ENSIAS

Rapport de Projet Machine Learning

Option : Business Intelligence & Analytics (BI&A)

Analyse Prédictive pour les Admissions dans les Écoles Publiques

---

Effectués par :

SAADI Naoufal
BENALI Amin
KJAOUJ Aymane
RAFIK Iliass

Proposée par :

Mme. Benbrahim Houda

Année Universitaire 2023-2024

Table des matières

Introduction :

Lors de l'analyse du processus d'admission dans les écoles publiques, une complexité évidente émerge quant à l'évaluation des candidatures. Cette complexité résulte de la variété des critères à considérer, des influences multiples sur les décisions d'admission, et de la nécessité d'assurer une sélection équitable et transparente des candidats. Au sein de ce contexte, il devient apparent qu'il existe un besoin crucial d'amélioration dans le processus d'admission. La détection et l'évaluation des facteurs déterminants de l'admission présentent des défis, notamment en termes de subjectivité et de gestion efficace des données. Cette observation soulève la question de savoir comment appliquer des approches plus objectives et automatisées, telles que la classification multiple par machine learning, pour optimiser ce processus complexe et garantir des décisions plus informées et justes. L'importance de cette application réside dans sa capacité à automatiser l'évaluation des critères d'admission, à identifier les facteurs déterminants, et à faciliter la sélection des candidats les mieux adaptés d'une façon objective. Ensuite, on effectue nos expériences afin d'évaluer la pertinence des algorithmes de classification multi class appliquées à l' ensemble des données. Comme métriques, on a utilisé

quelques critères de performance adéquats à notre problème (précision , recall et F1 score ..).

Enfin, nos expériences sont intéressantes du point de vue machine learning car elles abordent de manière directe et pragmatique le défi des classes déséquilibrées, offrant ainsi des insights précieux pour l'adaptation et l'optimisation des modèles dans des contextes similaires.

I. Travaux connexes et etat de l'art :

Dan le même contexte, Manuel Olave du Centre international des entreprises publiques à Ljubljana, en Yougoslavie, Vladislav Rajkovic de l'Institut Jozef Stefan à Ljubljana, en Yougoslavie, et de l'École des sciences organisationnelles à Kranj, en Yougoslavie, ainsi que par Marko Bohanec de l'Institut Jozef Stefan à Ljubljana, en Yougoslavie,ont rédigé un travail "An application for admission in public school systems" contribuant au ce domaine ,en présentant une exploration complète d'un système expert adapté à la tâche complexe des admissions dans les écoles maternelles publiques. Le système présenté utilise un système expert qui évalue, classe et classe systématiquement les candidatures, mettant en avant l'explication des connaissances sous-jacentes et des solutions proposées par le système.Ce Système expert au cœur de ce travail a été développé en utilisant DECMAK, une coquille de système expert adaptée à la prise de décision multi-attributs. Le chapitre explore en profondeur le contexte du problème de sélection, offre un aperçu concis de la coquille DECMAK, une description détaillée de l'architecture et du processus de développement du système expert, des résultats pratiques obtenus lors de son application, et une analyse de l'impact de la normativité sur les résultats.

Généralement , nos méthodes de machine learning adoptent une approche plus automatisée, apprenant des modèles à partir des données, tandis que les systèmes experts reposent sur des règles explicitement définies par des experts humains. Le choix entre les deux approches dépend des exigences spécifiques du problème d'admission dans les écoles maternelles, de la disponibilité d'une expertise humaine, et de la nécessité d'interprétabilité des décisions du système.

II. Données :

En explorant les sites des datasets connues, on a trouvé"UCI Nursery Data Set" de format csv dans Kaggle .

- Exploration des données :

	parents	has_nurs	form	children	housing	finance	social	health	final evaluation
0	usual	proper	complete	1	convenient	convenient	nonprob	recommended	recommend
1	usual	proper	complete	1	convenient	convenient	nonprob	priority	priority
2	usual	proper	complete	1	convenient	convenient	nonprob	not_recom	not_recom
3	usual	proper	complete	1	convenient	convenient	slightly_prob	recommended	recommend
4	usual	proper	complete	1	convenient	convenient	slightly_prob	priority	priority

L’ensemble des données comporte 12960 enregistrements ,et 9 colonnes .

On peut diviser les colonnes et ses valeurs en catégories pour les expliquer :

1 - Profession des parents et caractéristiques de la crèche :

a) Parents' Occupation (parents): (1) usual, (2) pretentious, (3) of great pretension

usuel : Les parents exercent des professions ordinaires ou courantes

Prétentieux : Les parents ont des professions qui sont perçues comme cherchant à impressionner ou à paraître plus importantes qu'elles ne le sont réellement

De grande prétention : Les parents occupent des postes ou des positions socialement prestigieuses ou hautement considérés.

b) Child's Nursery (has_nurs): (1) proper, (2) less proper, (3) improper, (4) critical, (5) very critical; une évaluation graduelle de la pertinence de l'enfant pour la garderie.

2 - Structure familiale et situation financière

c) Form of the family (form): (1) complete, (2) completed, (3) incomplete, (4) foster;

d) Number of children (children): 1, 2, 3 or more;

e) Housing conditions (housing): (1) convenient, (2) less convenient, (3) critical;

f) Financial standing of the family (finance): (1) convenient, (2) inconvenient;

3 - Statut social et de santé de la famille

g) Social conditions (social): (1) non-problematic, (2) slightly problematic, (3) problematic;

h) Health conditions (health): (1) acceptance is not recommended, (2) acceptance is recommended, (3) priority acceptance is recommended;

La variable cible : Final Evaluation (Target)

not recommended (1) recommend (2), priority acceptance (3) ,special priority (4)

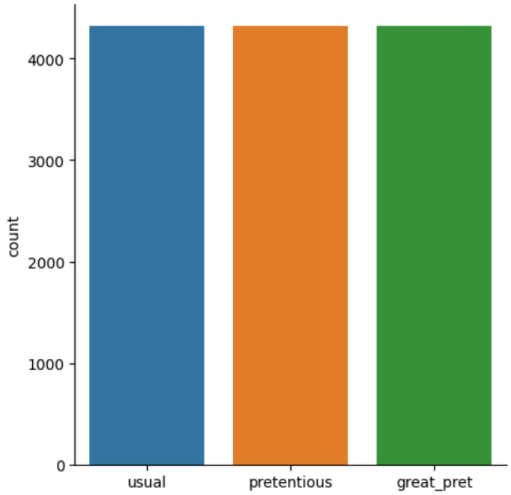
- On remarque que les valeurs de ces colonnes sont distribuées uniformément ce qui peut être bénéfique pour la construction de modèles.
- L’ensemble des données ne présente pas des valeurs manquantes.

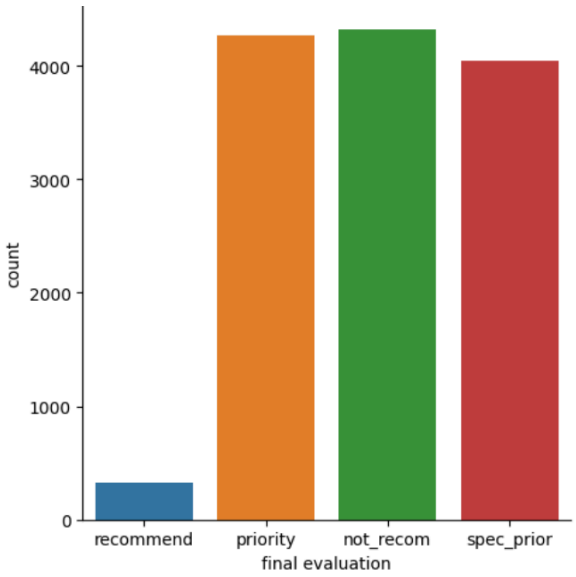
- Visualisation des données :

- Un exemple de visualisation d'un variable : barplot d'occupation des parents (parents) (à gauche) et la variable cible final evaluation (à droite)

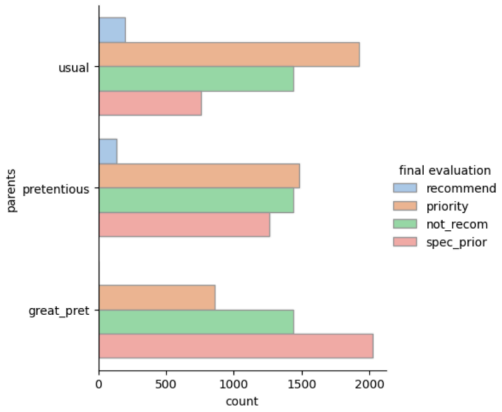
```
#bar plot d'occupation des parents
sns.catplot(data=data, x="parents", kind="count")
```

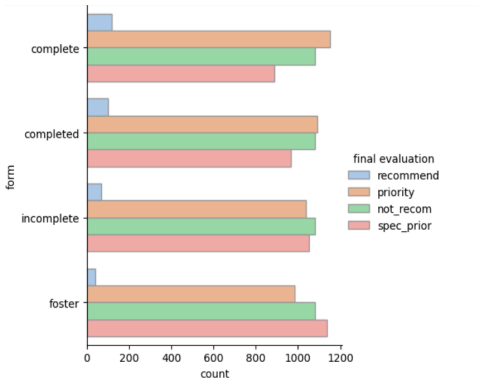
```
<seaborn.axisgrid.FacetGrid at 0x7e752d7a1990>
```





- Un exemple de visualisation d'un variable :Répartition des évaluations finales en fonction des professions des parents(à gauche) et Répartition des évaluations finales en fonction de la structure de la famille (à droite)





- Les candidats possédant des parents de grande prétention présentent le plus grande pourcentage au voie special priority.
- On remarque que les candidats issues d'une famille complète ont le plus pourcentage élevé au niveau voie priority , tandis que les candidats du famille de type foster présente pourcentage élevé au niveau du voie special priority .

III. Algorithmes ML :

1- K-Modes :

C’est un algorithme de clustering spécialement conçu pour traiter des ensembles de données catégorielles,L'objectif principal est de regrouper les observations en clusters de manière à minimiser la dissimilarité entre les éléments d'un même cluster, tout en maximisant la dissimilarité entre les clusters.

Paramètres utilisés :

n_clusters (Nombre de clusters) : Le nombre de clusters que l'algorithme doit former.

init (Initialisation) : L'argument init='Huang' spécifie une méthode particulière d'initialisation des centroids. Dans cet exemple, la méthode d'initialisation de Huang est utilisée, qui vise à obtenir une répartition initiale des centroïdes maximisant la diversité dans les clusters.

n_init (Nombre d'initialisations) : Le nombre d'initialisations différentes de l'algorithme avec différentes répartitions initiales des centroids. Le modèle sera ajusté plusieurs fois avec différentes initialisations, et la meilleure solution sera retenue.

verbose (Verbeux) : L'argument verbose=1 permet d'afficher des informations détaillées pendant l'ajustement du modèle, ce qui peut être utile pour suivre le progrès de l'algorithme.

2- Agglomerative Clustering :

C’est une méthode de clustering hiérarchique qui commence par considérer chaque observation comme un cluster distinct. Ensuite, elle fusionne progressivement les clusters les plus proches les uns des autres jusqu'à ce qu'un nombre souhaité de clusters soit atteint. Cette méthode construit une structure arborescente appelée dendrogramme, qui illustre les étapes de fusion des clusters.

Paramètres utilisés :

n_clusters (Nombre de clusters) : n_clusters=5 indique que l'algorithme doit former cinq clusters.

linkage (Méthode de liaison) : L'argument linkage='complete' spécifie la méthode de liaison utilisée pour mesurer la distance entre les clusters. Dans cet exemple, la méthode de liaison complète est utilisée, ce qui signifie que la distance entre deux clusters est mesurée par la plus grande distance entre leurs points respectifs.

affinity (Affinity) : L'argument affinity='hamming' indique que la distance de Hamming est utilisée comme mesure de la distance entre les observations. La distance de Hamming est adaptée aux données catégorielles où les variables sont en termes de catégories ou d'étiquettes.

3 - KNeighborsClassifier :

Paramètres utilisés :

n_neighbors (Nombre de voisins) : Le nombre de voisins à considérer lors de la prédiction. Un choix approprié dépend de la nature des données et peut affecter la performance du modèle. (5par défaut)

weights (Poids) : L'argument weights='distance' spécifie que la pondération des voisins est basée sur l'inverse de leur distance. Cela signifie que les voisins plus proches d'un point de requête auront une influence plus importante que les voisins plus éloignés. Cette pondération basée sur la distance est utilisée pour donner plus de poids aux voisins les plus proches dans la prise de décision.

4- DecisionTreeClassifier :

Paramètres utilisés :

criterion (Critère) : L'argument criterion='gini' spécifie le critère utilisé pour mesurer la qualité d'une division d'un nœud. Dans cet exemple, l'indice de Gini est utilisé. L'indice de Gini mesure l'impureté des classes dans un nœud et cherche à minimiser cette impureté lors de la construction de l'arbre.

class_weight (Poids de classe) : L'argument class_weight='balanced' indique que l'arbre de décision tiendra compte du déséquilibre de classe lors de la construction de l'arbre. En utilisant 'balanced', le modèle attribuera automatiquement des poids aux classes inversément

proportionnelles à leur fréquence dans l'ensemble de données. Cela est particulièrement utile lorsque les classes sont déséquilibrées.

random_state : L'argument random_state est utilisé pour initialiser la génération de nombres aléatoires. Fournir une valeur fixe à random_state garantit la reproductibilité des résultats.

5- Support Vector Machine (SVM) avec l'approche One vs Rest :

Pour effectuer la classification multiclasse. L'approche One vs Rest implique la formation d'un classificateur binaire pour chaque classe. Dans le contexte de la classification multiclasse, cela signifie qu'un classificateur est formé pour chaque classe pour prédire si une observation appartient à cette classe ou non.

Paramètres utilisés :

class_weight (Poids des classes) : L'argument class_weight=class_weights spécifie le poids des classes dans le modèle SVM. Le paramètre class_weights est une variable précédemment définie contenant les poids spécifiques attribués à chaque classe. Cela est utilisé pour traiter les classes déséquilibrées en attribuant plus de poids aux classes moins fréquentes. Par exemple, si une classe est sous-représentée, elle peut recevoir un poids plus élevé pour compenser.

6- Multi-layer Perceptron (MLP) :

models.Sequential est utilisé pour créer un modèle séquentiel. Un modèle séquentiel est une pile linéaire de couches, où chaque couche a exactement un tenseur d'entrée et un tenseur de sortie.

Trois couches Dense sont empilées les unes sur les autres :

Première couche (64 neurones,fonction d'activation ReLU,input_shape: la forme des données d'entrée.) | **Deuxième couche (** 32 neurones,fonction d'activation ReLU.)| **Troisième couche :(**4 neurones,fonction d'activation Softmax) produit une distribution de probabilités sur les différentes classes.

IV. Methodologie d'evaluation:

On peut évaluer un modèle machine learning en utilisant diverses critères de performance :

Accuracy

Précision: le nombre de vrais positifs divisé (TP) par le nombre de vrais positifs (TP) plus le nombre de faux positifs (FP).

Recall (Rappel):est le nombre de vrais positifs (TP) divisé par le nombre de vrais positifs (TP) plus le nombre de faux négatifs (FN).

F1 score: est la moyenne harmonique de la précision et du rappel. Il est utile lorsqu'on veut trouver un équilibre entre la précision et le rappel.

Courbe ROC et AUC: la courbe ROC (Receiver Operating Characteristic) et l'aire sous la courbe (AUC) sont utilisées pour évaluer la capacité d'un modèle à discriminer entre les classes. Plus l'AUC est proche de 1, meilleure est la performance.

Matrice de confusion: montre le nombre de vrais positifs, de vrais négatifs, de faux positifs et de faux négatifs. Cela nous permet de voir où notre modèle fait des erreurs et dans quelle mesure.

VI. Résultats expérimentaux

Pour le KNN on a obtenu les résultats suivants:

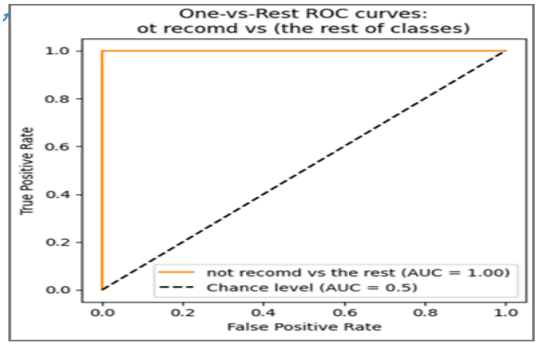
	precision	recall	f1-score	support
0	1.00	1.00	1.00	1320
1	0.89	0.88	0.88	1272
2	0.95	0.35	0.51	106
3	0.94	0.89	0.91	1190
micro avg	0.94	0.91	0.93	3888
macro avg	0.94	0.78	0.83	3888
weighted avg	0.94	0.91	0.92	3888
samples avg	0.91	0.91	0.91	3888

Les résultats pour le SVC(One vs Rest) sont:

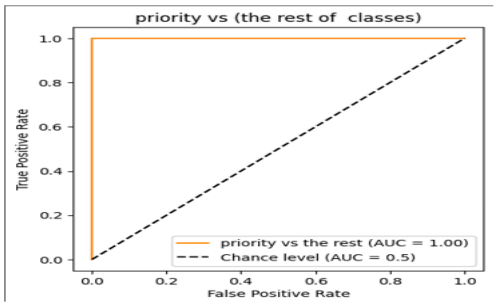
	precision	recall	f1-score	support
0	1.00	1.00	1.00	1320
1	0.92	1.00	0.96	1272
2	0.99	1.00	1.00	106
3	1.00	1.00	1.00	1190
micro avg	0.97	1.00	0.99	3888
macro avg	0.98	1.00	0.99	3888
weighted avg	0.97	1.00	0.99	3888
samples avg	0.99	1.00	0.99	3888

Les courbes ROC et les valeurs AUC:

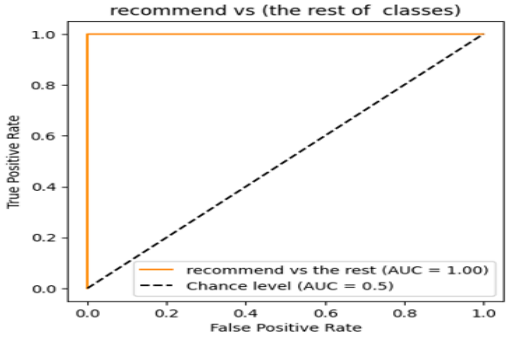
Pour classe "not recomended"



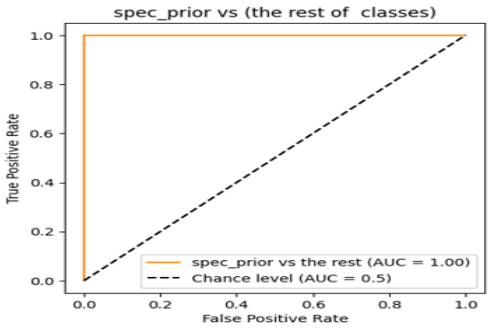
Pour classe "priority"



Pour classe "recomended"



Pour classe "special priority"

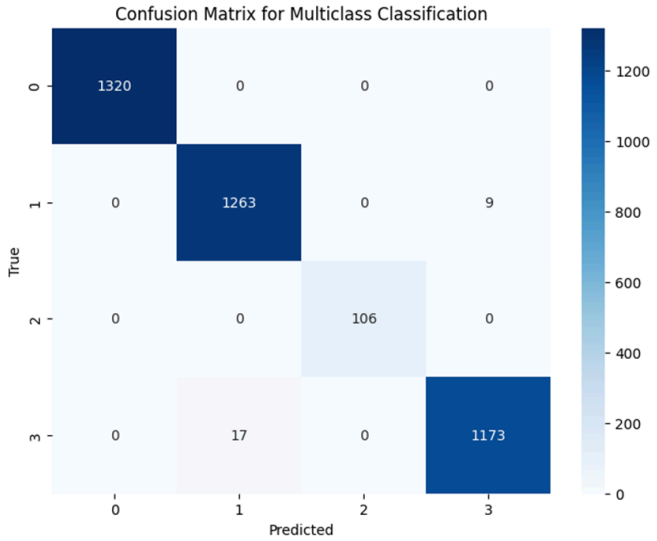


Pour les Les arbres de décision on a les résultats:

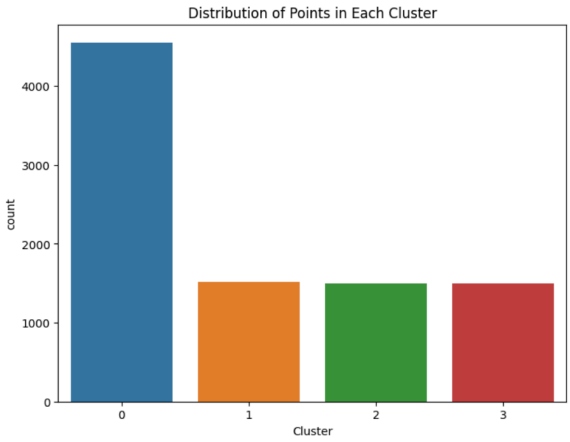
	precision	recall	f1-score	support
0	1.00	1.00	1.00	1320
1	0.99	0.99	0.99	1272
2	1.00	1.00	1.00	106
3	0.99	0.99	0.99	1190
micro avg	0.99	0.99	0.99	3888
macro avg	0.99	0.99	0.99	3888
weighted avg	0.99	0.99	0.99	3888
samples avg	0.99	0.99	0.99	3888

0.9933127572016461

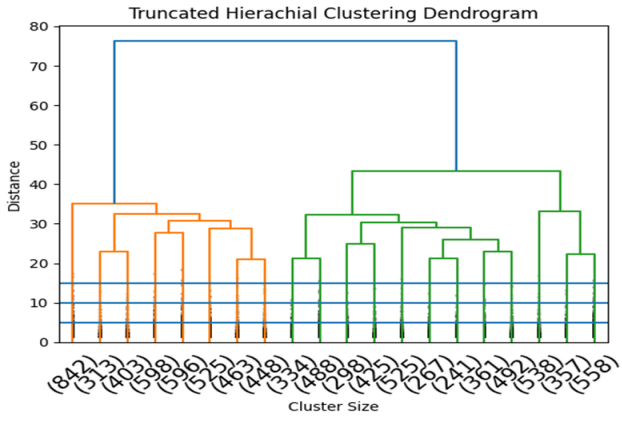
Pour le MLP on a la matrice de confusion :



Les résultats de K-modes :



et Agglomerative Clustering :



VI. Discussion

Pour le KNN :

- Comme on peut remarquer que la précision est >0.88 pour toutes les classes, ce qui indique que la majorité des prédictions positives sont correctes pour toutes les classes.

- De même le Recall est relativement grand pour toutes les classes sauf en classe 2, ce qui montre que le modèle identifie correctement la majorité des instances des classes 0,1 et 3 mais juste 35% d'identifications correctes pour classe 2.

- Pour le F1 score on a une performance excellente pour la classe 0, une performance très bonne et bonne pour les classes 3 et 1 respectivement mais une performance moyenne pour la classe 2.

Pour le SVC :

- Ces valeurs signifie que ce modèle semble avoir une excellente capacité à effectuer des prédictions précises pour chaque classe de manière uniforme, sans montrer de biais particulier. Cela est aussi supporté par les courbes ROC et les valeurs AUC .

Pour le arbres de décision :

On remarque que ces valeurs de précision, recall et F1 score sont >0.99 pour toutes les classes. Cela indique que le modèle admet une capacité excellente à prédire les classes des données.

Donc l'utilisation des paramètres spécifiés dans la partie algorithmes ML, ont permis d'obtenir une bonne classification .

Pour Kmodes :

Pour ce modèle on a calculé le score silhouette qui est une métrique utilisée pour évaluer la qualité du clustering. Sa valeur pour notre modèle est: 0.04253042860958706 qui est très proche de 0. Cela suggère que les clusters formés par le modèle K Modes ne sont pas très distincts. En d'autres termes, les points à l'intérieur d'un cluster ne sont pas beaucoup plus proches les uns des autres par rapport aux points des autres clusters.

En résumé, le score silhouette indique que la performance de clustering du modèle K Modes est loin d'être idéale, et qu'il sera mieux d'exploiter d'autres techniques ou de modifier l'approche de clustering.

VII. Conclusion :

En conclusion, l'analyse comparative des performances des modèles d'apprentissage automatique (KNN, SVC, arbres de décision,MLP) met en lumière des caractéristiques distinctes de chacun.

La classification des différentes classes a présenté de bons résultats pour la majorité de ces algorithmes .Mais pendant la tâche de clustering , les données sans variable cible ont posé un challenge pour les deux modèles K-modes et Agglomerative Clustering. Ces deux derniers ont donné des groupes avec performance moyenne.

Ces résultats sont importants car ils peuvent également suggérer des domaines où des améliorations pourraient être apportées, par exemple, en ajustant d'autres paramètres du modèle ou en utilisant des techniques d'ingénierie des caractéristiques, ou la proposition des nouveaux algorithmes .Ainsi que les conclusions tirées de ces évaluations sont essentielles pour prendre des décisions éclairées lors de la mise en œuvre d'un modèle dans un environnement réel.

Enfin et comme perspective, on peut penser à utiliser d'autres modèles ou architecture des réseaux de neurones , ou chercher d' autres datasets qui peuvent faciliter la tâche plus.

VII. ANNEXE :

Voir le notebook associer “Projet_BI&A.ipynb”