DÉPARTEMENT INFORMATIQUE

Projet de Session : Stroke Prediction

IFT603 - Techniques d'apprentissage

Auteurs:

Noms:
Maman Souley, Aicha (mama3101)
Sangare, Mahamadou (sanm0301)

Superviseur:

Martin Vallières

Session Hiver 2022

Contents

1 Introduction

1.1 Description du projet

Au Québec, environ 20 000 personnes par année subissent un AVC. Il est estimé que 130 000 personnes ayant subi un AVC vivent avec des séquelles tant psychologiques que physiques. Les impacts sur la qualité de vie de ces personnes et de leurs proches sont très importants, voire parfois dramatiques.

L'accident vasculaire cérébral (AVC) est un problème de santé non planifié et urgent qui doit être dépisté et traité rapidement. Sa cause similaire à celle de l'infarctus du myocarde est le résultat d'une diminution de l'apport sanguin dans le cœur ou dans le cerveau.Un AVC peut affecter les fonctions motrices (paralysie) mais aussi, le langage, la pensée, les capacités d'apprentissage et de communication et les émotions. Il est important de rappeler que vous pouvez agir pour vous protéger de l'AVC.

Notre projet portera sur une analyse de prédiction des accidents vasculaires cérébraux à l'aide d'un certains nombre d'algorithme d'apprentissage automatique utilisant un ensemble de données sur l'état de la santé, y compris divers types de facteurs de risque.

1.2 Nos motivations

L'AVC étant une maladie qui touche beaucoup la génération contemporaine, nous avons jugé utile d'étudier ce sujet afin d'effectuer un travail scientifique qui permettra de prédire cette maladie.

Le sujet est de prime abord, un sujet d'actualité. Au-delà, il traite d'un problème réel qui affecte plusieurs personnes de nos jours. Notre intérêt pour le sujet tient source de son ampleur ainsi que son importance pour le secteur de la santé en général et la médecine en particulier. En tentant de répondre aux différentes questions posées comme problématique, nous espérerons apporter notre contribution, peu soit elle, pour palier à ce problème.

2 Problématique

2.1 Problématique du sujet

Au Canada l'AVC est la 3e cause de décès et la recause d'incapacité grave chez l' adulte. La prise en charge rapide et l'offre de soins structurés aux personnes victimes d'AVC/accident ischémique transitoire (AIT) sont des facteurs déterminants pour la récupération et la qualité de vie suite à l' événement initial.

L'AVC survient lorsque la circulation sanguine dans une artère s'arrête, soit parce que cette dernière est bloquée ou éclaté. Ce blocage ou ces dégâts signifient que les cellules du cerveau ne reçoivent plus l'oxygène ni les nutriments nécessaires, ce qui provoque leur mort. Cela peut entraîner une maladie cérébrovasculaire.Cependant dans notre projet on va essayer de prédire en fonction de nos attributs **quels sont les individus susceptible d'avoir l'AVC ?** Une autre problématique à poser c'est de savoir quels sont les variables qui provoque cette maladie.On verra aussi si l'hypertension , l'âge ou bien le genre influence cette maladie.Mais la question farce de notre projet est de savoir quelle est le meilleure modèle qui décrit mieux nos données.
Dans notre étude nous tenterons de répondre à ses différentes questions.

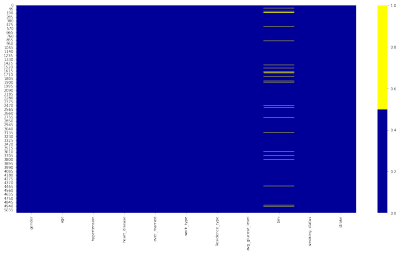
3 Méthodologie

3.1 Traitement des données

Le traitement de données est indispensable surtout lorsque l'on traite un dataset d'une certaine ampleur. Ainsi, nous avons fait le choix d'appliquer certaine méthode permettant de nettoyer et alléger sans pertes notre dataset. Afin de ne pas surcharger nos

machines, nous nous sommes lancés, comme il est coutume, dans la recherche de valeurs null qui viendraient fausser notre futur modèle. Ainsi, pour cela, nous avons pu mettre en place tant une visualisation qu'une interprétation numérique nous permettant de mettre en évidence les attributs ayant le plus de valeur null afin que l'on puisse assister la décision accompagnant cette recherche. En effet, il ne suffit pas d'exposer les valeurs null dans notre dataset mais bien de traiter ce problème. Ce dataset à quelques particularités

notoires qui ont nécessité une attention particulière. Tout d'abord, nous avons juste un attributs qui possèdent un grand nombre de valeurs *null* ce qui nous à fait nous poser ce qui ne représente juste 3% que la question de quoi faire de ces attributs.



```
gender               0
age                  0
hypertension         0
heart_disease        0
ever_married         0
work_type            0
Residence_type       0
avg_glucose_level    0
bmi                201
smoking_status       0
stroke               0
dtype: int64
```

Figure 1: Visualisation de la répartition de valeurs *null* dans le dataset

Bien qu'il soit souvent intéressant de favoriser la suppression des données manquantes lorsqu'on a assez de données, dans notre cas nous avons choisis de remplacer par des estimateurs statistiques comme la médiane les objects manquant ayant des valeurs null .

Une fois les données null remplacées par la médiane , nous avons pu effectuer une séparation des données catégoriques et numérique afin de procéder à la normalisation des données numériques,la normalisation reste une procédure simple mais très essentielle pour la conduite de notre projet. Ainsi, pour les données catégorique, nous avons eu recourt à la méthode $OneHotEncoder()$ qui permet d'encoder ces données ce qui augmente leur utilisabilité vis a vis du modèle. Pour Les données numériques elles sont simplement normalisées en soustrayant la variable par le min de la colonne diviser par le max de la colonne soustrayant le min de la colonne

De plus la quantité de données fournies par le dataset est conséquente ce qui nous

permet d'effectuer des set d'entraînement et de test. Pour cela nous avons utilisé une méthode de *sklearn* train_test_split qui nous permettant de réaliser ce découpage plus efficacement.

3.2 Description des données

Dans le cadre de notre projet , les données sont tirées de Kaggle comme mentionné. Un site de compétition en ligne Entre autre il y'a beaucoup de possibilités mais le plus important c'est que les données soient assez réaliste pour mener à terme notre étude dans notre dataset on 'a les variables ou attribut suivant

Table 1: Table de description .

Begin of Table		
Variable	Type	Description
id	Numeric	L'Id du patient
gender	Categorical	Le genre du patient (femme ou homme)
ever_married	Categorical	Cette variable permet de savoir si le client a déjà été marié ou pas
age	Integer	L'âge des patients
hypertension	Categorical	Cette variable décrit si le Client à de l'hypertention ou pas
heart_disease	Integer	Une cardiopathie
work_type	Categorical	Cet Attribut désigne le type de travail du patient
Residence_type	Categorical	Cette variable désigne le type de résidence du patient
avg_glucose_level	Integer	Cette variable designe le taux d'AVG glucose dans le sang du patient
smoking_status	Categorical	Cete variable montre si le client est fumeur ou pas
stroke	Categorical	stroke c'est notre variable à prédire
End of Table		

```
gender               object
age                 float64
hypertension          int64
heart_disease         int64
ever_married         object
work_type            object
Residence_type       object
avg_glucose_level   float64
bmi                 float64
smoking_status       object
stroke                int64
dtype: object
```

Figure 2: Type de variable

3.3 Analyse des données

Les AVC font partie des maladies de l'appareil circulatoire. En tant que tels, ils sont les principales cause de mortalité au monde.Une analyse statistique montre que cette maladie touche le plus les personnes entre 60 à 79 ans .Les enfants sont pas du tout concernées par cette maladie.Chez les jeunes la maladie est peu fréquente On peut voir une illustration ci dessous.

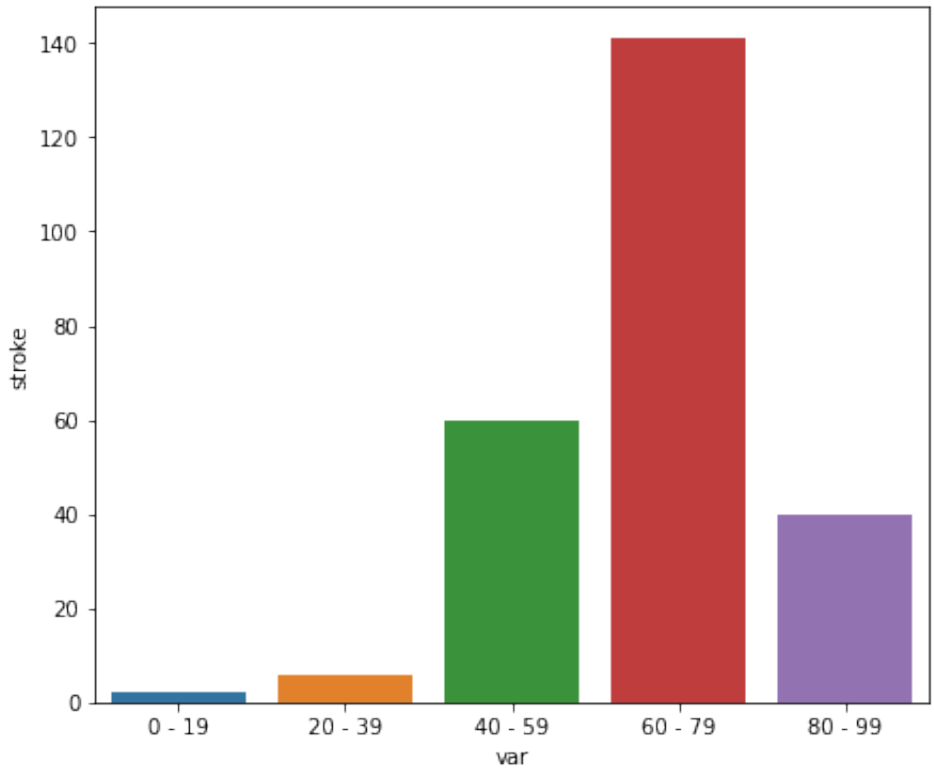


Figure 3: AVC par tranche d'age

Le corps des femmes est différent de celui des hommes; l'AVC les touche différemment,

et les femmes sont plus à risque d'en subir à un certains moments dans leur vie. On peut voir que L'AVC est plus fréquent chez les femmes que les hommes

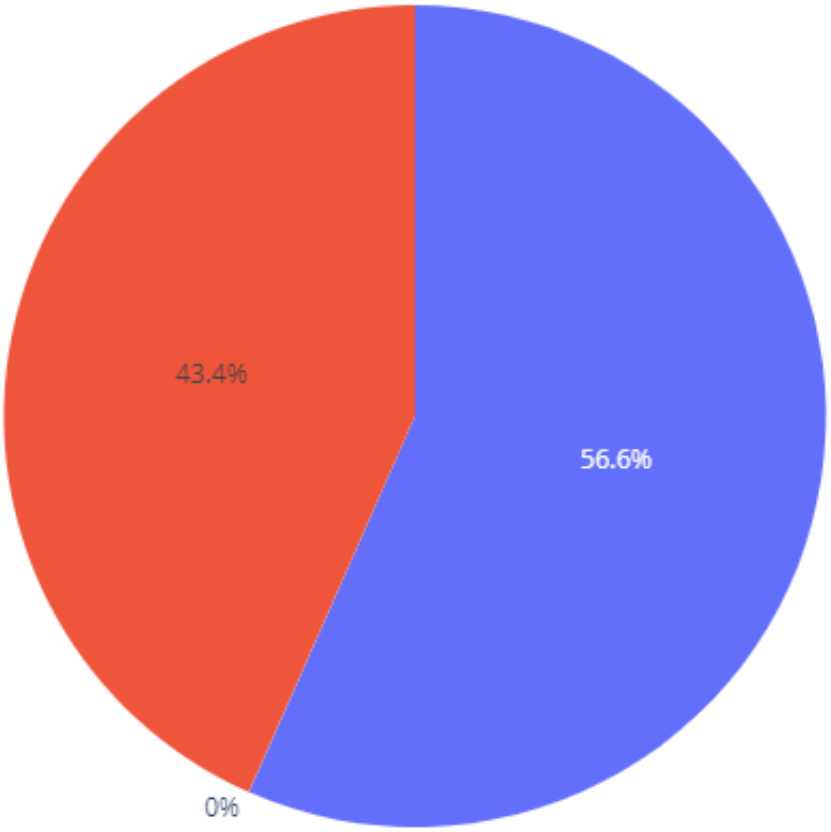


Figure 4: AVC selon l'age

Un point plus intéressant qu'on a constaté avec les statistiques est de voir que les patients ayant L'AVC plus de 80% n'ont vraiment pas d'hypertension artérielle, ou haute pression sanguine.

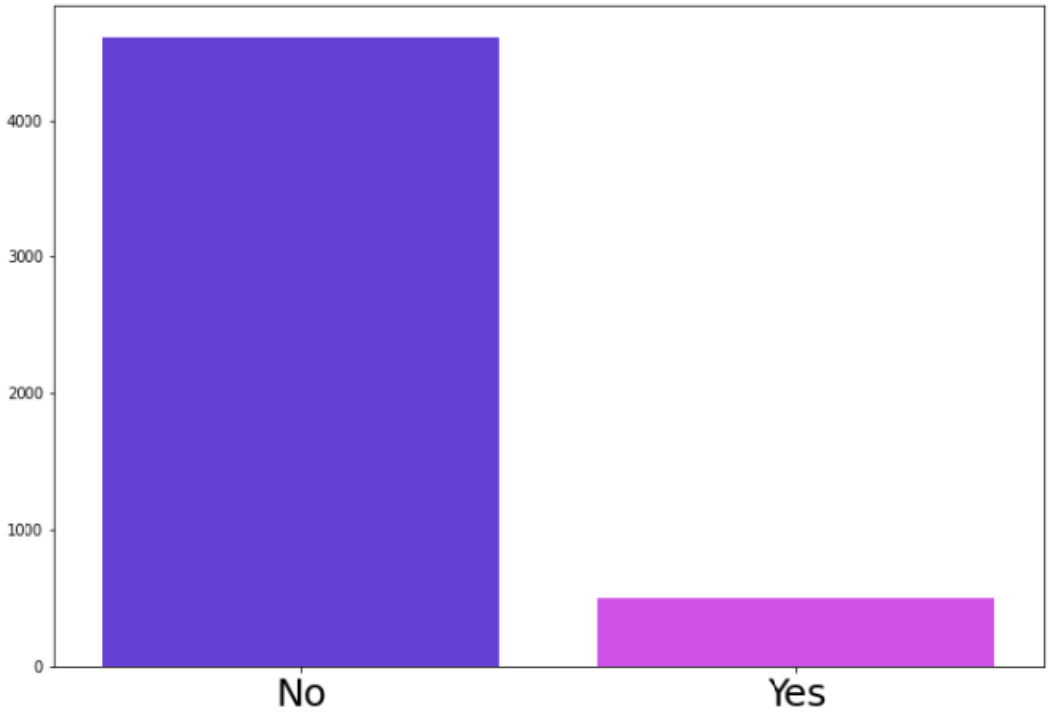


Figure 5: AVC selon l'age

Les patients sont d'autre de milieu Urbain et d'autre de milieu rural notre étude nous montre que l'urbanisation n'a pas d'impact sur la maladie

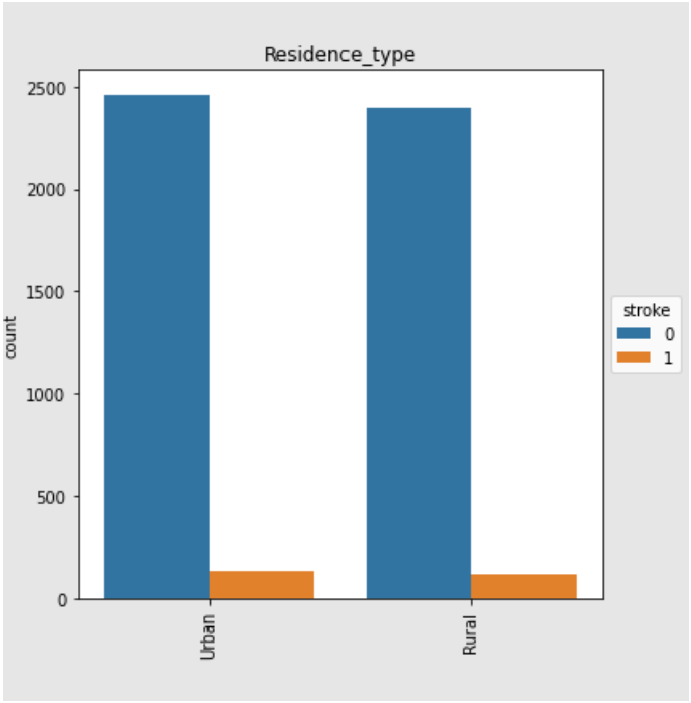


Figure 6: AVC en fonction du milieux

Les études statistiques nous indique que l'AVC est presque néant chez les personnes qui ne sont jamais mariée.

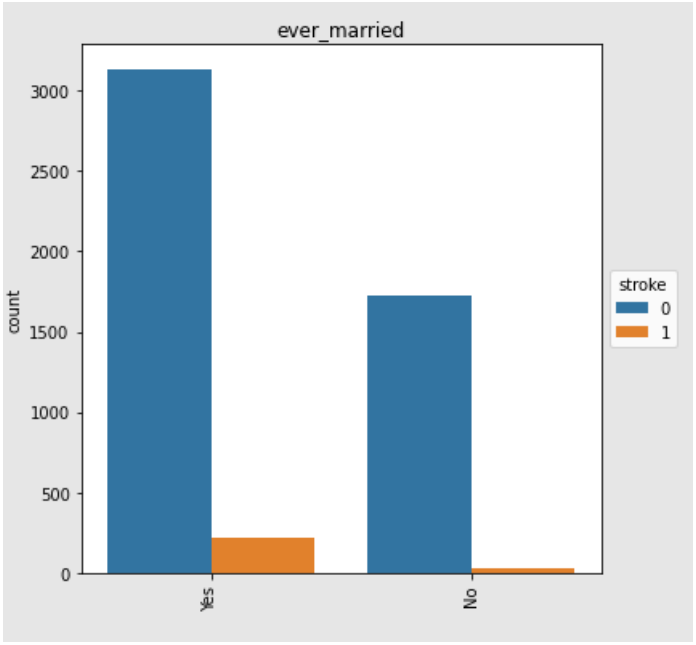


Figure 7: AVC en fonction du milieux

4 Récherhe d'hyperparametres

La quantité de données fournies par le dataset est conséquente ce qui nous permet d'effectuer des set d'entraînement et de test. Pour cela nous avons utilisé une méthode de *sklearn* nous permettant de réaliser ce découpage plus efficacement.Mais une étape très important qu'on a eu à faire On 'a vu que nos données n'était pas équilibré ce qui pourrait biaisé (conduire notre modèle à Overffiter vers la classe majoritaire) les résultats.de ce fait On a utilisé une méthode d'emballencement des données $SMOTE$ qui nous à permit d'équilibrer nos données. On peut voir que les données sont maintenant équilibrer On rappelle que cette méthode à été effectuer sur uniquement le Train.

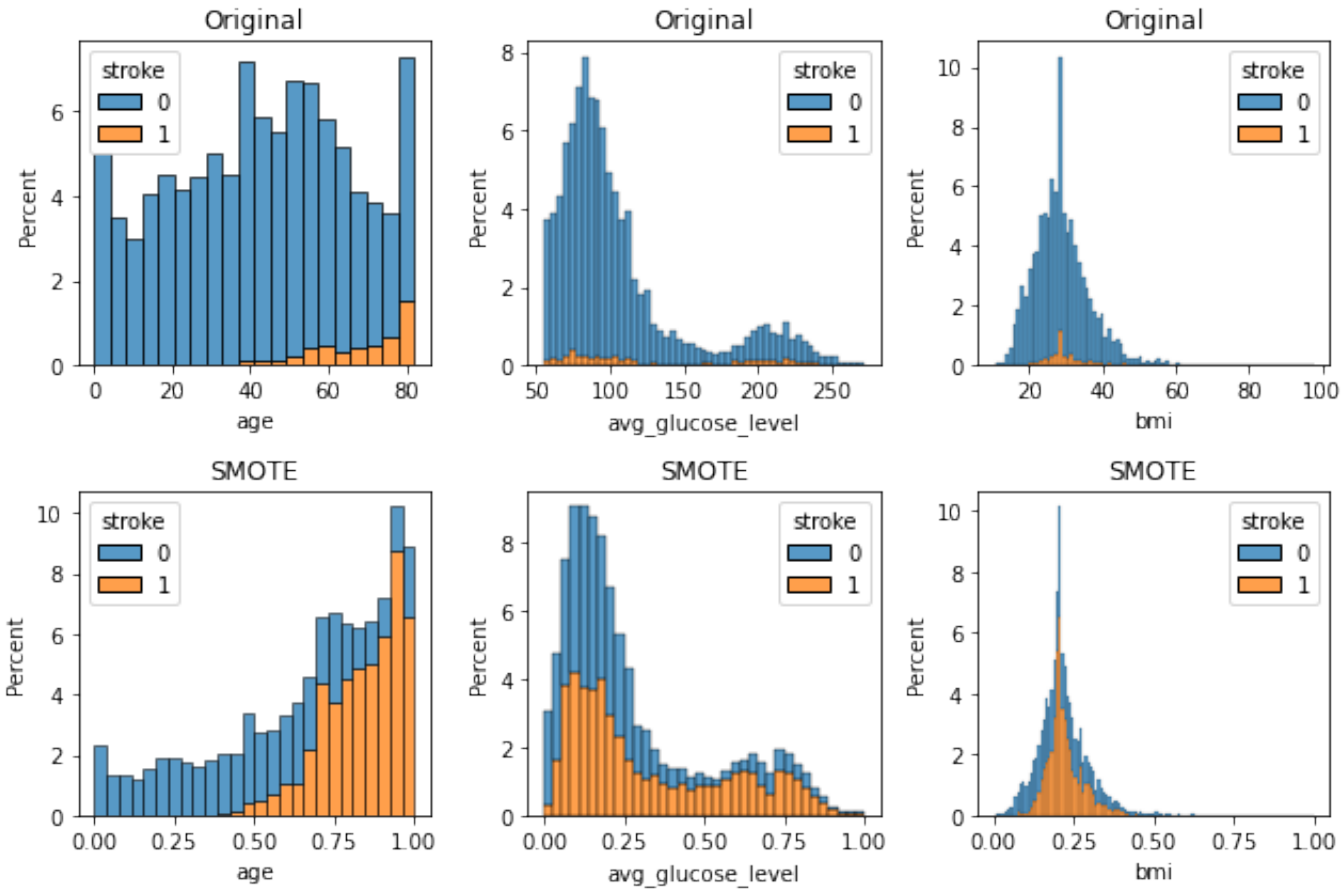


Figure 8: L'Augmentation des données

Maintenant parlons des hyperparamètres Cette partie est l'une des parties clés d'un projet en manchine learning car elle permet d'avoir les bons paramètres pour nos modèles. Dans notre cas On a utilisé une méthode de *Slearn* appelle GridSearch() Cette GridSearch() fonctionne avec le $KFOLD$ On verra juste en bas Pour chacun de nos modèles les meilleures parametres pour notre projet

4.1 Random forest

Compte tenu de la prévalence des maladies cérébrovasculaires, prédire avec précision l'AVC est essentiel pour stratifier les soins de réadaptation qui doivent être administrés, en particulier aux patients ayant la maladie ou pas.Cependant la première méthode qu'on 'a utilisé est le random forest de la bibliothèque *Sklearn* On a fait une recherche d'hyperparamètre et de tester l'efficacité du modèle sur nos données de Test On peut alors voir le résultats du test sur la figure suivante.

```
Using randomized search:
Fitting 5 folds for each of 10 candidates, totalling 50 fits

Best cross val accuracy : 0.9326574980493936
Best estimator:
RandomForestClassifier(max_depth=32, max_features='log2', min_samples_leaf=3,
                       min_samples_split=8, n_jobs=4)

Accuracy validation: 86.064%
```

Figure 9: Résultats Random Forest

On peut clairement voir les meilleures paramètres pour le random Forest et en validant notre modèle sur les données du Test avec ces meilleures paramètres on retrouve aussi L'Accuracy qui est de 86.064%

4.2 K Nearest Neighbors (KNN)

Il est intéressant de tester le modèle KNN sur nos données pour ce faire on à utiliser aussi la bibliothèque *SKlearn* avec les mêmes procédés précèdent on retrouve que l'accuracy avec les meilleures paramètres sur nos données de Test est de 94.7162%

```
Using randomized search:
Fitting 5 folds for each of 10 candidates, totalling 50 fits
Best cross val accuracy : 0.9505301685891748
Best estimator:
KNeighborsClassifier(leaf_size=40, n_jobs=4, n_neighbors=41)

Accuracy validation: 96.822%
best_cross_val_acc : 0.9505301685891748

Best estimator:
KNeighborsClassifier(leaf_size=40, n_jobs=4, n_neighbors=41)
[Estimator parameters]
L Accuracy de la validation: 96.822%
```

Figure 10: Résultats KNN

4.3 Arbres de décision

Pour l'Arbre de decision On peut voir aussi les meilleures paramètres suivants sur la figure

```
Using randomized search:
Fitting 5 folds for each of 10 candidates, totalling 50 fits
best_cross_val_acc : 0.9306895149364094

Best estimator:
DecisionTreeClassifier(criterion='entropy', max_depth=2000, min_samples_leaf=5)
```

Figure 11: Résultats Arbres de decision

et l'accuracy sur le train avec les meilleurs paramètres est de 92.95560%

4.4 SVM

Un SVM a ensuite été entraîné sur ces meilleurs paramètres comme on peut le voir avec les meilleurs paramètres à classifier les vecteurs de caractéristique on obtient alors avec ce modèle un accuracy de : 74.51%

```
Using randomized search:
Fitting 5 folds for each of 10 candidates, totalling 50 fits
Best cross val accuracy : 0.8939101197360468
Best estimator:
SVC(C=100, gamma=0.1, kernel='poly', probability=True)

Accuracy validation: 83.863%
```

Figure 12: Résultats SVM

On 'a testé les meilleures paramètre du SVM/SVC sur nos données de Test et On 'a eu 83.9530%

4.5 Neural Network

Pour ce modèle on peut voir les meilleures paramètres ainsi que l'accuracy avec les meilleures paramètres du modèle.

```
Using randomized search:
Fitting 5 folds for each of 10 candidates, totalling 50 fits

Best cross val accuracy : 0.9505301685891748
Best estimator:
MLPClassifier(alpha=0.01, hidden_layer_sizes=(200,), learning_rate_init=10.0,
              max_iter=800)

Accuracy validation: 96.822%
```

Figure 13: Résultats Neural network

Avec les meilleures paramètres on 'a fait un test avec nos données de Test et on a eu l'accuracy suivante 94.7162%

4.6 Régression Logistique

Pour la régression logistique on peut voir les meilleures paramètres

```
Using randomized search:
Fitting 5 folds for each of 10 candidates, totalling 50 fits
Best cross val accuracy : 0.7922507493089791
Best estimator:
LogisticRegression(C=10, n_jobs=4, solver='newton-cg')

Accuracy validation: 73.594%
```

Figure 14: Résultats Regression Logistique

et en le testant testant avec notre test On 'a eu $76,61\%$

5 Discussion et Comparaison de modèles

Comme on le peut le voir dans notre Étude on 'a testé différente méthode et chacune de ces méthodes a ses avantages et ses inconvénients. L'algorithme du Knn est simple et facile à mettre en œuvre. Il n'est pas nécessaire de créer un modèle de régler plusieurs paramètres ou de formuler des hypothèses supplémentaires. L'algorithme est polyvalent. Il peut être utilisé pour la classification ou la régression. L'algorithme devient beaucoup plus lent à mesure que le nombre d'observation et de variables indépendantes augmente.

Pour le perceptron Il s'agit d'un algorithme pour l'apprentissage supervisé de classificateurs binaires. Le Perceptron joue un rôle essentiel dans les projets de Machine Learning. Il est massivement utilisé pour classifier les données, ou en guise d'algorithme permettant de simplifier ou de superviser les capacités d'apprentissage de classificateurs binaires On 'a aussi utilisé les arbres de décision (Un seul arbre de décision est un prédicteur faible, mais il est relativement rapide à construire. Un plus grand nombre d'arbres permet d'obtenir un modèle plus robuste et d'éviter les sur-ajustements. Cependant, plus vous avez d'arbres, plus le processus est lent. Chaque arbre de la forêt doit être généré, traité et analysé.

En outre, plus le nombre de caractéristiques est élevé, plus le processus est lent (il peut parfois prendre des heures, voire des jours) ; la réduction du nombre de caractéristiques peut accélérer considérablement le processus.c'est pourquoi le Random Forest est adéquate. les modèles Les Réseaux Neurones ,de Régression logistique ont été tester aussi sur nos données mais à travers les accuracy.

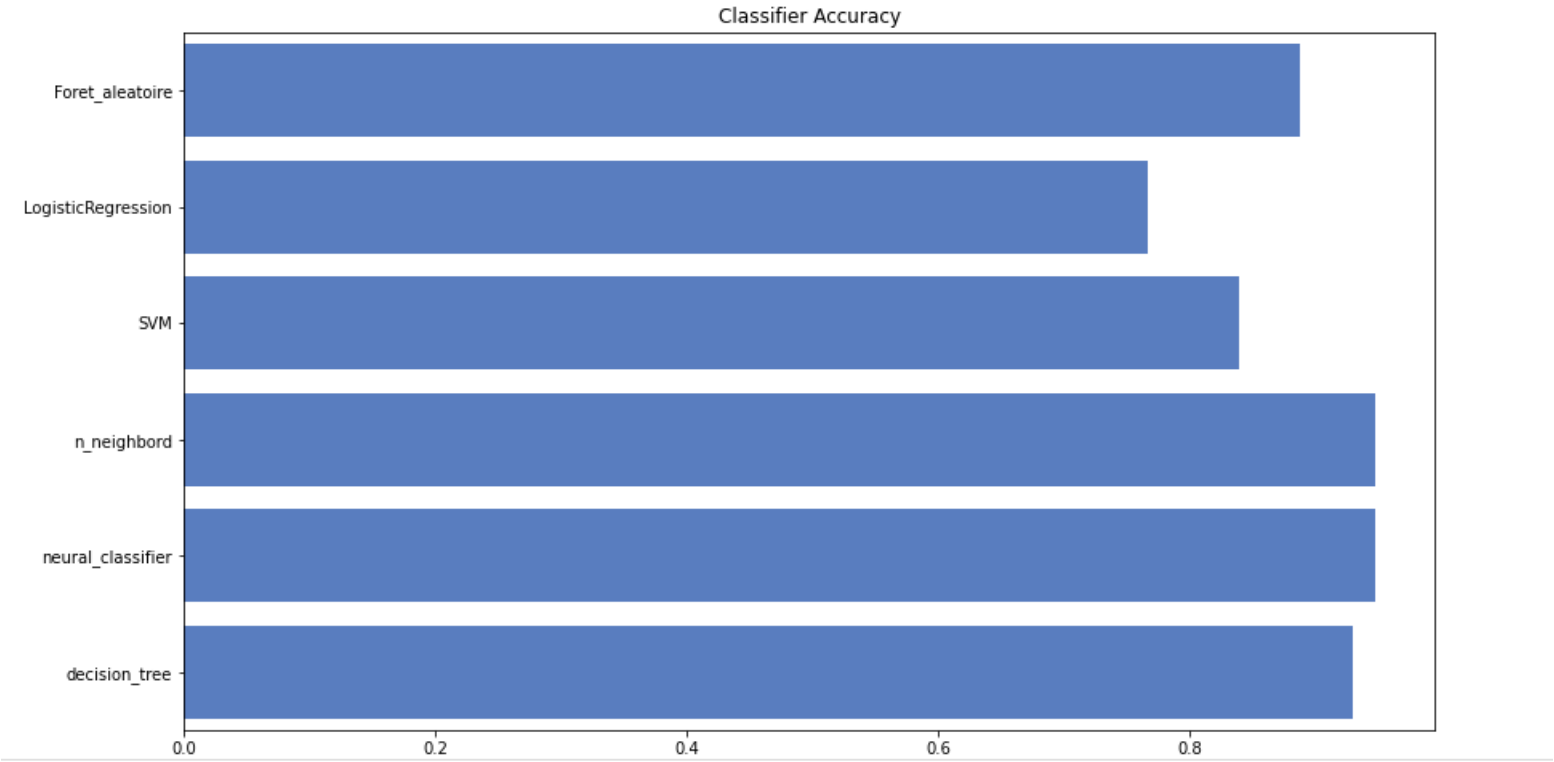


Figure 15: diagramme des accuracy

on peut conclure que les deux meilleures modèles pour notre projet est le Réseaux neurones et le Knn leurs meilleures paramètres sont affichées en haut.

6 Conclusion

Au terme de notre étude, il revient de constater que la performance des algorithmes du machine learning diffèrent selon les données. Pour ce projet, qu'est la prédiction des attaques cardiaques, nous constatons que le classifier des plus proches voisins et celui de Neural Network sont les plus adaptés à ce type de problème.

References

[1] Inconnu. Article sur l'avc par gouvernement du quebec. https://www.msss.gouv.qc.ca/professionnels/traumatismes-et-traumatologie/avc.

[2] Kaggle. Stroke prediction. https://https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset.

[3] Sangare Mahamadou Maman Souley Aicha. Dépot git du projet. https://github.com/MamanSouleyAicha/Projet_ift603.

[4] Sklearn. documentation sur sklearn. https://scikit-learn.org/.