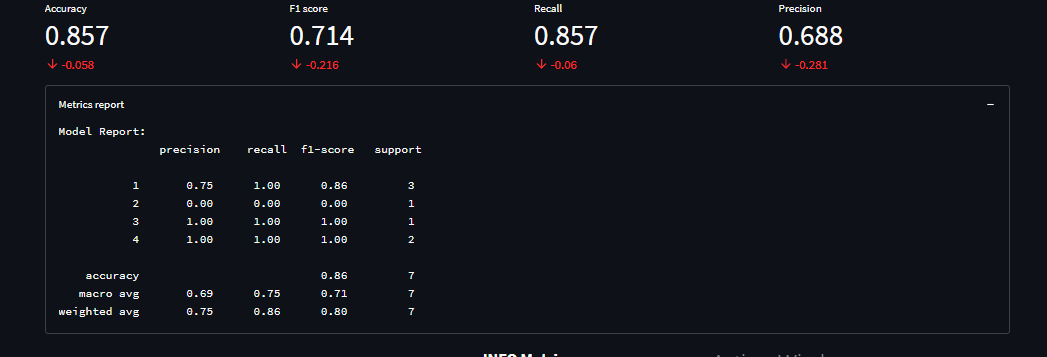
**Paramétrage du modèle (parameters tuning)**

On offre la possibilité de choisir les meilleurs hyperparametres afin que le modèle ait de très bonnes performances.



Résultats

Pour chaque modèle nous allons affiché l’accuracy,F1 score, Recall et la precision.

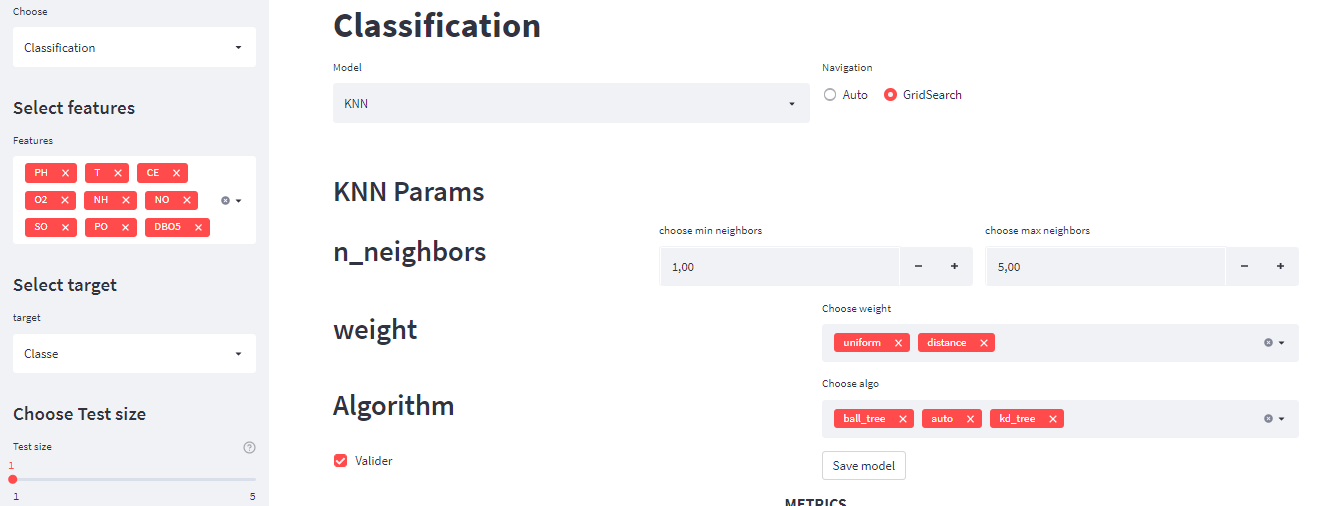


**KNN**

**Definition**

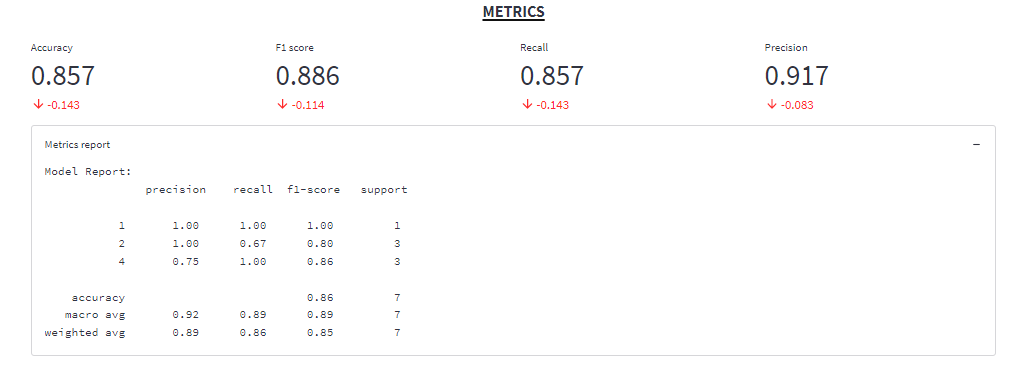
La classification basée sur les voisins est un type d’apprentissage basé sur *les instances* ou d'apprentissage *non généralisant* : il ne tente pas de construire un modèle interne général, mais stocke simplement des instances des données d'apprentissage. La classification est calculée à partir d'un vote à la majorité simple des plus proches voisins de chaque point : un point de requête se voit attribuer la classe de données qui a le plus de représentants parmi les plus proches voisins du point.

**Paramétrage du modèle (parameters tuning)**

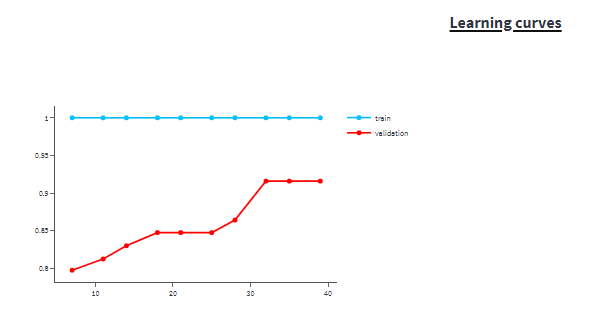


**Résultat du modèle**

Pour ces hyperparametres on obtient les metrics suivants



Et des learning curves du train set et du test set



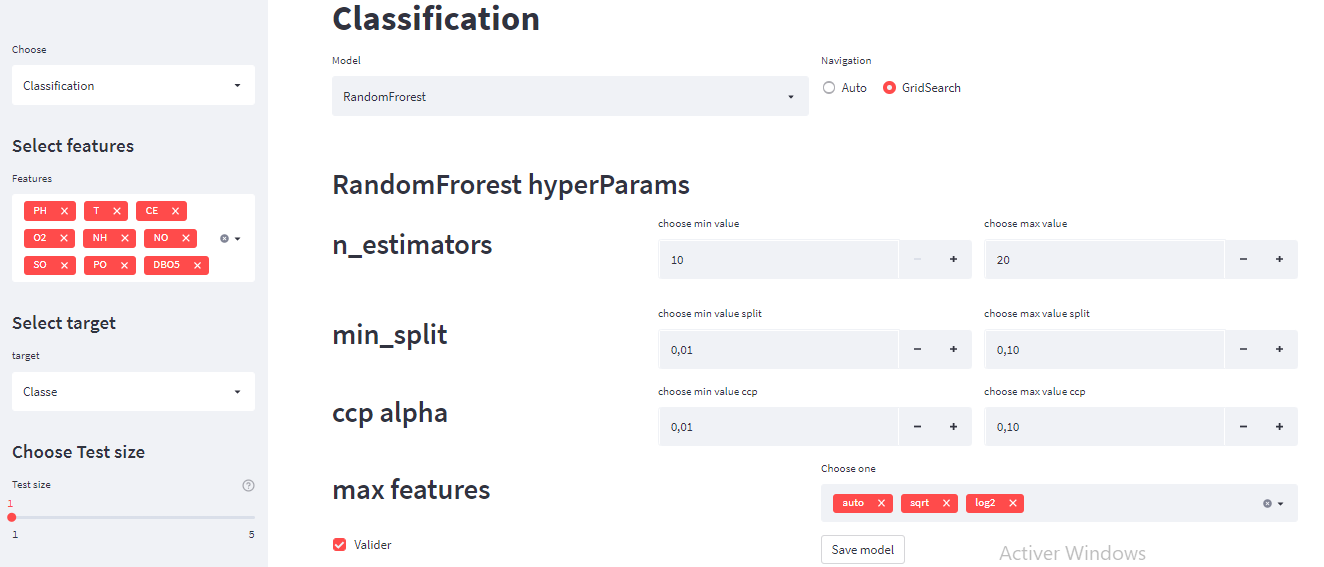
**RandomForest Classifier**

**Définition**

Les forêts aléatoires ou forêts de décision aléatoires sont une méthode d'apprentissage d'ensemble pour la classification, la régression et d'autres tâches qui fonctionnent en construisant une multitude d'arbres de décision au moment de la formation. Pour les tâches de classification, la sortie de la forêt aléatoire est la classe sélectionnée par la plupart des arbres.

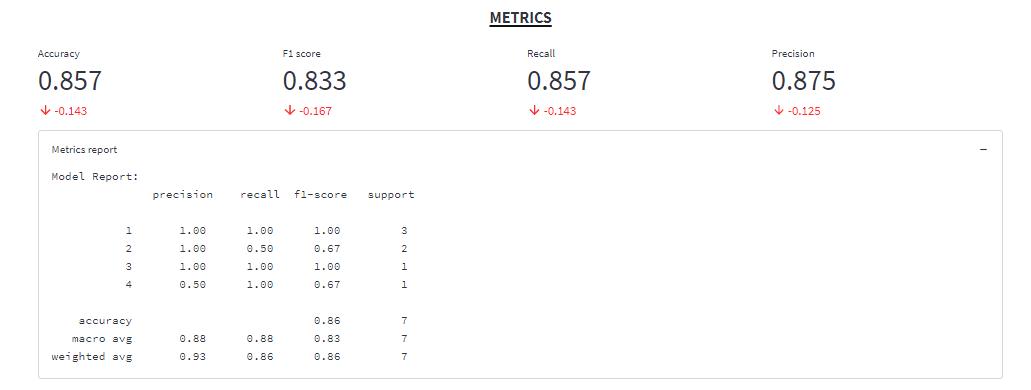
Les forêts de décision aléatoires corrigent l'habitude des arbres de décision de s'adapter à leur ensemble d'entraînement .Les forêts aléatoires surpassent généralement les arbres de décision, mais leur précision est inférieure à celle des arbres à gradient boosté. Cependant, les caractéristiques des données peuvent affecter leurs performances.

**Paramétrage du modèle (parameters tuning)**

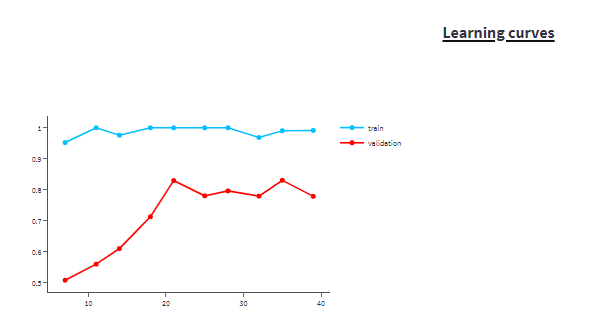


**Résultats obtenus**

Les metrics obtenus



Learning curves



**Tree**

**Définition**

L'arbre de décision est une technique d'apprentissage supervisé qui peut être utilisée à la fois pour les problèmes de classification et de régression, mais elle est généralement préférée pour résoudre les problèmes de classification. Il s'agit d'un classificateur arborescent, où les nœuds internes représentent les caractéristiques d'un ensemble de données, les branches représentent les règles de décision et chaque nœud feuille représente le résultat.

Dans un arbre de décision, il y a deux nœuds, qui sont le nœud de décision et le nœud feuille. Les nœuds de décision sont utilisés pour prendre n'importe quelle décision et ont plusieurs branches, tandis que les nœuds feuilles sont la sortie de ces décisions et ne contiennent pas d'autres branches.

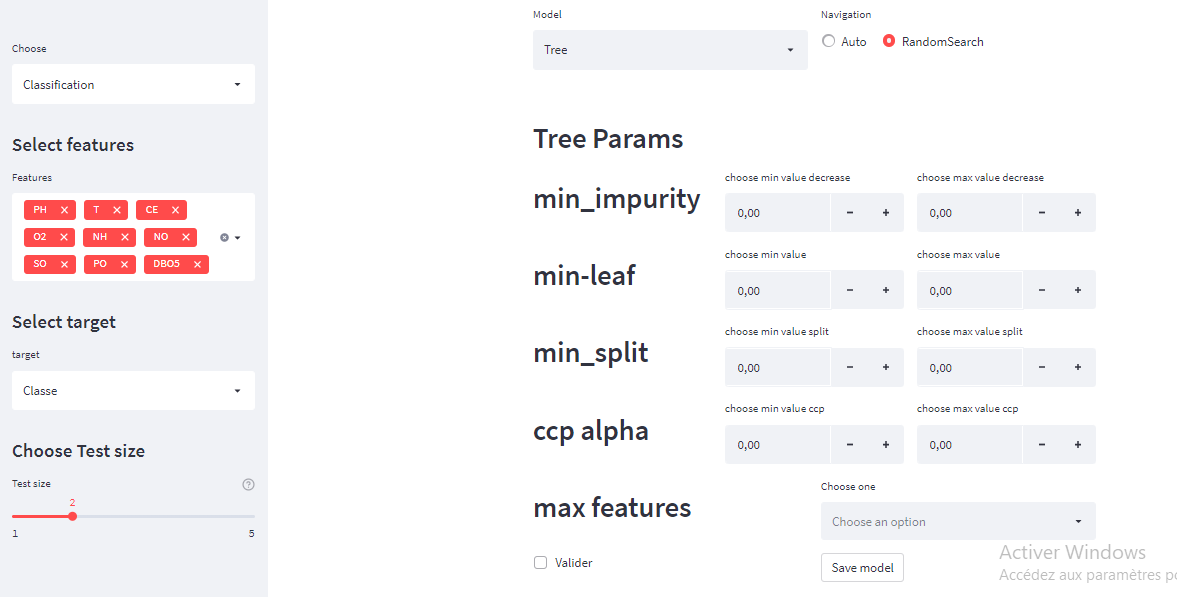
Les décisions ou le test sont effectués sur la base des caractéristiques de l'ensemble de données donné. Il s'agit d'une représentation graphique permettant d'obtenir toutes les solutions possibles à un problème/une décision en fonction de conditions données.

C'est ce qu'on appelle un arbre de décision car, semblable à un arbre, il commence par le nœud racine, qui se développe sur d'autres branches et construit une structure arborescente.

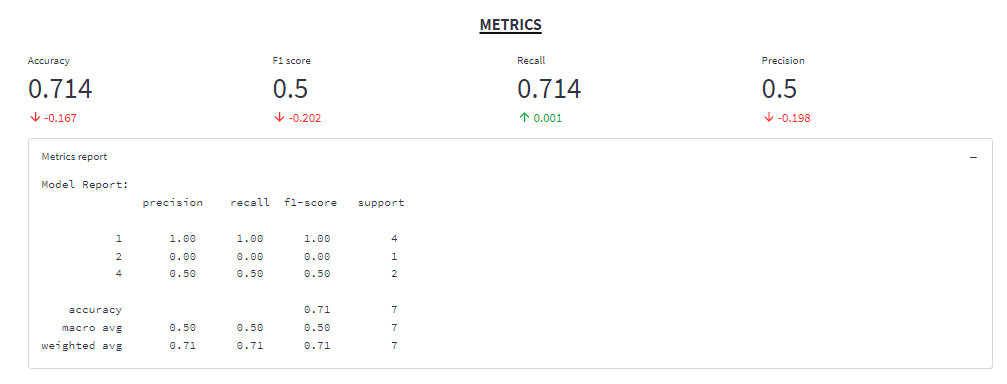
Pour construire un arbre, nous utilisons l'algorithme CART, qui signifie Classification and Regression Tree algorithm.

Un arbre de décision pose simplement une question et, en fonction de la réponse (Oui/Non), il divise ensuite l'arbre en sous-arbres.

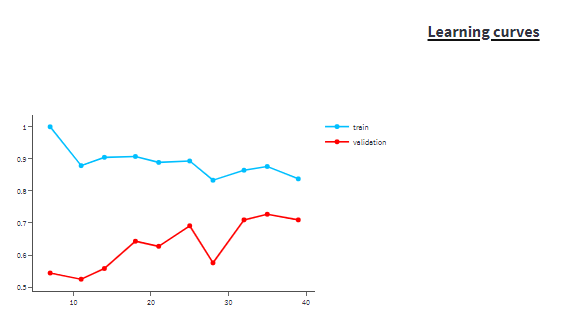
**Paramétrage du modèle ( parameters tuning)**



Resultats



Learning curve



**Voting**

**Définition**

Le vote est une méthode permettant à un groupe, tel qu'une assemblée ou un électorat, de prendre une décision collective ou d'exprimer une opinion généralement à la suite de discussions, de débats ou de campagnes électorales.

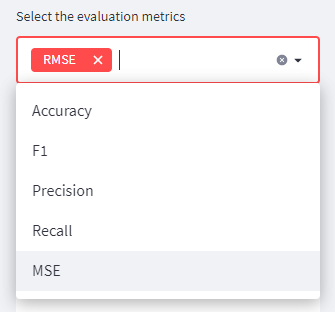
C’est le même principe qu’on appliquer pour les modèles.

1. **Régression**

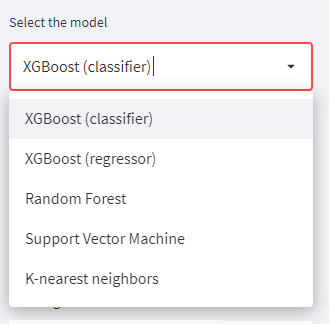
Dans cette partie on offre à l’utilisateur la possibilité diviser ses données : un pourcentage pour l’entrainement et un autre pour le test.



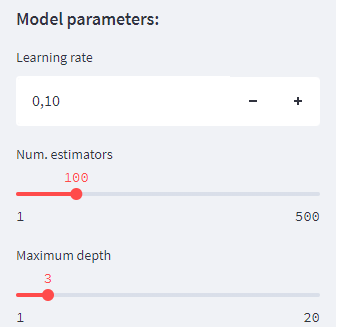
A côté de cela on offre également la possibilité de choisir la métrique pour évaluer la performance du modèle de prédiction.



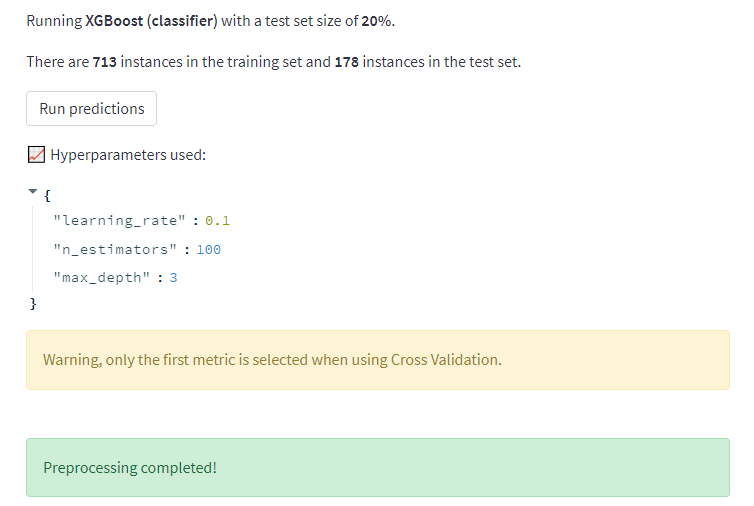
Choisir le modèle



Par ailleurs on peut également paramétrer le modèle comme dans la partie classification.



Enfin on entraine le modèle



1. **Prédiction**

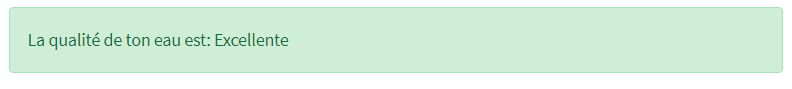
Maintenant qu’on a entrainé notre modèle, la dernière étape consistera à faire des tests, prédire la qualité avec notre modèle déjà entrainé.

Pour cela il suffit de donner des valeurs aux entrées de notre modèle.



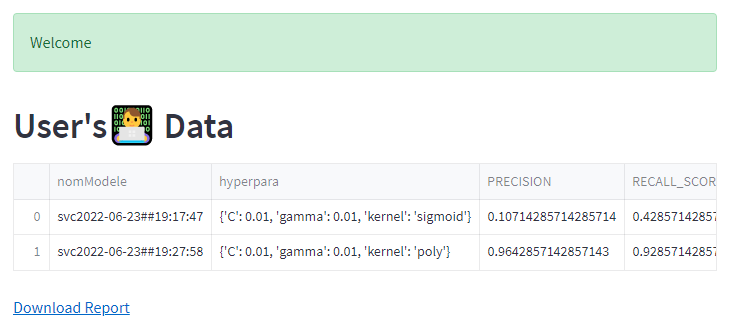


Et on obtient la prédiction suivante :

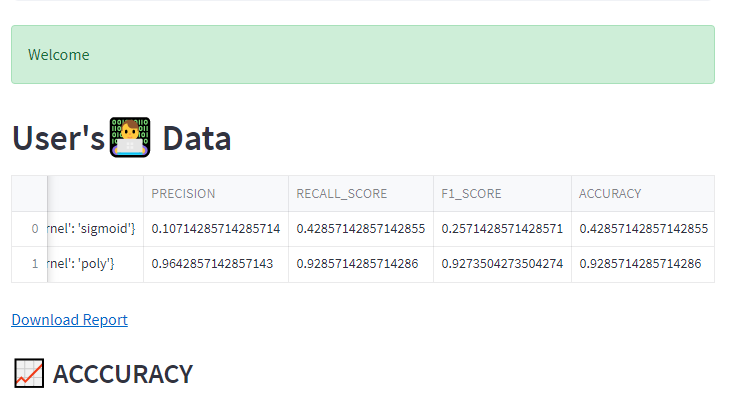


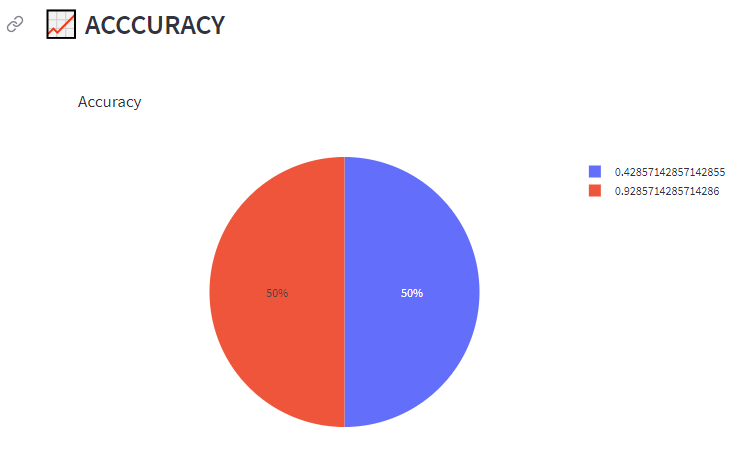
1. **Le rapport**

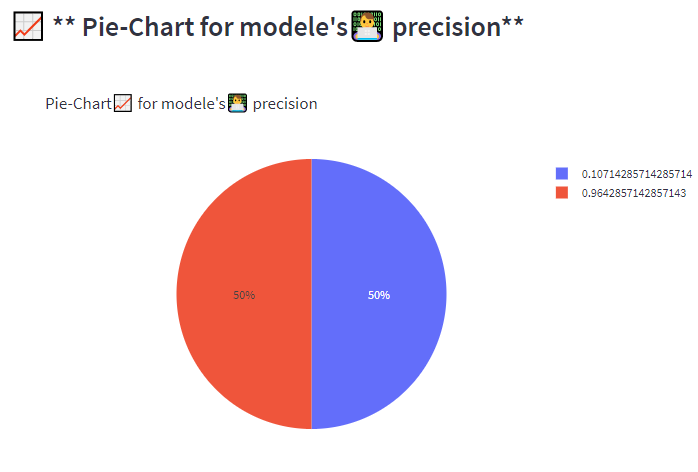
Cette dernière partie comportera un aperçue claire des modèles entrainés avec leur précision, les meilleurs hyperparamètres, l’accuracy ainsi que des graphiques pour la visualisation de ces données.

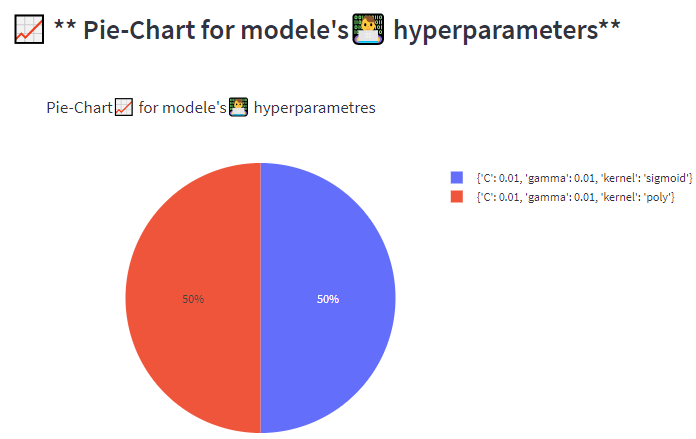


On a la possibilité de télécharger ces données pour les réutiliser après.



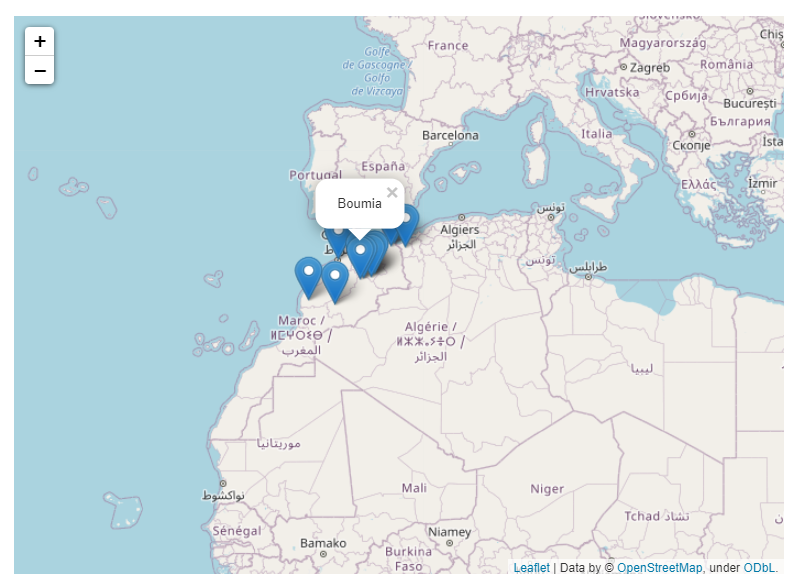






1. **Le map**





# Conclusion

Ce projet nous ont permis d’avoir une maitrise en termes de prétraitement des données et visualisation.

Pouvoir télécharger des données, les visualiser par des graphiques concis, détecter et supprimer les valeurs manquantes, le pourcentage de valeurs manquantes, entrainer des modèles d’apprentissage sur des données bien traiter et enfin sauvegarder sous forme de base de données, telles sont les principaux objectifs réalisés de ce projet.

Références

<https://github.com/antonin-lfv/Online_preprocessing_for_ML/blob/master/main.py>

Le Framework Streamlit