RAPPORT DE PROJET DE MACHINE LEARNING

Estimation de niveaux d'obésité en fonction des habitudes alimentaires et de condition physique

Réalisé par :

Pierjos Francis COLERE MBOUKOU

Encadré par :

Ikram CHAIRI

Hasnae ZEROUAOUI

Année Universitaire : 2020-2021

REMERCIEMENTS

La réalisation de ce projet a été possible grâce au concours de plusieurs personnes à qui nous voudrions témoigner toute notre gratitude. Nous voudrions, tout d'abord, adresser toute notre reconnaissance aux Professeurs Ikram CHAIRI et Hasnae ZEROUAOUI, pour leur disponibilité et surtout leurs judicieux conseils, qui ont exponentiellement contribué à alimenter notre réflexion. Elles ont été toujours à l'écoute de nos nombreuses questions. Nous tenons à remercier toute personne qui lira attentivement de notre rapport ainsi que pour les remarques qu'elle nous adressera afin d'améliorer notre travail. Nous désirions aussi remercier toute l'équipe pédagogique du département Al Khawarizmi de l'Université Mohammed VI Polytechnique qui mettent à notre disposition leurs expériences et leurs compétences et qui nous ont fourni les outils nécessaires à la réussite de ce projet. Nous voudrions enfin exprimer notre reconnaissance envers les amis et collègues qui nous ont apporté, de près ou de loin, leur soutien moral et intellectuel tout au long de ce projet.

RÉSUMÉ

Le présent rapport synthétise le travail effectué dans le cadre de notre projet de Machine Learning. Il vise à estimer les niveaux d'obésité d’un individu en fonction de ses habitudes alimentaires et de sa condition physique. L’ensemble de données utilisé provient des individus du Mexique, du Pérou et de la Colombie.

Ces données sont utilisées, dans ce projet, pour générer des outils de calcul intelligents afin d’identifier le niveau d'obésité d'un individu et de construire des systèmes (algorithmes) qui surveillent les niveaux d'obésité. Nous avons appliqué l’apprentissage supervisé, plus précisément la classification, afin de prédire ces niveaux. KNN, SVM, RandomForest et Réseau de neurones (ANN) sont les algorithmes utilisés pour développer une application Web prédisant ces niveaux d’obésité.

Ce document définit le projet avec son contexte et décrit les étapes nécessaires à sa réalisation.

Mots clés:

Donnée, Machine Learning, Apprentissage supervisé, Classification, Prédiction, KNN, SVM, RandomForest, Réseau de neurones, Application Web.

ABSTRACT

This report summarizes the work done as part of our Machine Learning project. It aims to estimate an individual's obesity levels based on their eating habits and physical condition. The data set used comes from individuals in Mexico, Peru and Colombia.

These data are used, in this project, to generate intelligent calculation tools to identify an individual's level of obesity and to build systems (algorithms) that monitor obesity levels. We applied supervised learning, specifically classification, to predict these levels. KNN, SVM, RandomForest and Neural Network (ANN) are the algorithms used to develop a web application predicting these levels of obesity.

This document defines the project with its context and describes the steps necessary for its realization.

Keywords:

Data, Machine Learning, Supervised Learning, Classification, Prediction, KNN, SVM, RandomForest, Neural Network, Web Application.

INTRODUCTION

L’obésité correspond à un excès de masse grasse et à une modification du tissu adipeux (contenant des cellules graisseuses), entraînant des inconvénients pour la santé et pouvant même réduire l’espérance de vie. Ses causes sont complexes. Elle résulte de plusieurs facteurs (alimentaires, génétiques épigénétiques et environnementaux) impliqués dans le développement et la progression de cette maladie chronique. Ainsi, accéder à une meilleure compréhension des causes et des mécanismes biologiques conduisant à l’obésité est aujourd’hui un des plus grands enjeux de la recherche. Comme toutes les maladies chroniques, l’obésité devient en effet irréversible lorsqu’elle est installée : prévenir son développement est donc primordial si l’on veut enrayer l’épidémie mondiale.

C'est dans cette perspective, nous allons comprendre les différentes causes de cette maladie et implémenter un système intelligent (basé sur le Machine Learning) au travers de données récoltées au Mexique, Pérou et Colombie.

Pour mener à bien ce projet, dans un premier temps nous comprendrons de quoi sont constituées les données utilisées. Ensuite, cette compréhension de données nous permettra de faire une analyse profonde (Inférence) de données afin de tirer quelques hypothèses sur les causes de cette maladie (Obésité).

Après avoir analyser les données, il serait judicieux de traiter (pré-traitement) ces dernières afin de les modéliser correctement. Puis nous allons optimiser les modèles retenus afin d’augmenter les performances, la précision de ces modèles.

Finalement, nous allons développer une application Web qui permettra aux utilisateurs de connaître leur niveau de santé (normal, obèse ou en insuffisance de poids).

Nous utiliserons, tout au long de ce projet, Python comme langage de programmation.

TABLE DE MATIÈRES

CHAPITRE 1: COMPRÉHENSION DES DONNÉES

L’ensemble données utilisé est issu de UCI Machine Learning Repository et contient 17 variables (caractéristiques) et 2111 enregistrements, les enregistrements sont étiquetés avec la variable de classe NObeyesdad (Niveau d’obésité), qui permet la classification des données en utilisant les valeurs Insufficient Weight (poids insuffisant), Normal Weight (poids normal), Overweight Level I (surpoids niveau I), Overweight Level II (surpoids niveau II), Obesity Type I (obésité de type I), Obesity Type II (obésité de type II) et Obesity Type III (obésité de type III).
Les variables liées aux habitudes alimentaires sont:

- ✔ Consommation fréquente d'aliments riches en calories (FAVC),
- ✔ Fréquence de consommation de légumes (FCVC),
- ✔ Nombre de repas principaux (NCP),
- ✔ Consommation d'aliments entre les repas (CAEC),
- ✔ Consommation d'eau quotidienne (CH20),
- ✔ Consommation d'alcool (CALC).

Les variables liées à la condition physique sont:

- ✔ Surveillance de la consommation de calories (SCC),
- ✔ Fréquence de l'activité physique (FAF),
- ✔ Temps d'utilisation des appareils technologiques (TUE),
- ✔ Moyen de transport souvent utilisé (MTRANS)

Les autres variables obtenues : le sexe (Gender), l'âge (Age), la taille (Height) et le poids (Weight), SMOKE associée au fumeur si sa valeur est 1 et non-fumeur dans le cas contraire, family_history_with_overweight égal à 1 signifie la personne a un antécédant en surpoids et 0 la personne n’a pas un antécédent en surpoids.

CHAPITRE 2 : ANALYSE DE DONNÉES

I – ANALYSE DE FORME

1 - Dimension de données

L’ensemble données contient 17 variables (caractéristiques) et 2111 enregistrements.

2 - Analyse de types

Sur l'ensemble de données, nous avons neuf (9) variables qualitatives et 8 variables quantitatives comme l’illustre l’image ci-dessous

```
In [4]: data.dtypes.value_counts()
Out[4]:
object      9
float64     8
dtype: int64
```

3 - Analyse de données manquantes

Nous constatons que la base de données ou l’ensemble de données ne contient pas de données manquantes.

```
Gender                                0.0
CALC                                  0.0
TUE                                   0.0
FAF                                   0.0
SCC                                   0.0
CH2O                                  0.0
SMOKE                                 0.0
MTRANS                                0.0
CAEC                                  0.0
FCVC                                  0.0
FAVC                                  0.0
family_history_with_overweight        0.0
Weight                                0.0
Height                                0.0
Age                                   0.0
NCP                                   0.0
NObeyesdad                            0.0
dtype: float64
```

4 – Analyse descriptive

D'après l'étude descriptive ci-dessous, les valeurs des variables ne sont pas sur la même échelle. Ce qui nécessiterait la normalisation des données.

Indice	Age	Height	Weight	FCVC	NCP	CH2O	FAF	TUE
count	2111	2111	2111	2111	2111	2111	2111	2111
mean	24.3126	1.70168	86.5861	2.41904	2.68563	2.00801	1.0103	0.657866
std	6.34597	0.0933048	26.1912	0.533927	0.778039	0.612953	0.850592	0.608927
min	14	1.45	39	1	1	1	0	0
25%	19.9472	1.63	65.4733	2	2.65874	1.58481	0.124505	0
50%	22.7779	1.7005	83	2.3855	3	2	1	0.62535
75%	26	1.76846	107.431	3	3	2.47742	1.66668	1
max	61	1.98	173	3	4	3	3	2

II – ANALYSE DE FOND

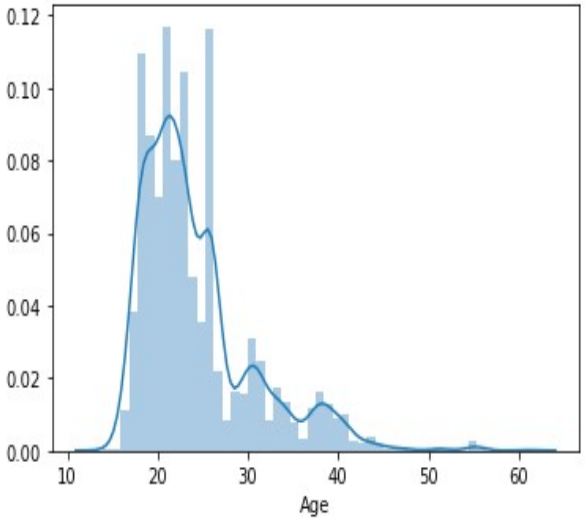
1 - Visualisation de niveaux d'obésité (NObeyesdad)

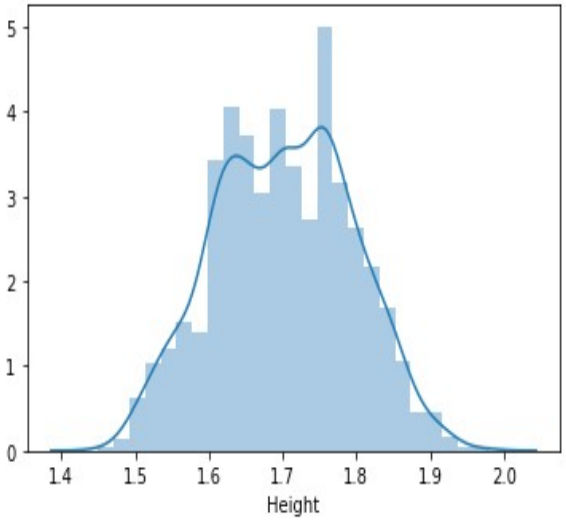
Obesity_Type_I	0.166272
Obesity_Type_III	0.153482
Obesity_Type_II	0.140692
Overweight_Level_II	0.137376
Overweight_Level_I	0.137376
Normal_Weight	0.135955
Insufficient_Weight	0.128849

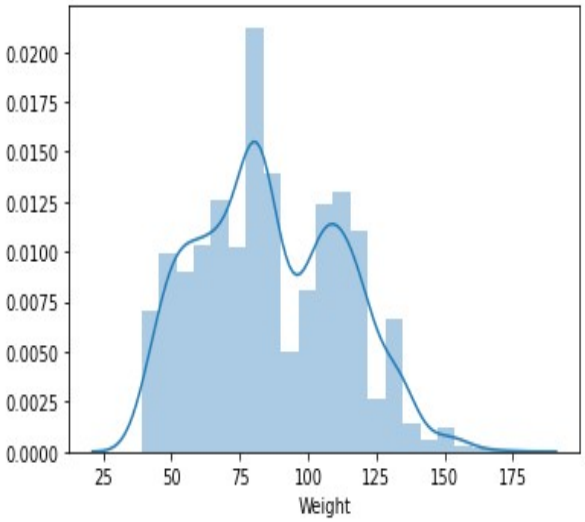
On remarque les classes à prédire sont presque équilibrées. Il n' y a donc pas le problème de déséquilibre de classe (unbalanced data).

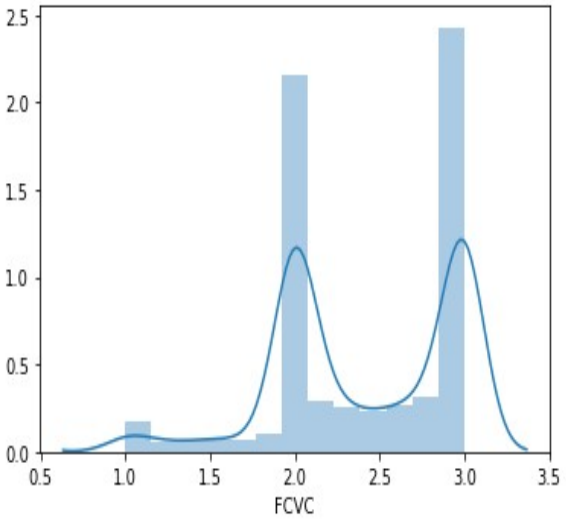
2 – Analyse des variables quantitatives

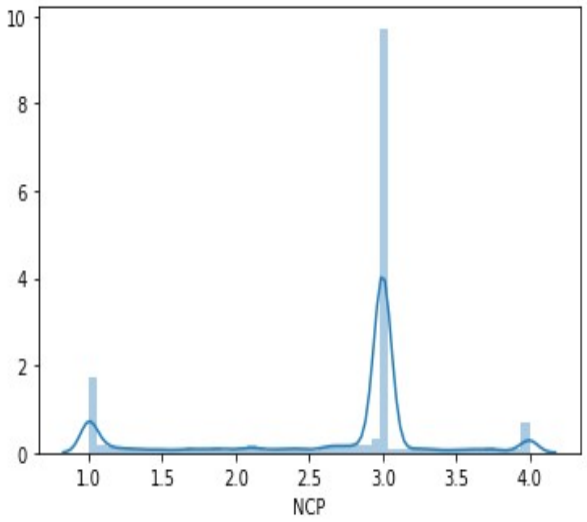
Nous avons représenté les histogrammes des variables quantitatives afin de voir leur distribution (densité). Ainsi, ci-dessous les résultats obtenus :

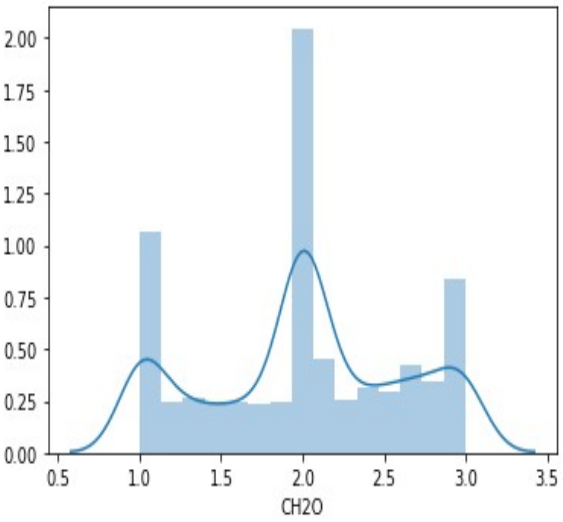

0.12
0.10
0.08
0.06
0.04
0.02
0.00
10
20
30
40
50
60
Age

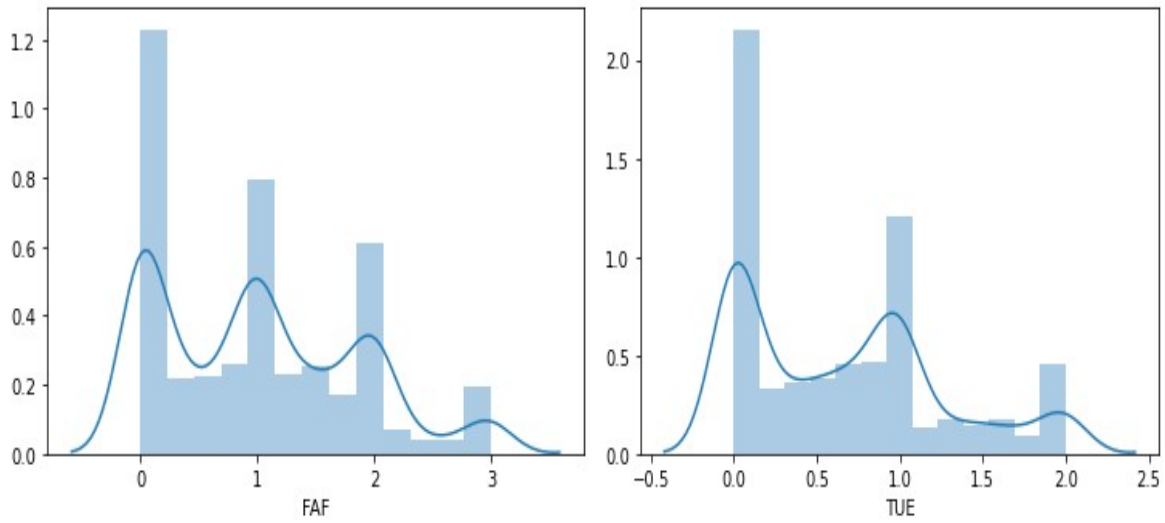

5
4
3
2
1
0
1.4
1.5
1.6
1.7
1.8
1.9
2.0
Height


0.0200
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
25
50
75
100
125
150
175
Weight


2.5
2.0
1.5
1.0
0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
FCVC


10
8
6
4
2
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
NCP


2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.5
1.0
1.5
2.0
2.5
3.0
3.5
CH2O

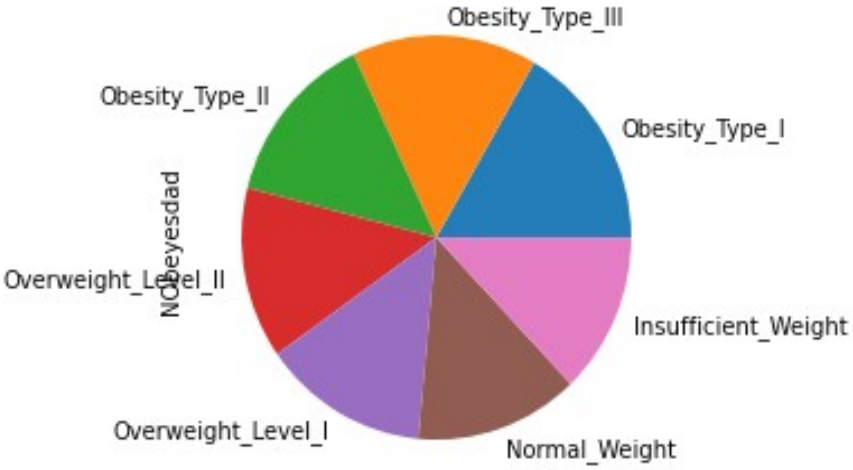


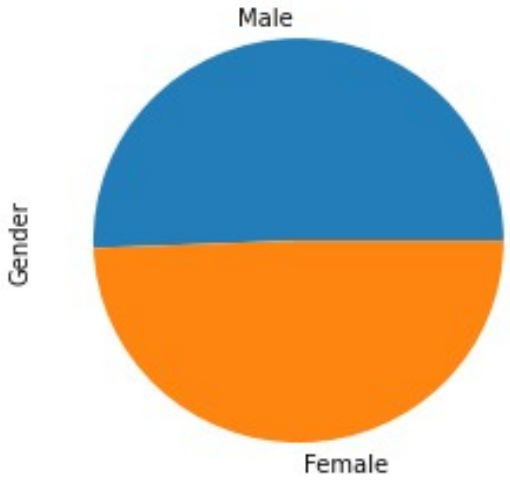
Hypothèses :

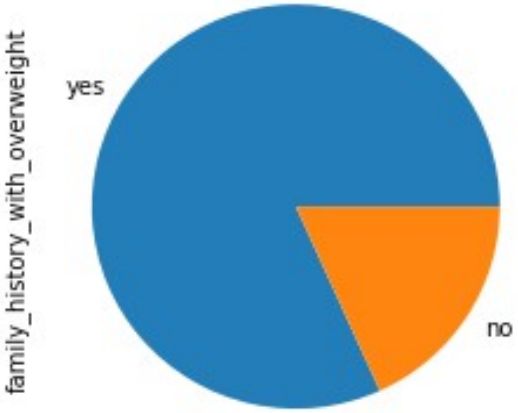
L'âge suit presque la loi gaussienne dont la moyenne est entre 20 et 25. De même pour la taille (1.70 < moyenne < 1.8), NCP (2.5 <= moyenne <= 3.0). Il est difficile d'interpréter les autres variables.

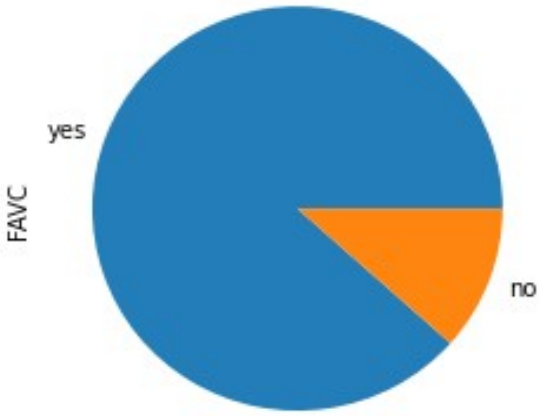
3 – Analyse de variables qualitatives

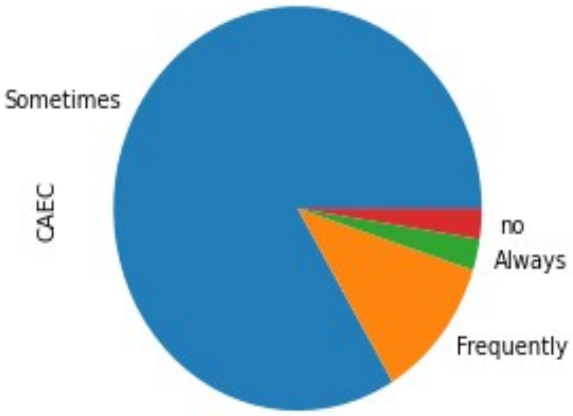
D’après les diagrammes circulaires de ces variables, on constate que le genre (Gender) et le niveau d’obésité (NObeyesdad) sont presque équilibrés. Mais les autres classes ne le sont pas. Les figures ci-dessous nous permet de mieux comprendre ce que nous venons de dire. Par exemple la plupart, soit 98%, des personnes de l’ensemble de données sont non-fumeurs (SMOKE), ont de parents qui sont ou ayant été en surpoids (family_history_with_overweight).

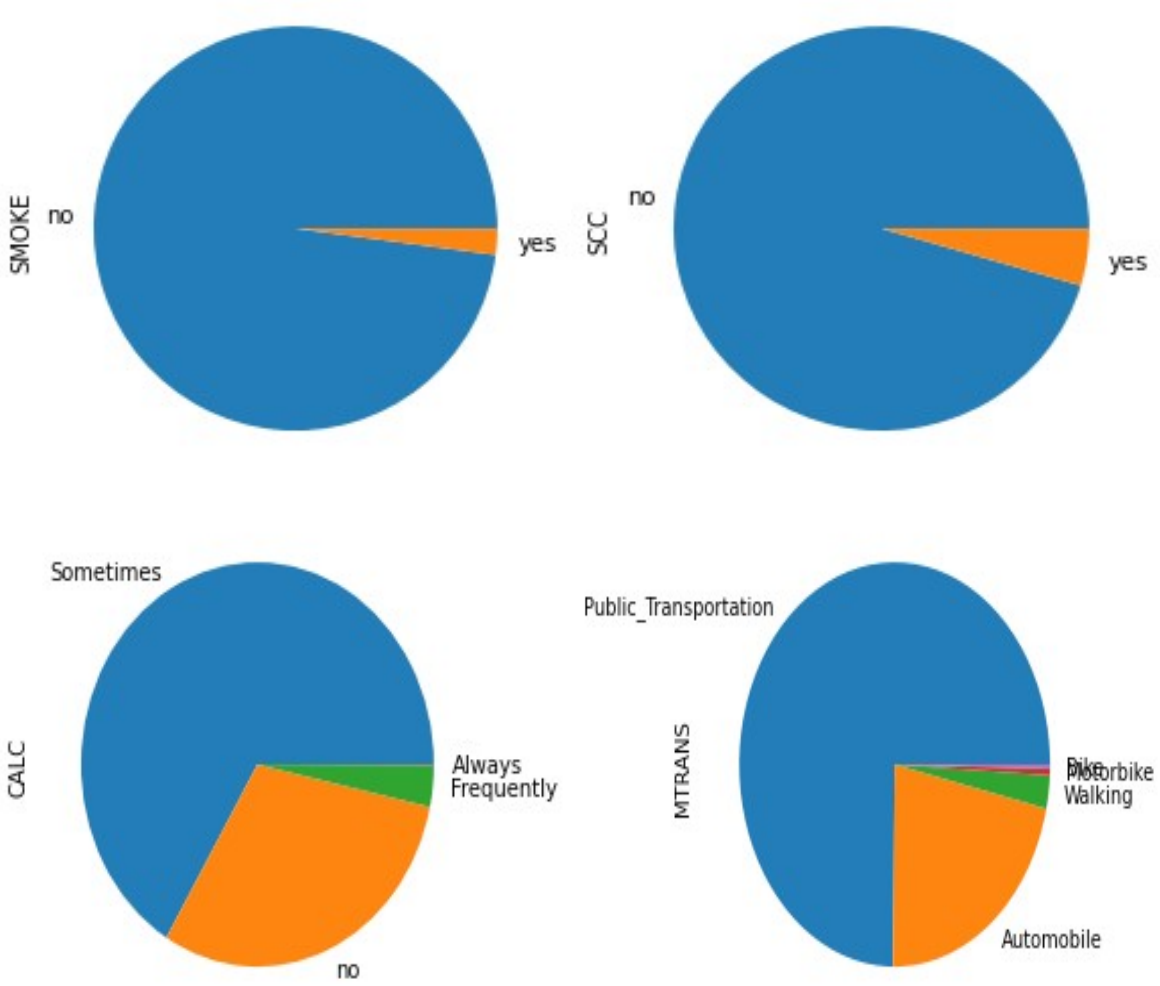

Obesity_Type_III
Obesity_Type_II
Obesity_Type_I
NObeyesdad
Overweight_Level_II
Insufficient_Weight
Overweight_Level_I
Normal_Weight


Male
Gender
Female


family_history_with_overweight
yes
no


FAVC
yes
no


CAEC
Sometimes
no
Always
Frequently

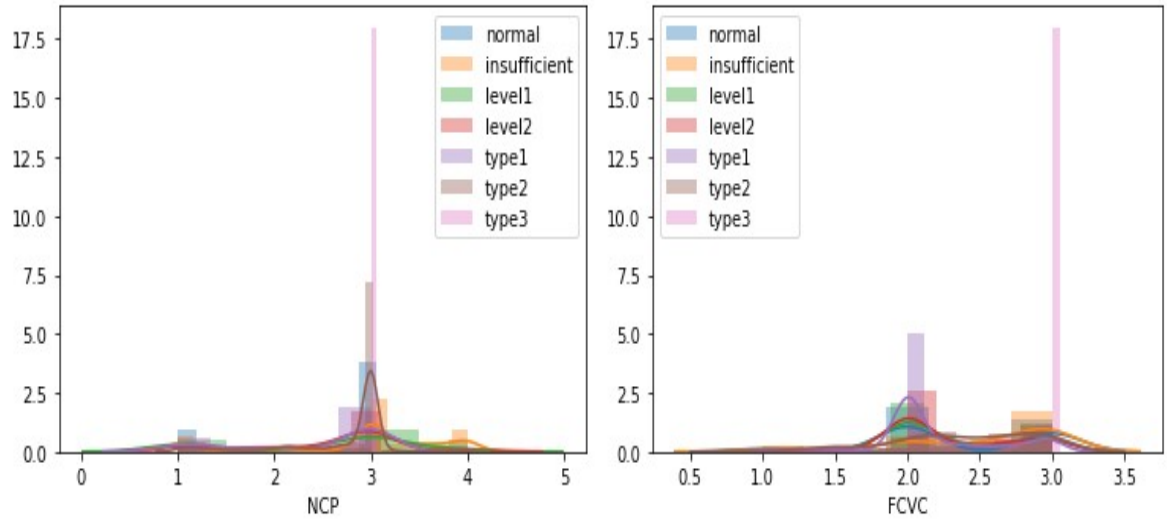


Nous allons ainsi voir les relations existantes entre le niveau d'obésité et les variables quantitatives et qualitatives. Pour ce faire, nous allons créer de sous-ensembles de données suivants :

```
normal_df = data[data['NObeyesdad'] == 'Normal_Weight']
insufficient_df = data[data['NObeyesdad'] == 'Insufficient_Weight']
level1_df = data[data['NObeyesdad'] == 'Overweight_Level_I']
level2_df = data[data['NObeyesdad'] == 'Overweight_Level_II']
type1_df = data[data['NObeyesdad'] == 'Obesity_Type_I']
type2_df = data[data['NObeyesdad'] == 'Obesity_Type_II']
type3_df = data[data['NObeyesdad'] == 'Obesity_Type_III']
```

Concernant l'interprétation, par exemple, le sous-ensemble normal_df contient seulement l'ensemble de données de personnes normales c'est-à-dire ne souffrant pas de surpoids, d'insuffisance de poids ou d'obésité.

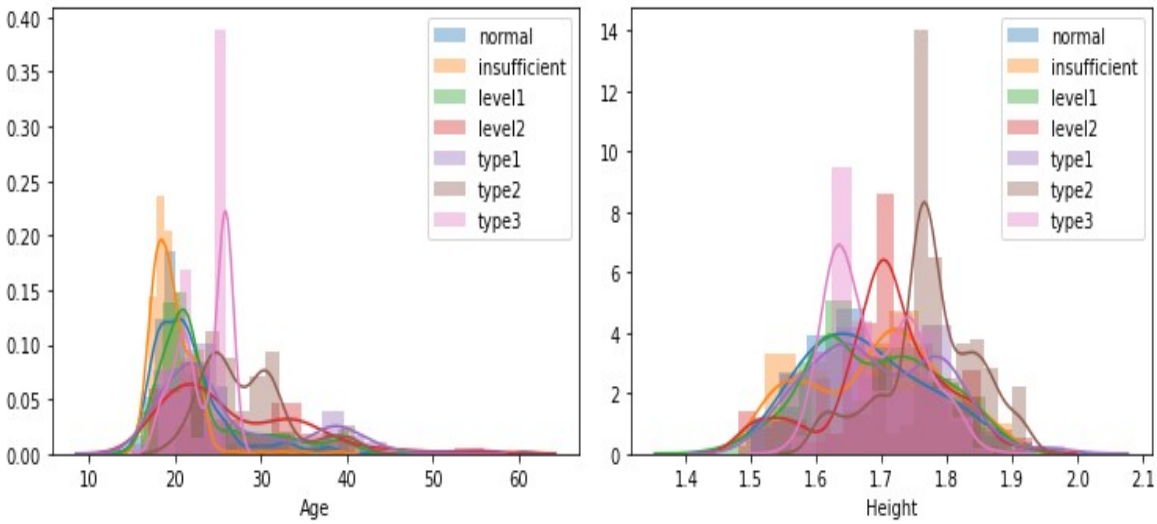
4 - Relations entre NObeyesdad et variables quantitatives

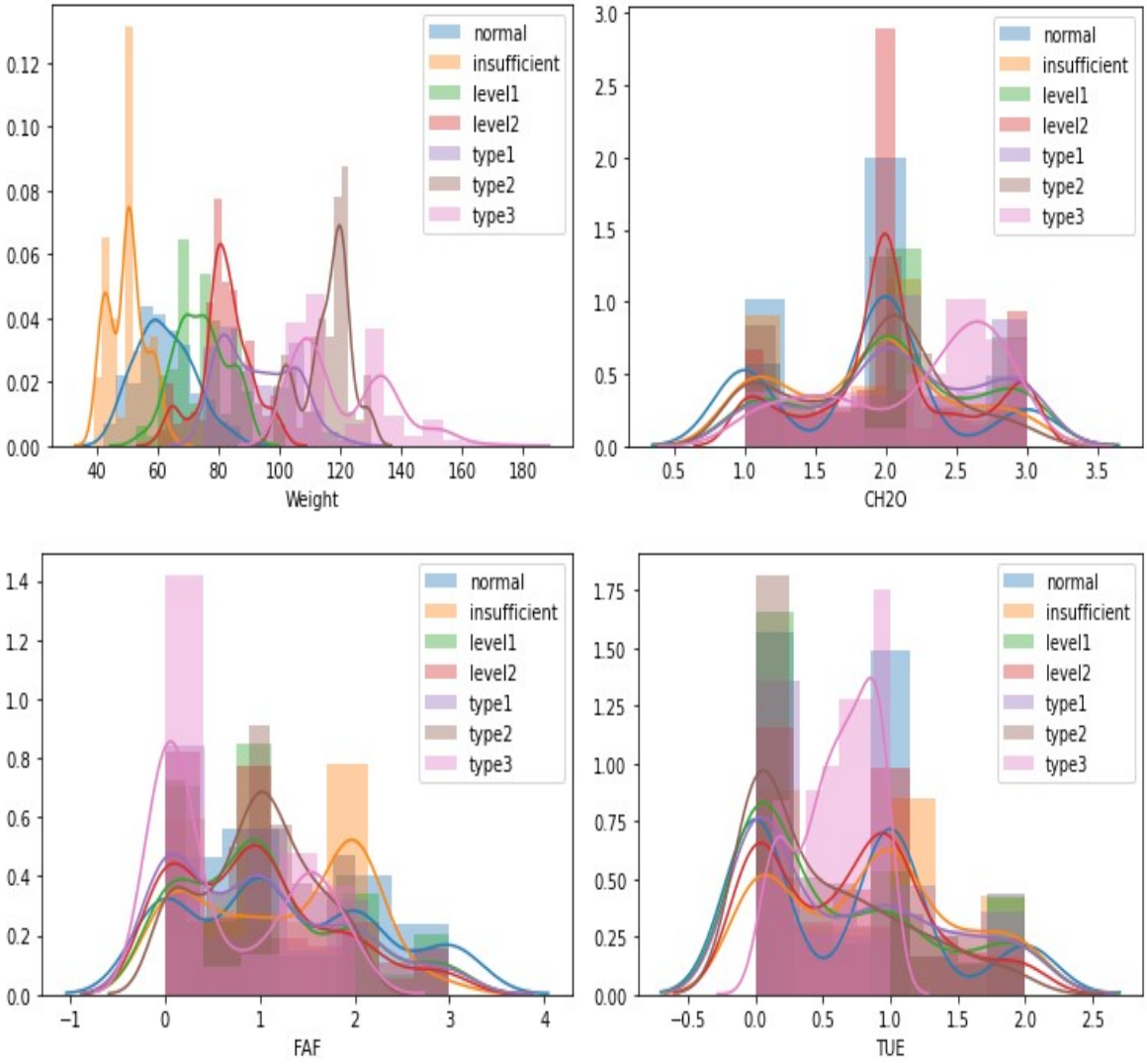


En considération la variable NCP tous les types de la variable NObeyesdad suivent presque la même loi (qui semble la loi normale de moyenne comprise entre 2.5 et 3). De même pour FCVC, les types semblent suivre la même loi normale.

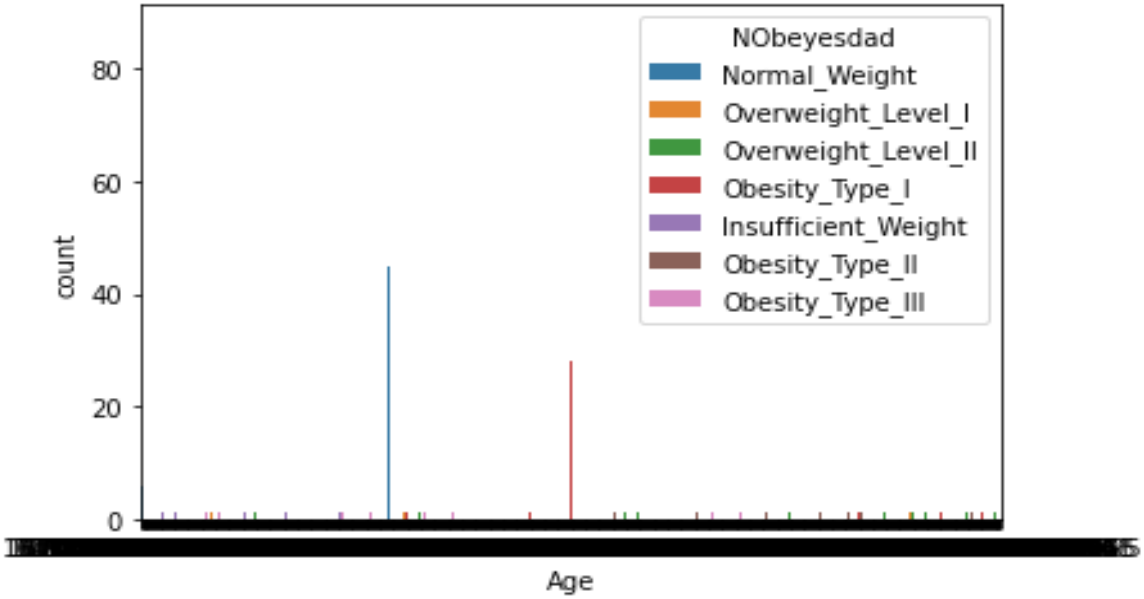
En conclusion, ces différents types sont identiquement distribués pour les variables NCP et FCVC.

Pour les autres variables, il reste toujours difficile d'interpréter les résultats. Elles semblent au moins suivre des distributions normales comme le montrent les figures ci-dessous :




normal
insufficient
level1
level2
type1
type2
type3
Weight
CH2O
FAF
TUE

Cas particulier : Age vs NObeyesdad



Par rapport à l'âge, les personnes âgées de 21 ans sont, en majorité, normales. Les personnes âgées de plus de 21 ans ont souvent de problèmes de surpoids et d'obésité.

De plus, la classe normale et l'obésité de type I dominent (sont en majorité) par rapport aux autres dont les probabilités sont presque nulles. On peut le confirmer en calculant la moyenne de chaque sous-ensemble résumé sous la figure ci-après.

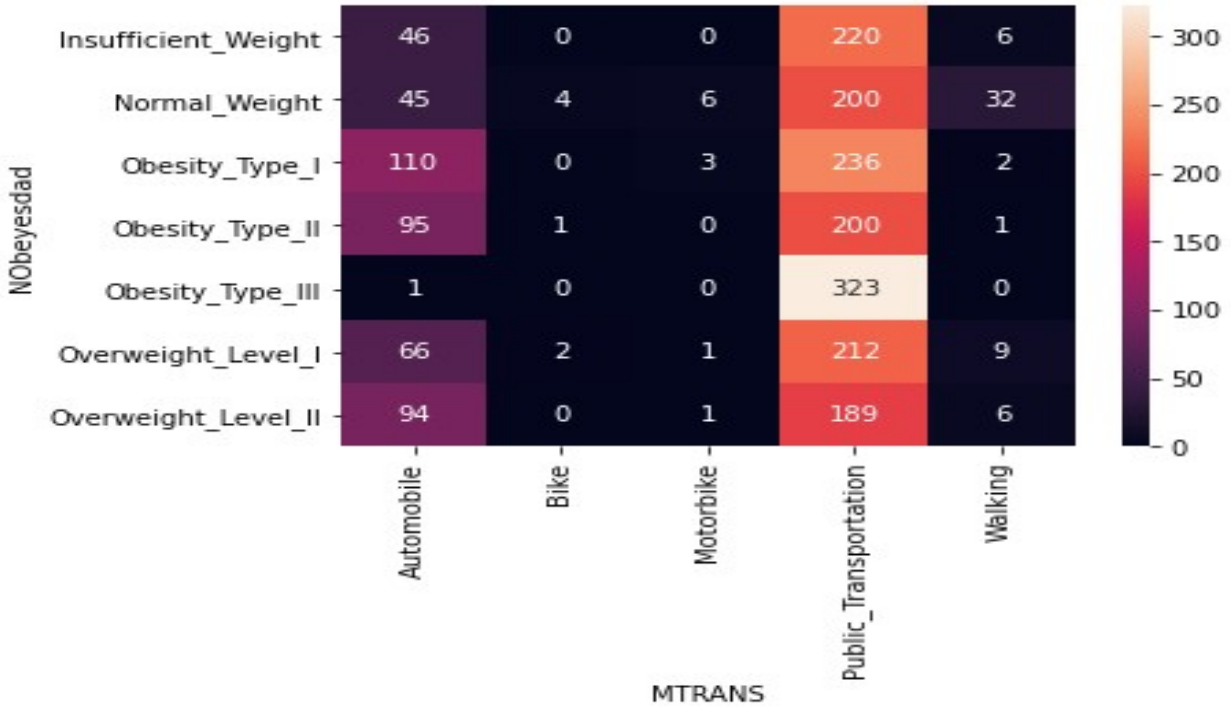
Normal	21.7387
Insuffisance	19.7832
Niveau 1	23.4177
Niveau 2	26.997
Type 1	25.8849
Type 2	28.2338
Type 3	23.4956

Types d’obésité et leur moyenne

5 - Relation entre NObeyesdad et les variables qualitatives

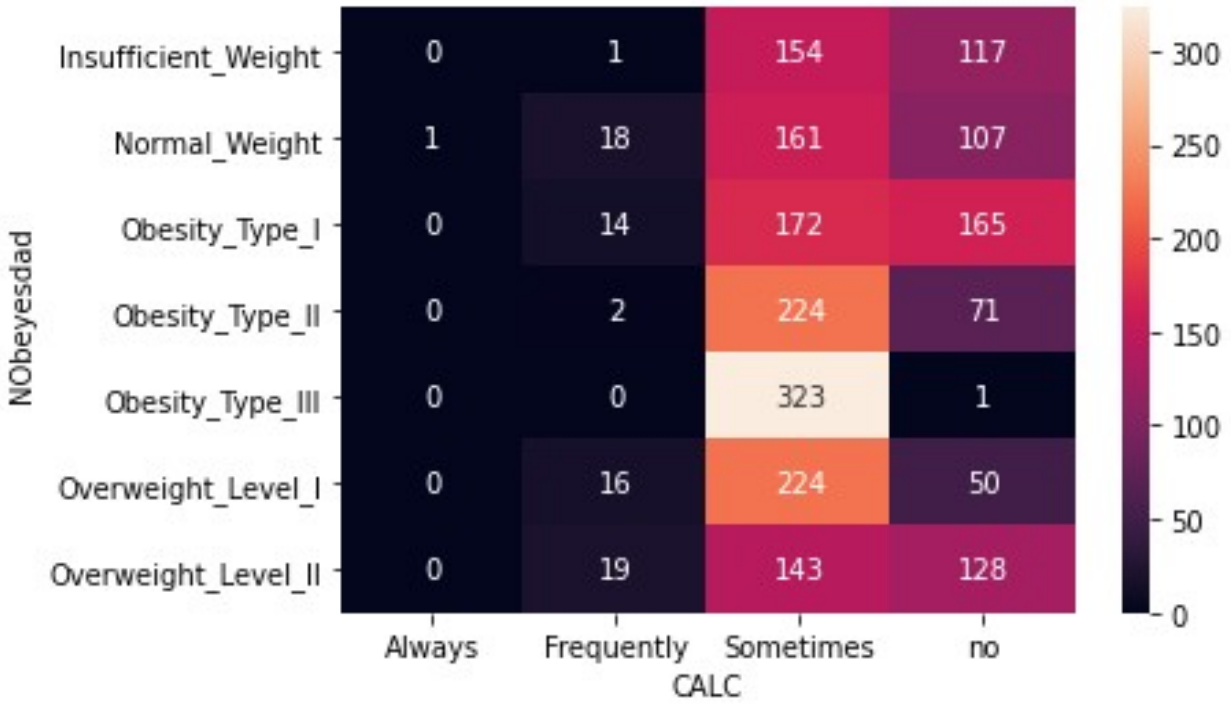
Dans cette partie, nous allons représenter les tableaux de contingence entre les variables catégorielles et le niveau d’obésité.

5-1- MTRANS vs NObeyesdad



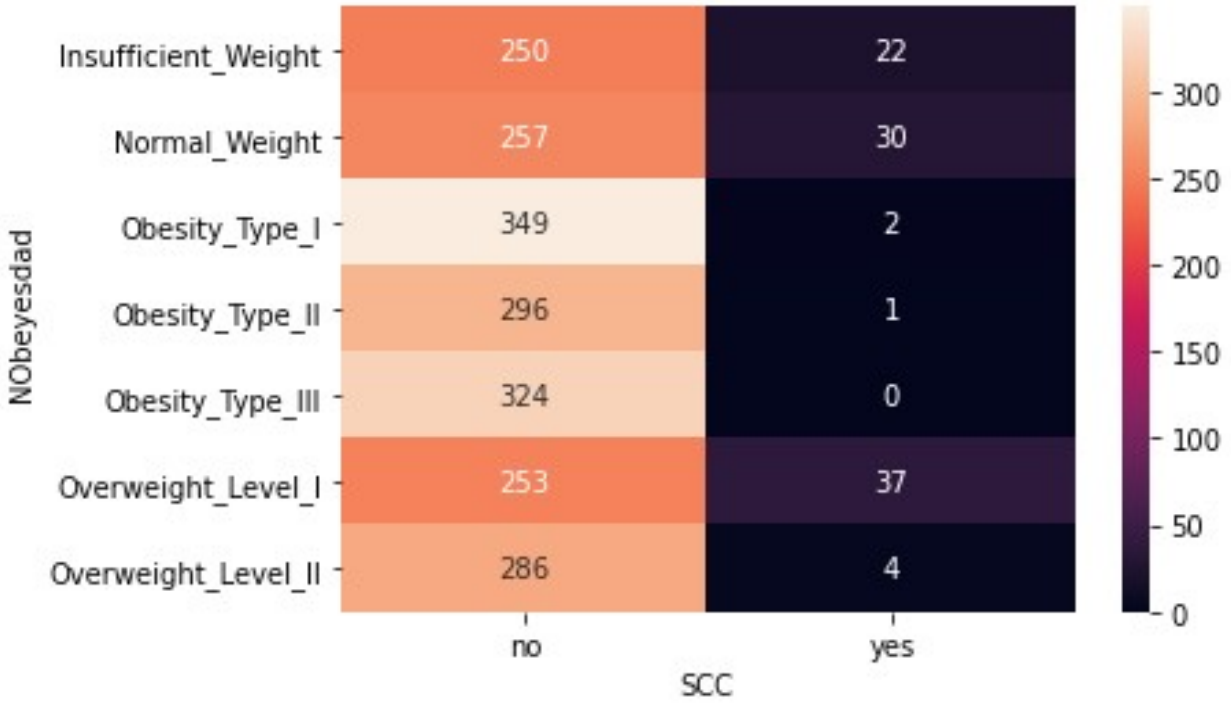
Les moyens de transport agissent sur l‘état de santé d‘une personne. En d'autres termes, ils influencent le niveau d'obésité. En effet, plus la personne utilise les automobiles ou les transports publics plus elle a de chance d'avoir les problèmes de surpoids et d'obésité et inversement.

5-2- CALC vs NObeyesdad



On remarque, le risque de surpoids est plus élevé chez les personnes consommant parfois de l'alcool qu'aux personnes qui en consomment régulièrement. Notre hypothèse pourrait être validée par cet article Section Méconnaissance.

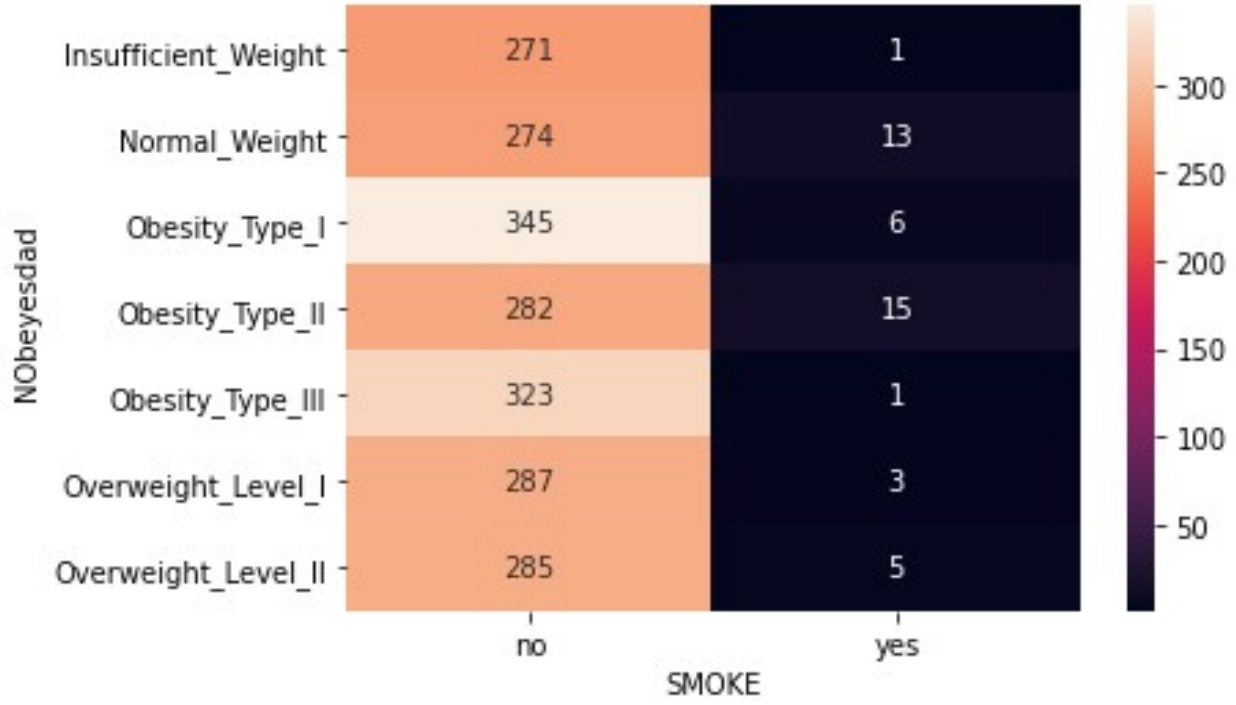
5-3- SCC vs NObeyesdad



Moins on surveille ou consomme les nourritures riches en énergie caloriques (glucides, lipides et protéines) plus on s'expose au risque d'avoir de problèmes de santé (surpoids et d'obésité).

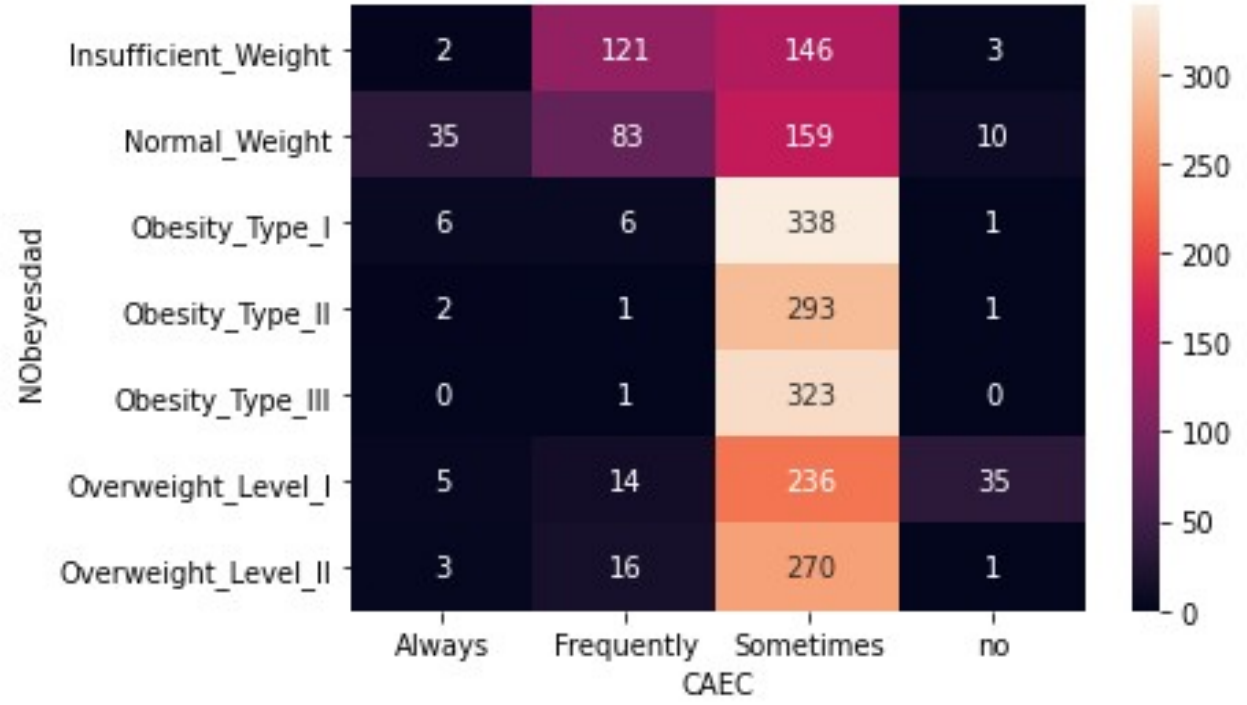
L'article suivant https://www.medicalnewstoday.com/articles/325885 appuie également cette hypothèse. Pour en savoir plus, veuillez consulter cet article.

5-4- SMOKE vs NObeyesdad



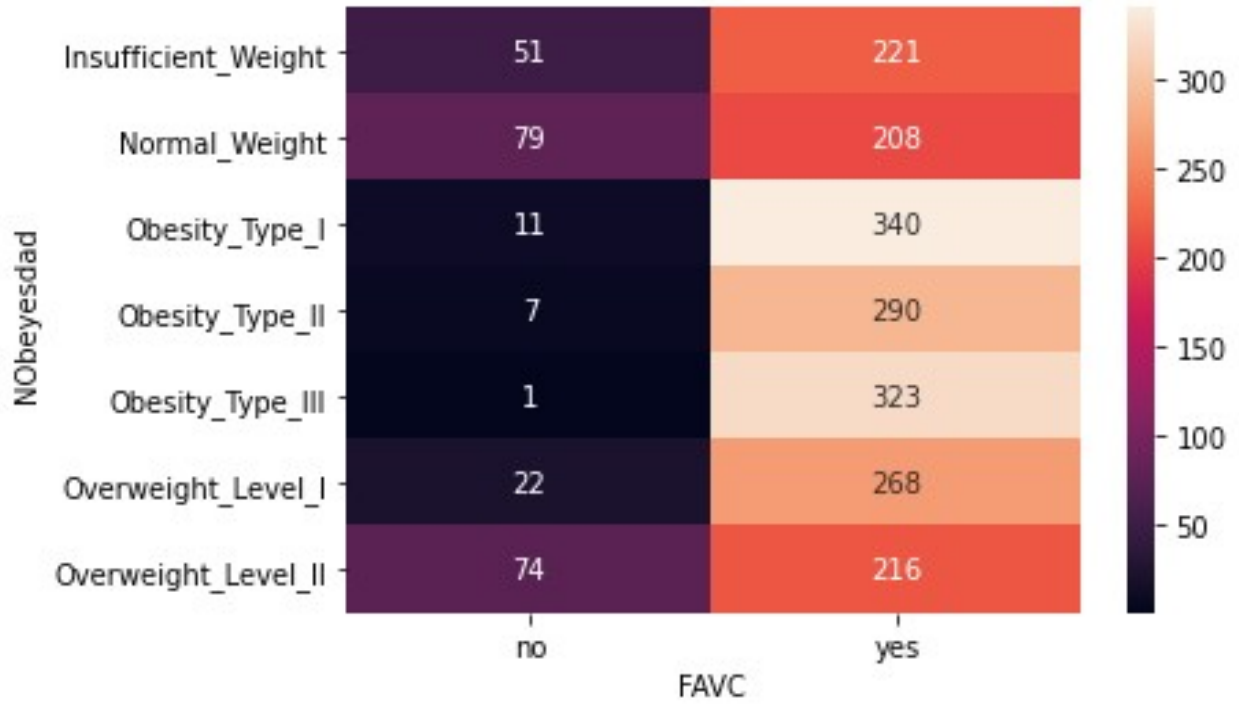
D'après ces informations, le tabagisme agit sur la santé (Poids de l'homme). En effet, on remarque que les non-fumeurs sont souvent en surpoids comparés aux fumeurs. Pour en connaître les raisons, nous vous invitons à lire l'article : Tabagisme-obesite-et-diabete-une-interaction-cliniquement-importante Section : Tabagisme et poids.

5-5- CAEC vs Nobeyesdad



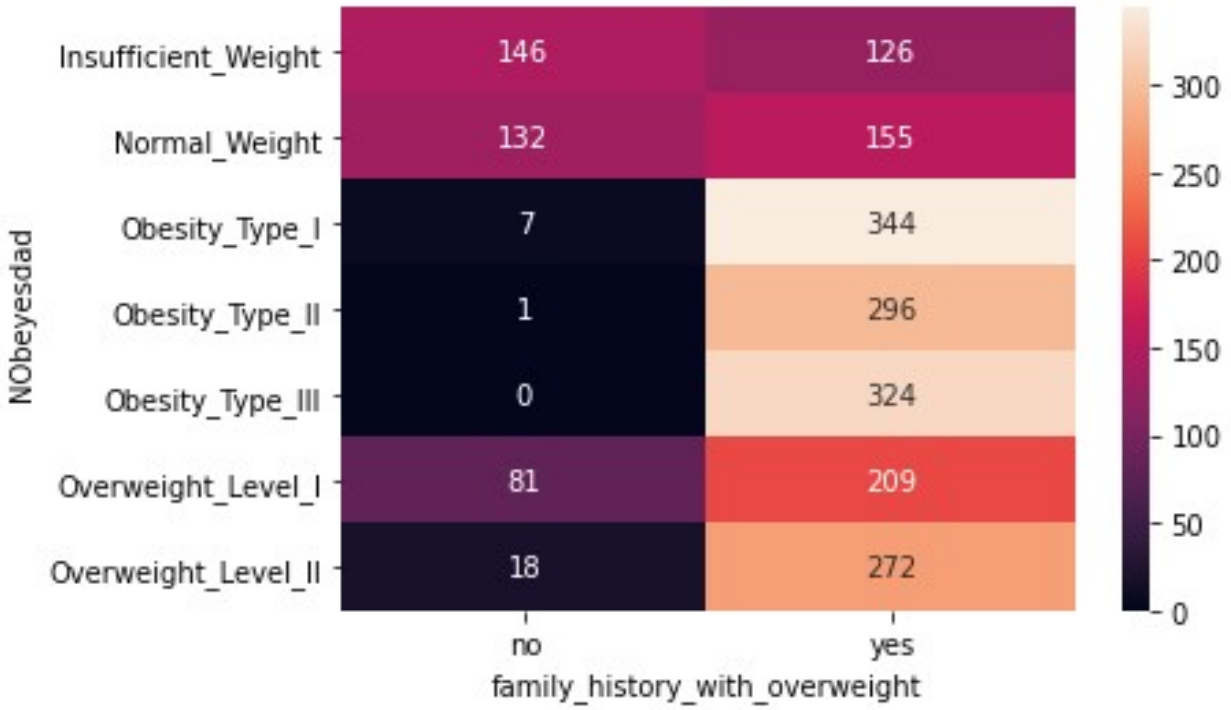
En analysant ces informations, on peut dire que la consommation de nourriture entre les repas n'agit pas sur la santé c’est-à-dire cela n'est pas vraiment source de surpoids voire d'obésité.

5-6- FAVC vs NObeyesdad



Il y a une grande différence entre les personnes consommant les nourritures très caloriques et celles qui n'en consomment pas. Mais il est vraiment difficile d'interpréter ces résultats car parmi les personnes consommant les nourritures très caloriques on trouve une bonne proportion des personnes normales (208) même si les proportions d'obésité et de surpoids sont aussi élevées. On ne peut tirer une conclusion sur cette variable.

5-7- family_history_with_overweight vs NObeyesdad

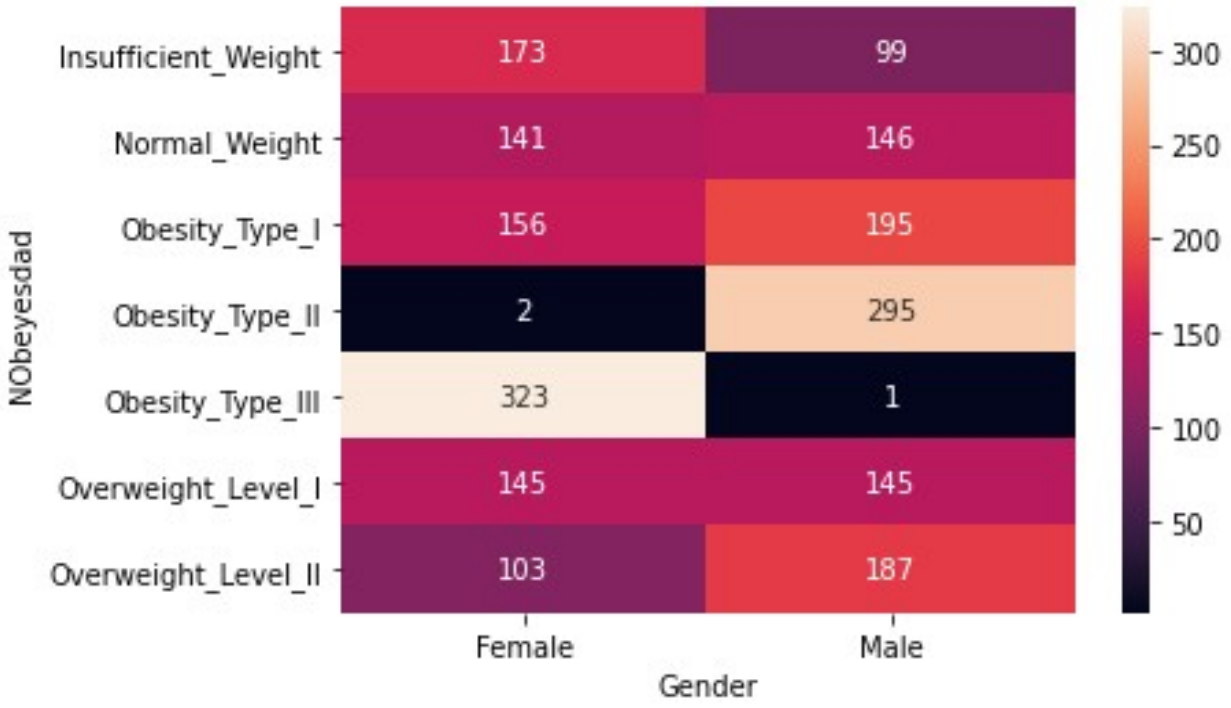


Avoir une famille dont les antécédents sont en surpoids augmente le risque d'être également en surpoids ou en obésité.

5-8- Gender vs NObeyesdad

Tout genre (féminin ou masculin) peut souffrir de l’obésité. Mais comparées aux hommes, les femmes souffrent plus d'obésité et de sous poids. Pour surpoids, ils sont sensiblement égaux. Cette variable nécessite donc une attention particulière.

Il faut bien noter que toutes les hypothèses faites doivent être vérifiées en consultant, par exemple, un expert du domaine (Santé).



6 - Relations entre variables quantitatives

Indice	Age	Height	Weight	FCVC	NCP	CH2O	FAF	TUE
Age	1	-0.0259581	0.20256	0.0162909	-0.0439437	-0.0453039	-0.144938	-0.296931
Height	-0.0259581	1	0.463136	-0.0381211	0.243672	0.213376	0.294709	0.0519117
Weight	0.20256	0.463136	1	0.216125	0.107469	0.200575	-0.0514363	-0.0715614
FCVC	0.0162909	-0.0381211	0.216125	1	0.0422163	0.0684615	0.0199394	-0.101135
NCP	-0.0439437	0.243672	0.107469	0.0422163	1	0.057088	0.129504	0.0363256
CH2O	-0.0453039	0.213376	0.200575	0.0684615	0.057088	1	0.167236	0.0119653
FAF	-0.144938	0.294709	-0.0514363	0.0199394	0.129504	0.167236	1	0.0585621
TUE	-0.296931	0.0519117	-0.0715614	-0.101135	0.0363256	0.0119653	0.0585621	1

Matrice de corrélation

La matrice de corrélation ci-dessous montre qu’aucune valeur de la matrice de corrélation en valeur absolue est supérieure à 0.5 car toutes sauf sur la diagonale (corrélation de la variable elle-même qui est toujours égale à 1).

D'où les variables (quantitatives) n'ont aucune relation forte entre elles. Donc elles sont indépendantes.

Après avoir fait une analyse profonde (inférence) et bien compris les données, passons maintenant au traitement de données. En somme, nous allons encoder les données qualitatives, puis normaliser l'ensemble de données car, comme vu dans la première partie, les données ne sont pas toutes sur une même échelle. Finalement, nous diviserons l'ensemble de données en deux : une partie pour l’entraînement et une autre pour le test du modèle qu'on développera dans la suite.

CHAPITRE 3 : PRÉ-TRAITEMENT DE DONNÉES

I – TRAITEMENT DE DONNÉES MANQUANTES

Comme vu dans la partie précédente, l'ensemble de données il n'y a pas de de données manquantes. Donc, nous n'allons pas traiter ce problème tout au long du projet.

```
Gender                                0.0
CALC                                  0.0
TUE                                   0.0
FAF                                   0.0
SCC                                   0.0
CH2O                                  0.0
SMOKE                                 0.0
MTRANS                                0.0
CAEC                                  0.0
FCVC                                  0.0
FAVC                                  0.0
family_history_with_overweight        0.0
Weight                                0.0
Height                                0.0
Age                                   0.0
NCP                                   0.0
NObeyesdad                            0.0
dtype: float64
```

II – ENCODAGE DE DONNÉES

Nous avons dans notre ensemble de données neuf (9) variables qualitatives (catégorielles).

```
In [4]: data.dtypes.value_counts()
Out[4]:
object     9
float64    8
dtype: int64
```

La présence de ces variables dans les données complique généralement l'apprentissage. En effet, la plupart des algorithmes d'apprentissage automatique prennent des valeurs numériques en entrée. Ainsi, il faut trouver une façon de

transformer nos modalités en données numériques. Sur ce, nous allons utiliser un algorithme implémenter sur la librairie sklearn : LabelEncoder().

Après avoir transformé ces variables, on a maintenant :

```
In [7]: data_encoded.dtypes.value_counts()
Out[7]:
int64      9
float64    8
dtype: int64
```

On remarque bien qu'il n'y a plus de variables de type Object autrement dit catégoriel. Or vraiment bien qu'il n'y ait plus de variables de type objet autrement dit catégorielles.

III – NORMALISATION DE DONNÉES

De même dans la partie analyse nous avons vu que les données ne sont pas sur la même échelle ainsi cela nécessitera une normalisation.

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, nous allons faire appel la méthode StandardScaler de sklearn.preprocessing.

IV – DONNÉES D'ENTRAÎNEMENT ET DE VALIDATION

Comme nous le savons, l'entraînement d'un modèle revient à mesurer l'erreur de la sortie de l'algorithme avec les données d'exemple et chercher à la minimiser. Travailler donc avec toutes les données d'un seul coup engendre le problème de sur-apprentissage(Overfitting) ou sous-apprentissage(Underfitting).

Pour minimiser ce problème, nous allons séparer, dès le départ, notre jeu de données en deux parties distinctes :

- Le training set, qui va nous permettre d’entraîner notre modèle et sera utilisé par l’algorithme d’apprentissage. C'est celui dont on a parlé depuis le début.
- Le testing set qui permet de mesurer l’erreur du modèle final sur des données qu’il n’a jamais vues. On va simplement passer ces données comme s'il s'agissait de données que l’on n'a encore jamais rencontrées (comme cela va se passer ensuite en pratique pour prédire de nouvelles données) et mesurer la performance de notre modèle sur ces données.

70% de l’ensemble de données seront utilisés pour l’entraînement du modèle et 30% pour la validation.

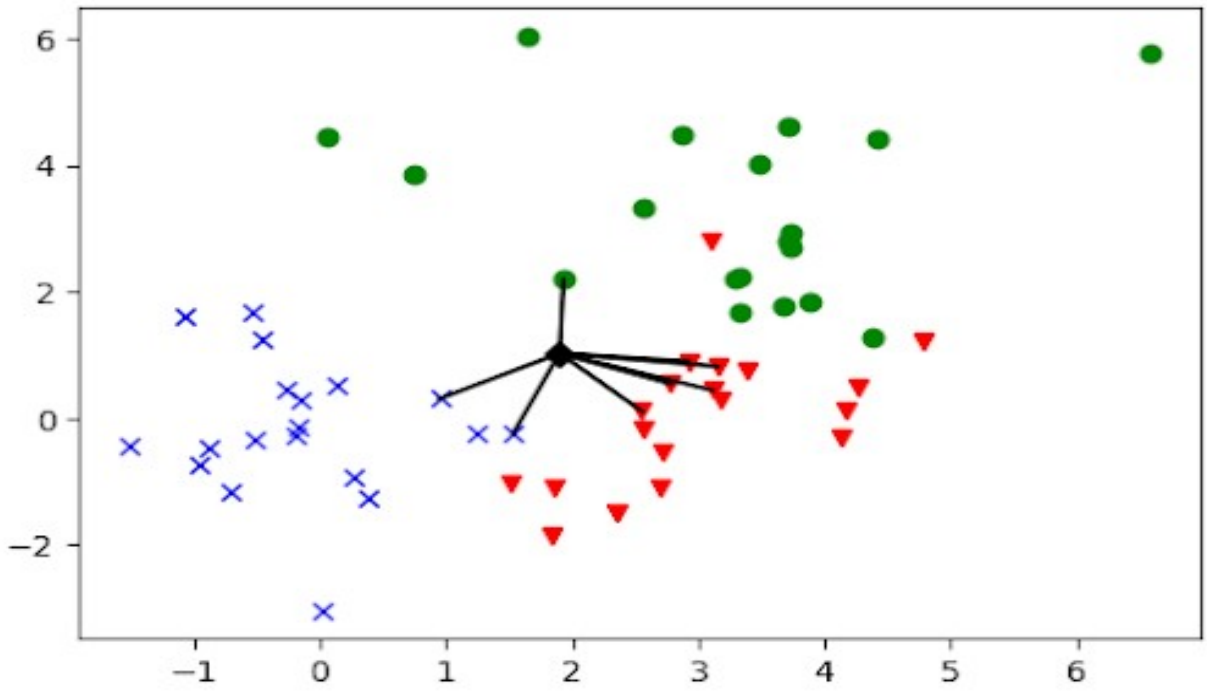
Fini avec le pré-traitement de données. Nous allons maintenant modéliser ces données. En gros, déterminer quel algorithme décrit ou résume correctement notre ensemble de données.

CHAPITRE 4 : MODÉLISATION ET OPTIMISATION

Nous allons, dans ce chapitre, classifier les données en niveaux d'obésité. La classification est un type d'apprentissage supervisé. Il spécifie la classe à laquelle appartiennent les éléments de données et est mieux utilisé lorsque la sortie a des valeurs finies et discrètes. Il prédit également une classe pour une variable d'entrée. Il s'agit ici d'une classification multi-classes car on a plus de deux niveaux d'obésité dans la variable cible (NObeyesdad). Les algorithmes utilisés pour classifier ces données sont KNN (K-Nearest Neighbors en français k plus proches voisins), SVM (Support Vector Machine), RandomForest et Réseaux de neurones.

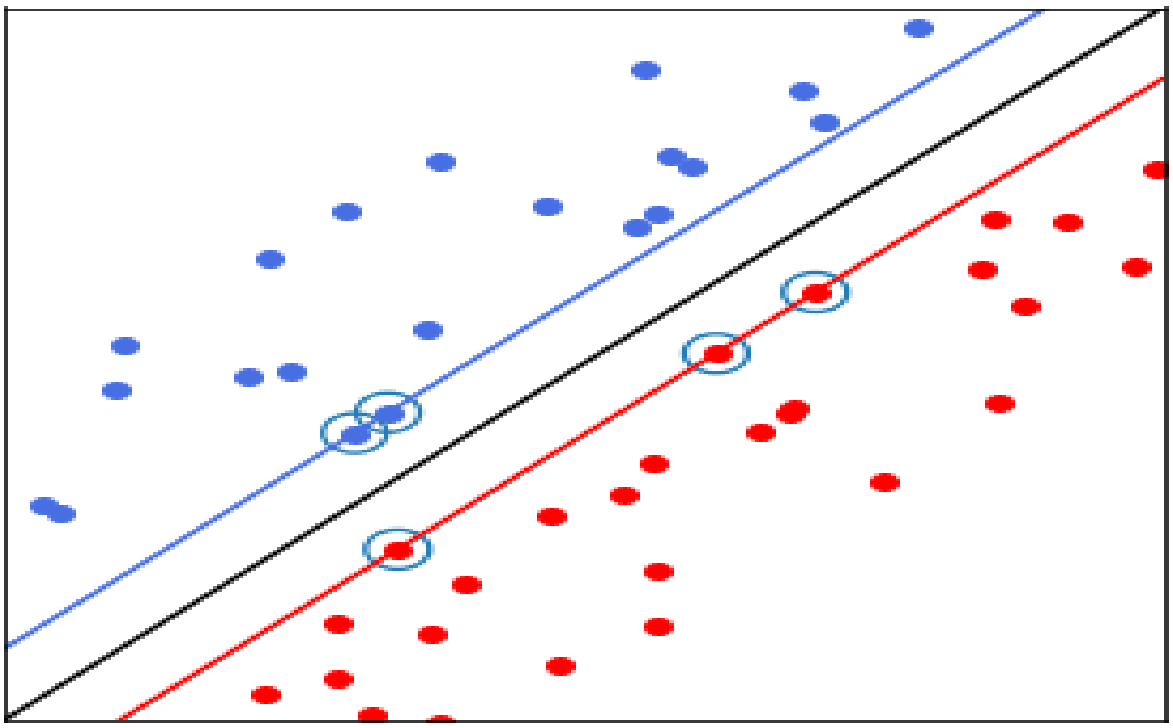
I - DÉFINITION DES ALGORITHMES

La méthode des K plus proches voisins (KNN) a pour but de classifier des points cibles (classe méconnue) en fonction de leurs distances par rapport à des points constituant un échantillon d'apprentissage (c'est-à-dire dont la classe est connue a priori).



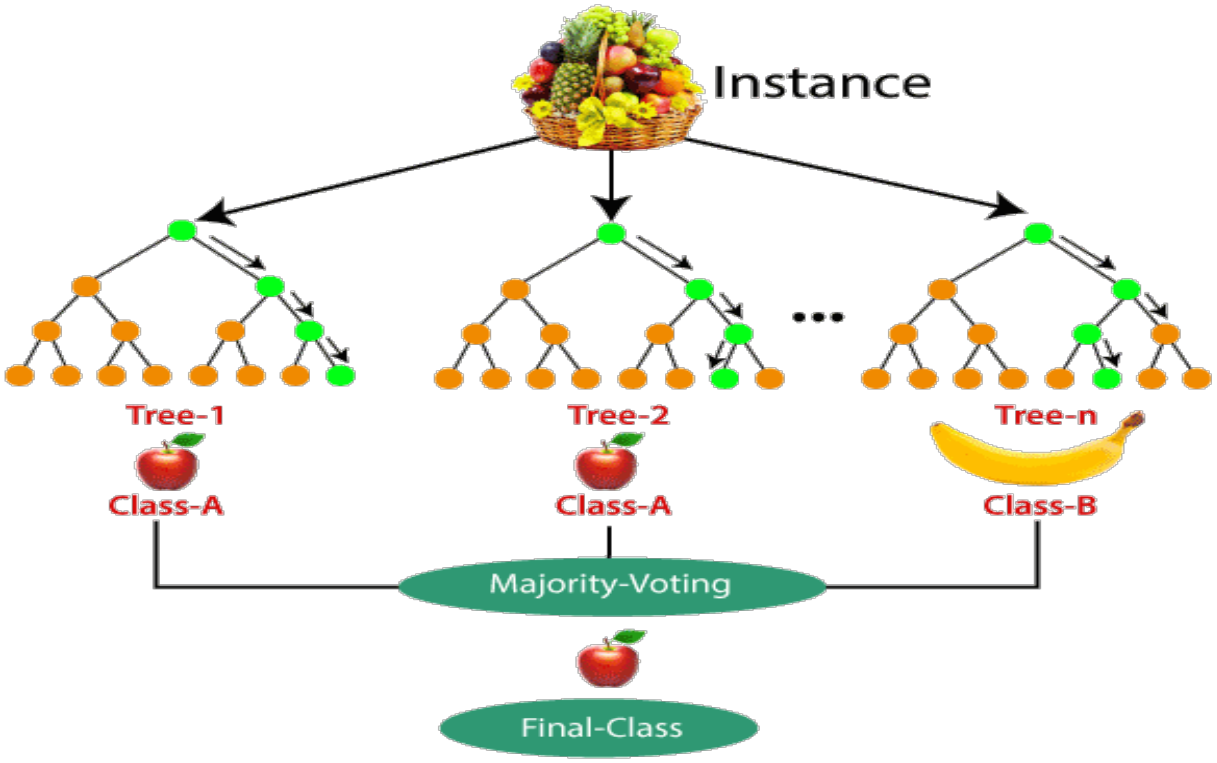
Source : Classification avec KNN

Les machines à vecteurs de support ou séparateurs à vaste marge SVM ont pour but de séparer les données en classes à l'aide d'une frontière aussi « simple » que possible, de telle façon que la distance entre les différents groupes de données et la frontière qui les sépare soit maximale. Cette distance est aussi appelée « marge » , les « vecteurs de support » étant les données les plus proches de la frontière.



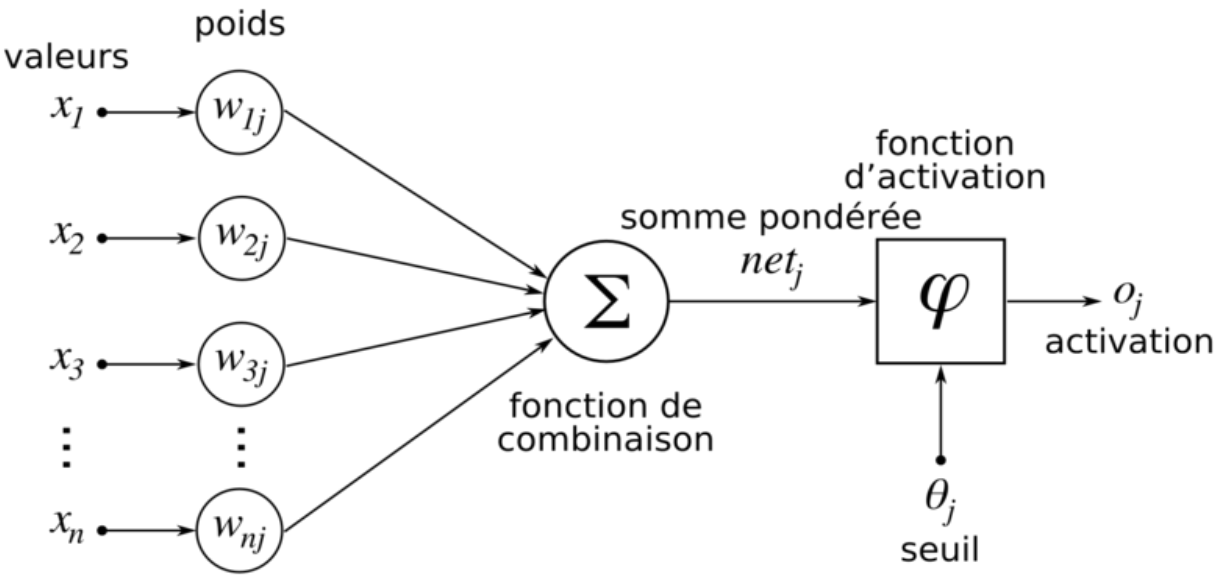
Source : Classification avec SVM

RandomForest effectue un apprentissage en parallèle sur de multiples arbres de décision construits aléatoirement et entraînés sur des sous-ensembles de données différents. Les prédictions sont ensuite moyennées lorsque les données sont quantitatives ou utilisés pour un vote pour des données qualitatives, dans le cas des arbres de classification.



Source : Classification avec RandomForest

Les réseaux de neurones (ANN), communément appelés des réseaux de neurones artificiels sont des imitations simples des fonctions d'un neurone dans le cerveau humain pour résoudre des problématiques d'apprentissage de la machine (Machine Learning). Dans notre cas, le réseau de neurones a une couche d'entrée, deux couches cachées et une couche de sortie.



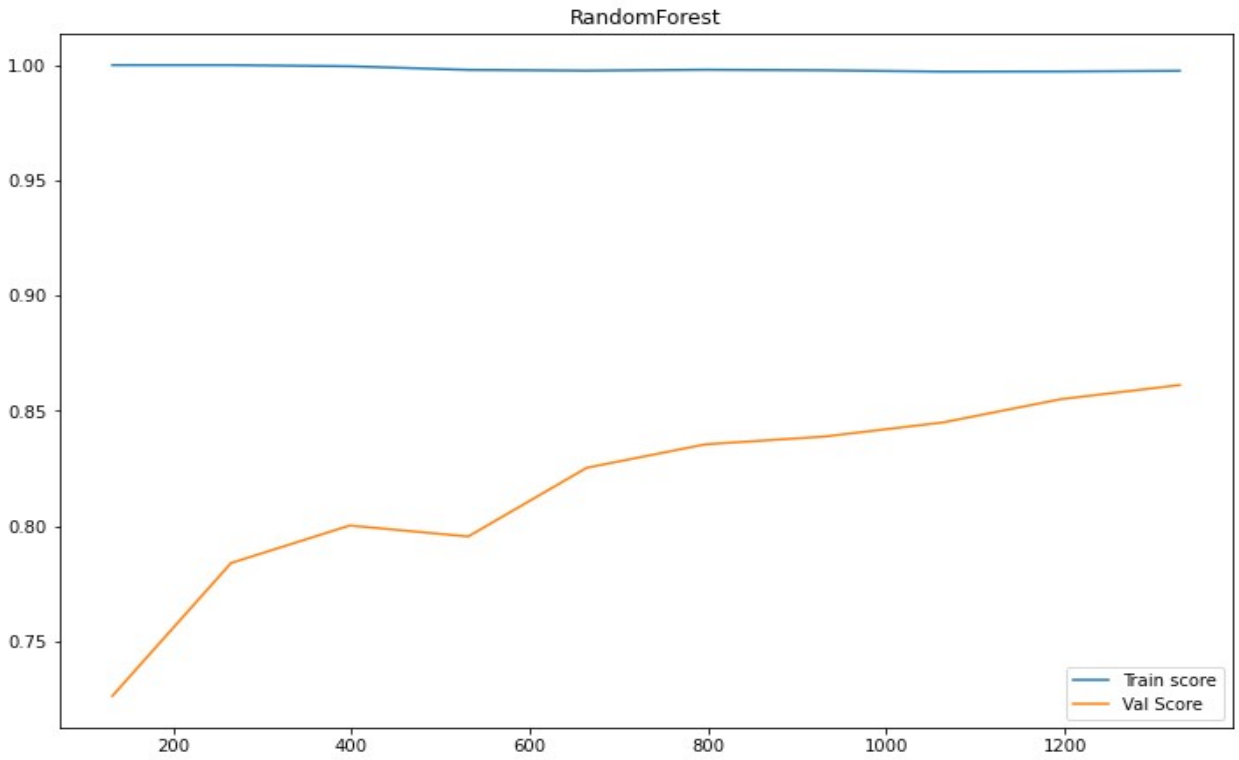
Source : Architecture d'un réseau de neurones

Vu le nombre de variables que contient cet ensemble de données, nous allons faire une sélection des variables dans la deuxième partie de la classification afin d'éliminer les variables ne contribuant pas à l'explication de la variable de décision Nobeyesdad.

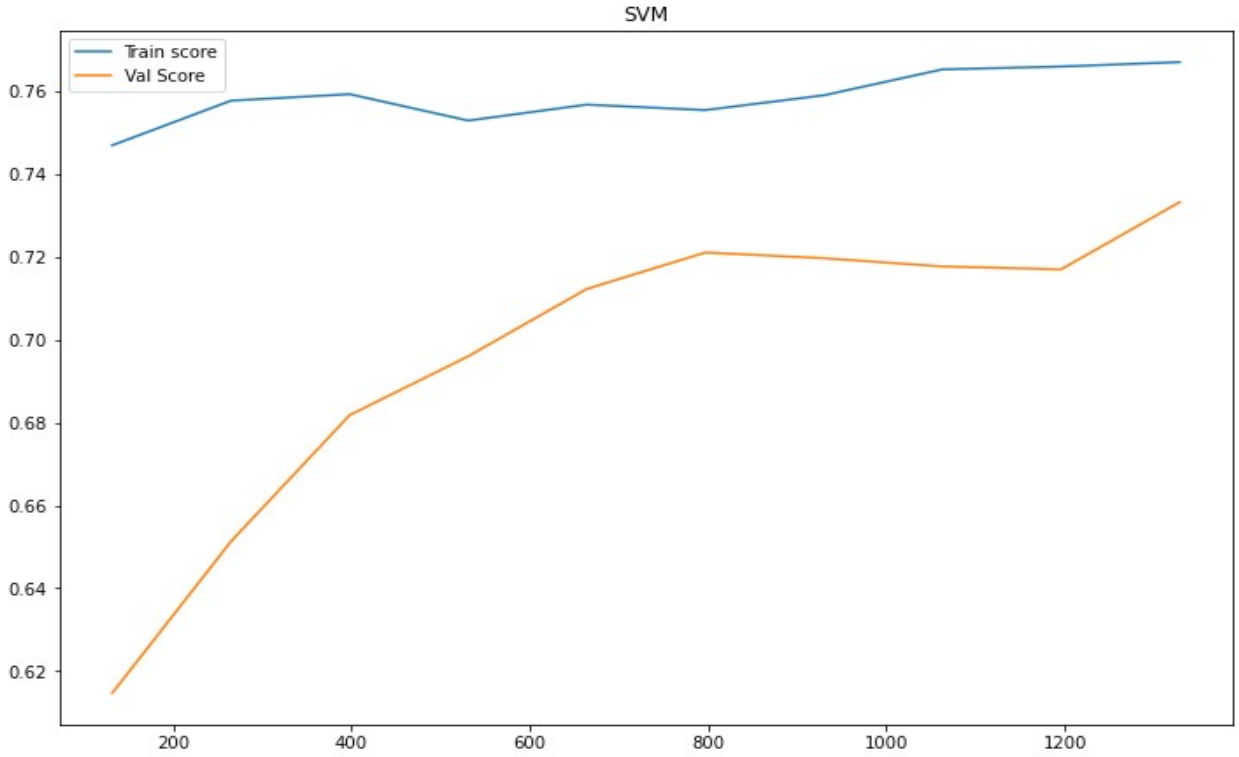
II – CLASSIFICATION SANS SÉLECTION DES VARIABLES

Dans cette section nous classifions les données sans éliminer une variable. En d'autres termes, on applique la classification sur toutes les données. On utilise bel et bien le training set et testing set pour mesurer la performance plus précisément la précision (accuracy) de modèles cités ci-dessus.

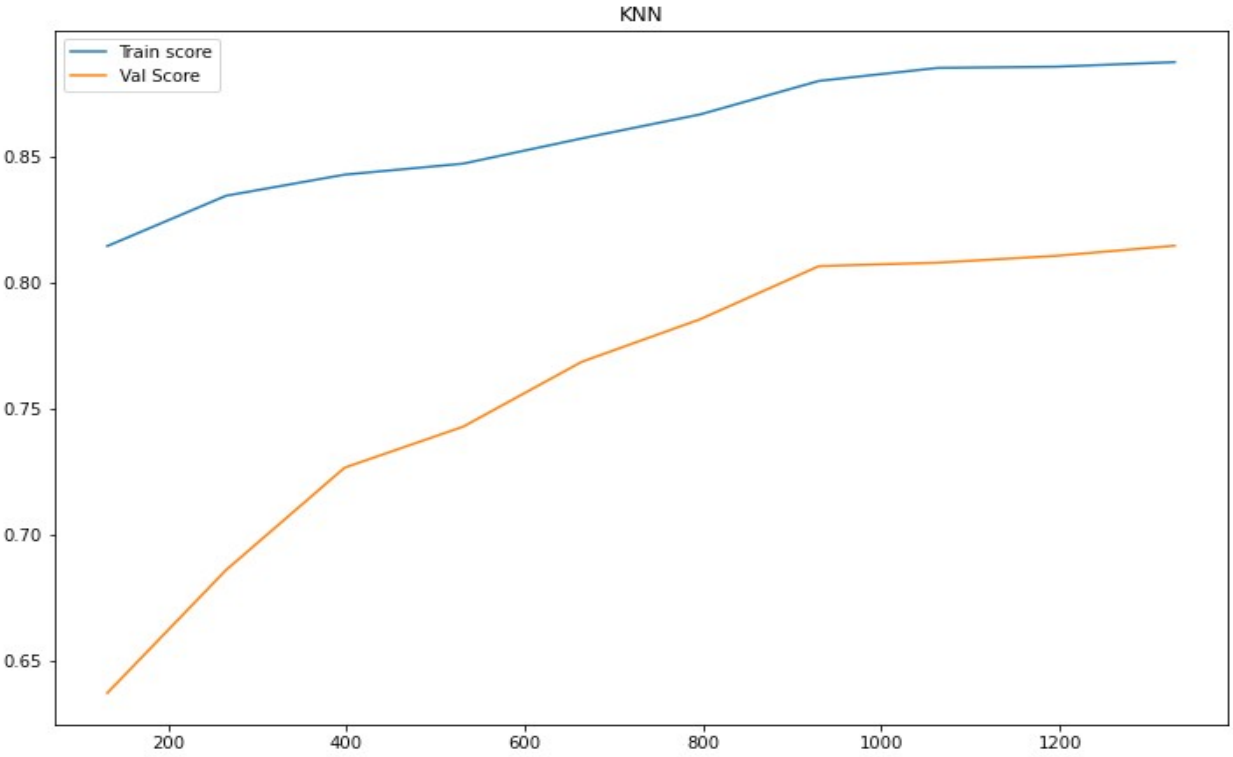
Après avoir implémenté ces algorithmes, RandomForest donne une précision de 86%, 82% pour KNN, 75% SVM et Réseau de neurones est le meilleur modèle avec 95% de précision.



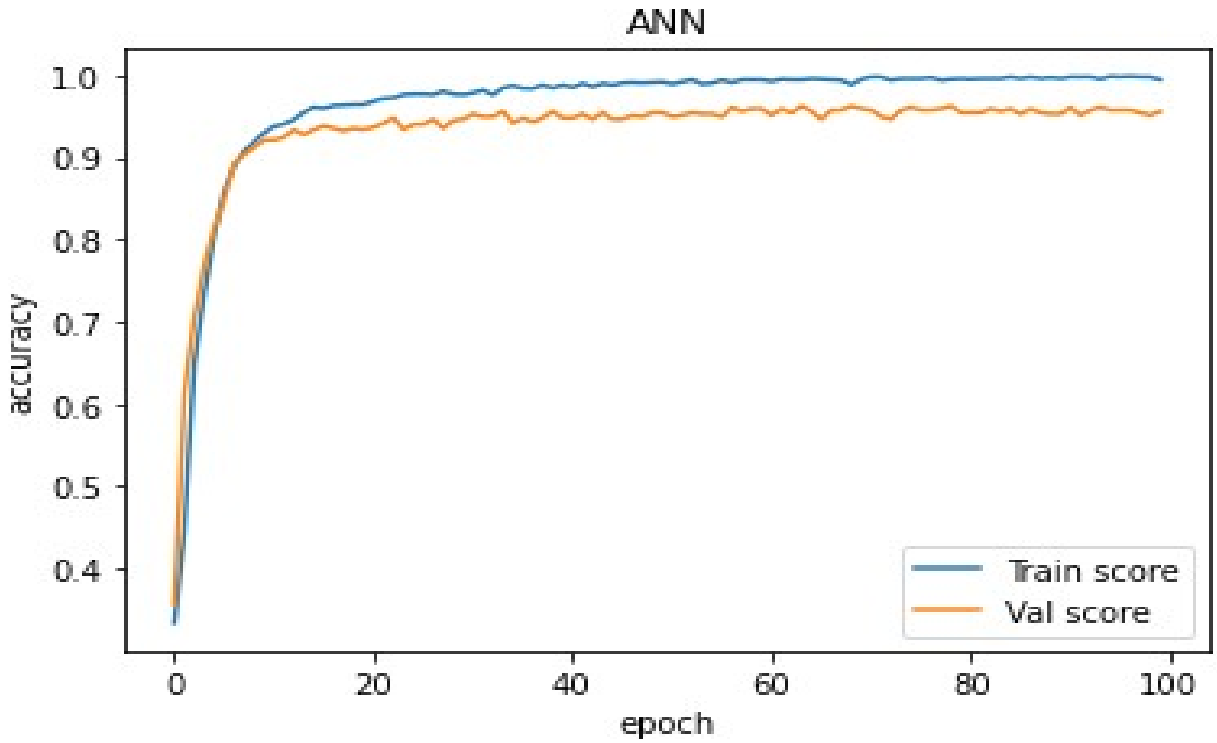
Performance de RandomForest sur les données d'apprentissage et de test



Performance de SVM sur les données d'apprentissage et de test



Performance de KNN sur les données d'apprentissage et de test



Performance de Réseau de neurones sur les données d'apprentissage et de test

On remarque que l’on peut s’arrêter à 40 epochs au lieu de 100 pour notre réseau de neurones car on a sensiblement les mêmes performances.

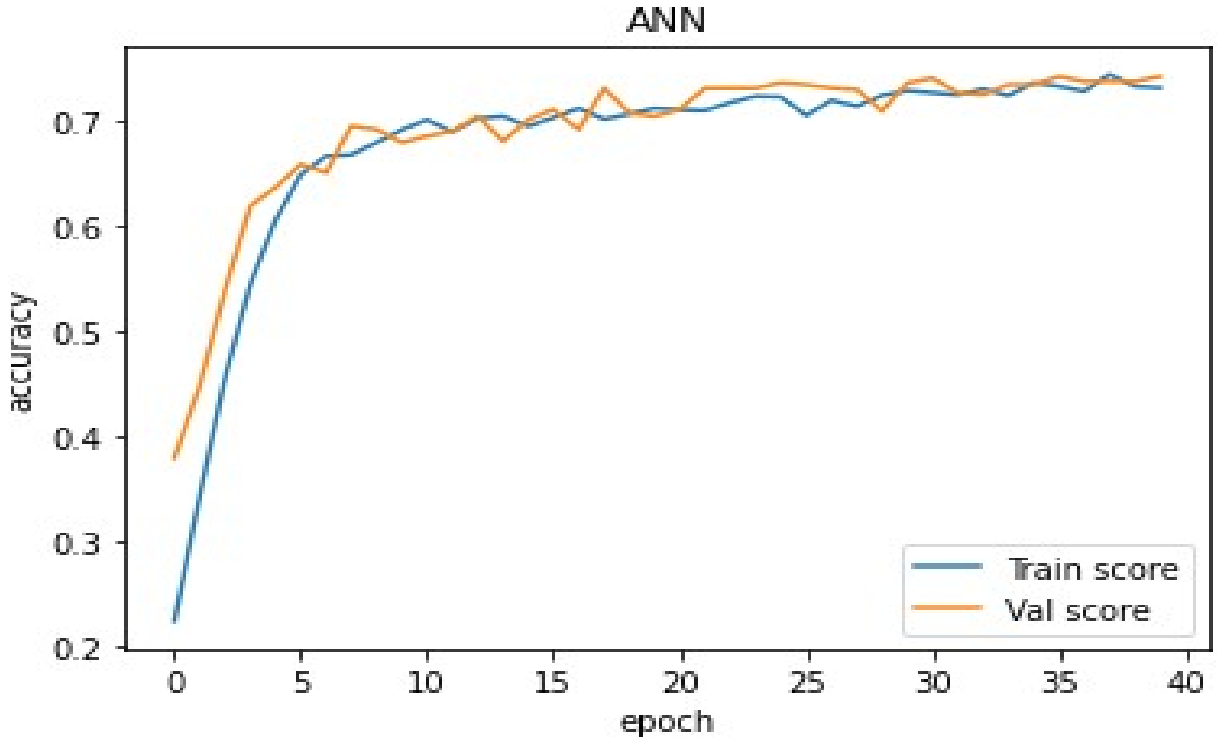
III – SÉLECTION DES VARIABLES (FEATURE SELECTION)

Pour faire de sélection de variables, nous allons utiliser deux méthodes : SelectKBest et SelectFromModel.

1 – SelectKBest

SelectKBest sélectionne les K variables de X dont le score du test de dépendance chi2 avec y est plus élevé. Après une recherche de nombre de paramètres à garder sans modifier la performance des algorithmes, on a trouvé k égal à 8 comme le nombre minimal de variables avec lesquelles on peut travailler et dépendent fortement de niveaux d’obésité. Ainsi, les variables retenues sont : Gender (genre), Age (âge), Weight (poids), family_history_with_overweight (parents en surpoids), FCVC (Fréquence de consommation de légumes), SCC (Surveillance de la consommation de calories), FAF (Fréquence de l'activité physique), MTRANS (Moyens de transports).

Nous avons ainsi obtenu les performances suivantes : 86% de précision avec RandomForest, 82% avec KNN, 75% avec SVM et 76% Réseau de neurones. On remarque que, les trois premiers algorithmes ont gardé les mêmes performances. Mais, la précision de réseau de neurones a fortement chuté comme le montre la figure ci-après :

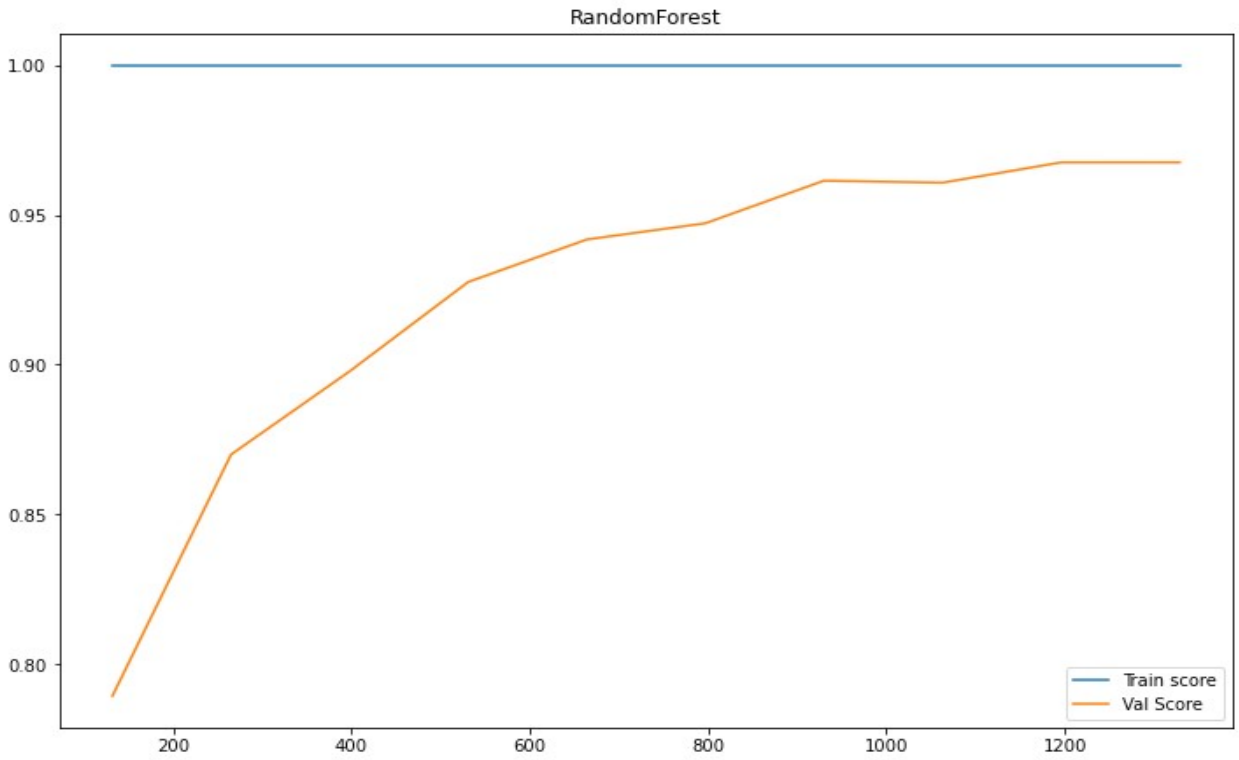


2 – SelectFromModel

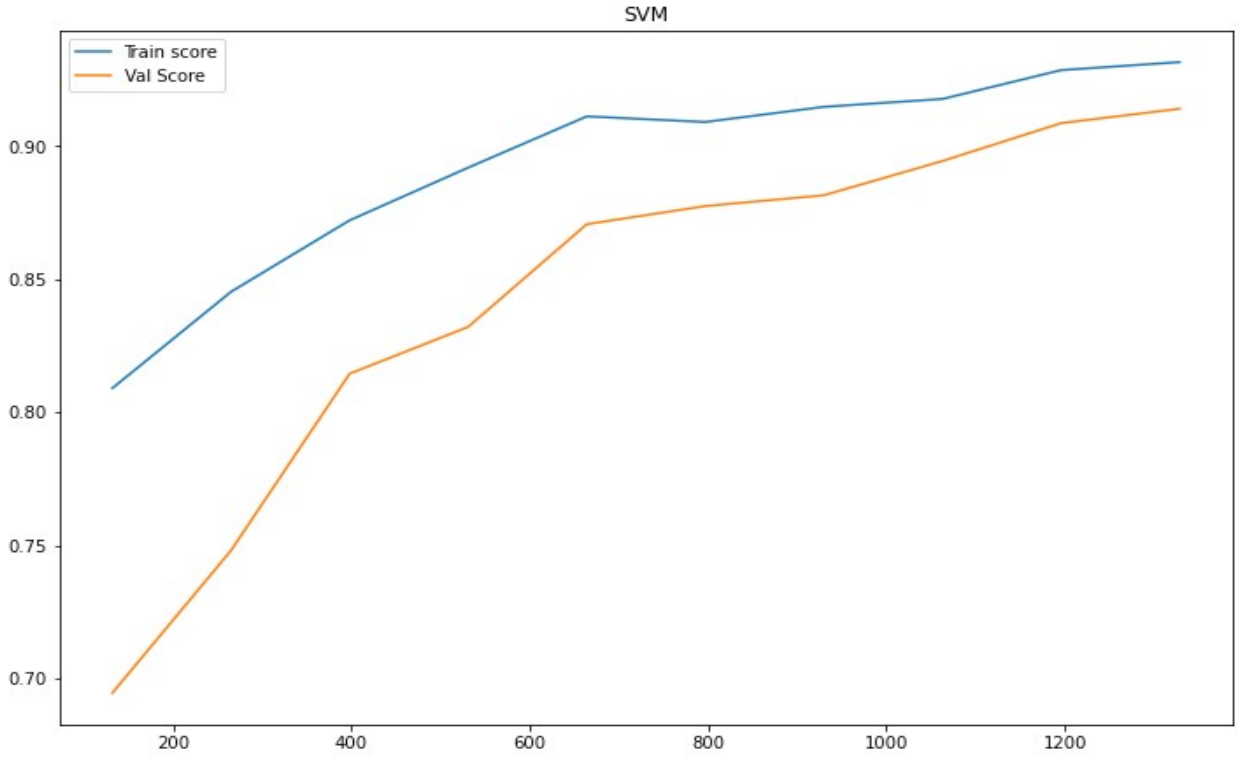
SelectFromModel est utilisé pour sélectionner les variables les plus importantes dans un modèle afin qu'elles puissent être utilisées pour entraîner un autre modèle. On peut remarquer, jusqu'ici, que RandomForest est plus efficace alors on l'utilisera pour sélectionner ces variables.

Cette méthode nous renvoie quatre variables essentielles à la prédiction du niveau d'obésité : Age, Height, Weight et FCVC.

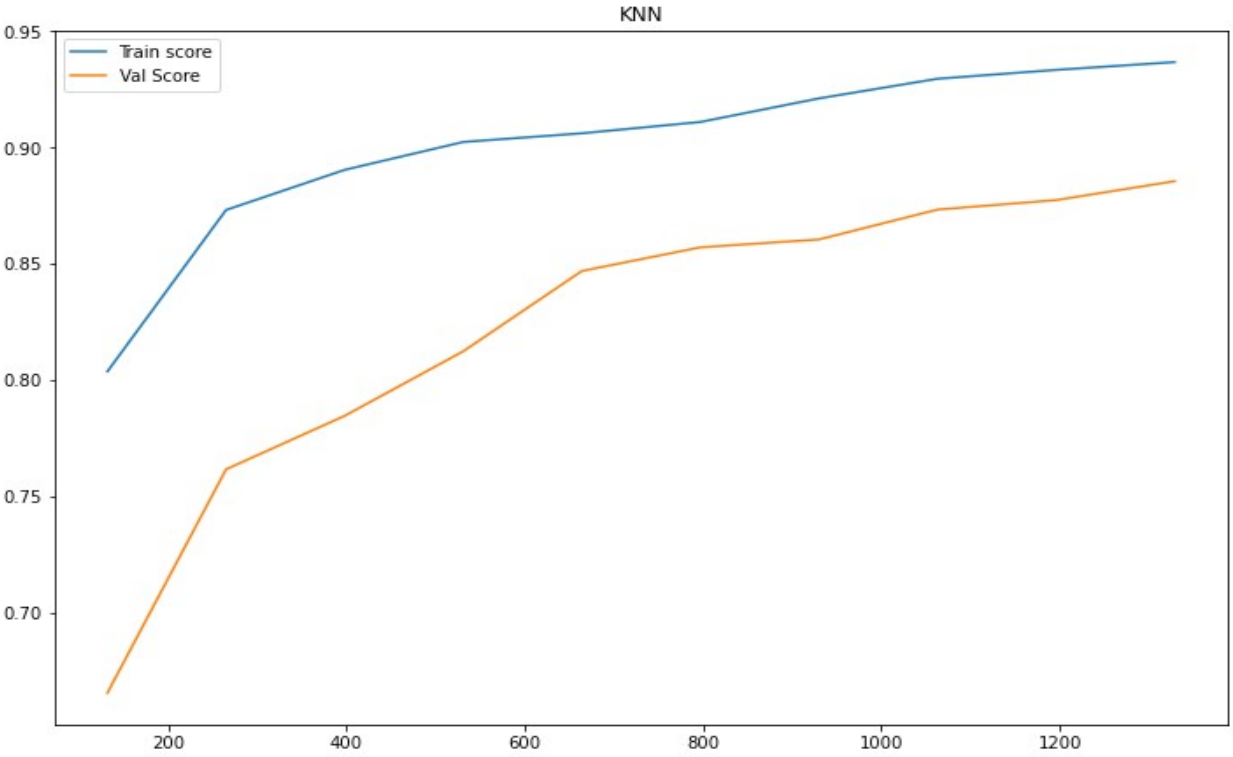
Avec ces dernières, on trouve de bonnes performances. 96% de précision pour RandomForest, 92% pour KNN, 92% pour SVM et 94% pour le réseau de neurones.



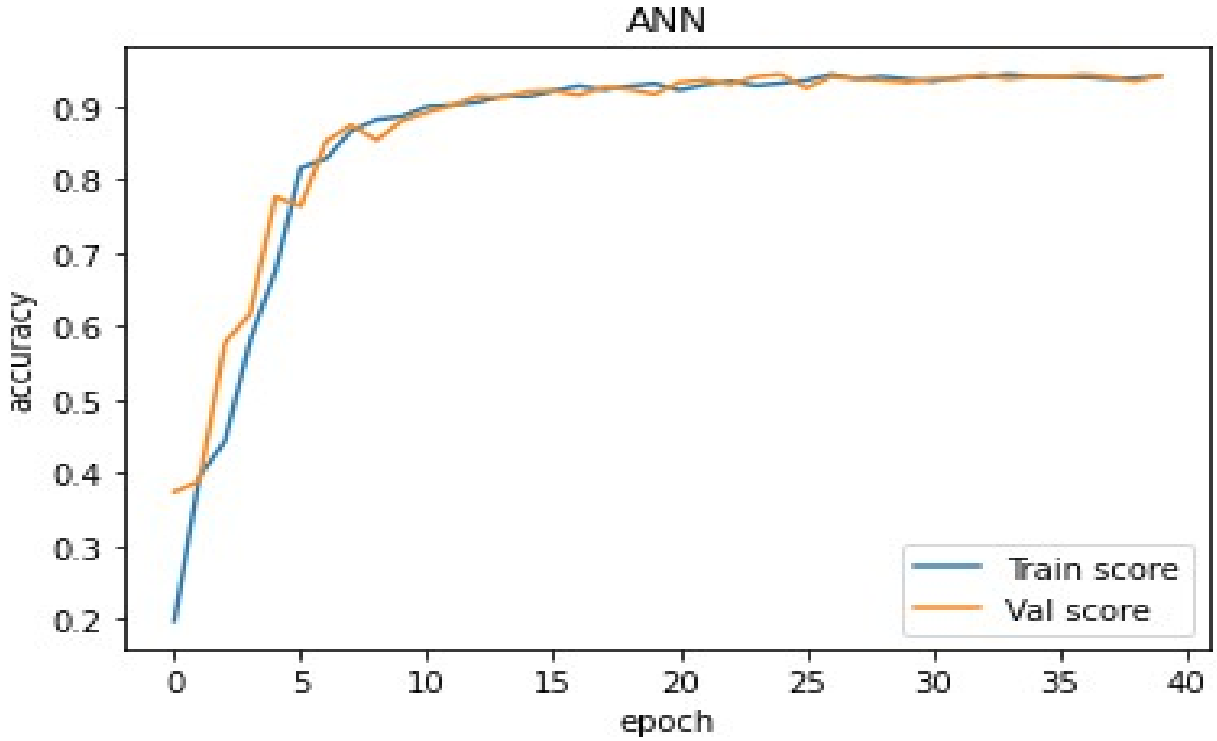
Performance de RandomForest



Performance de SVM



Performance de KNN



Performance de Réseau de neurones

On peut donc retenir ces quatre variables afin d’optimiser ces modèles par la méthode GridSearchCV. Ces dernières nous donnent des bonnes performances.

IV – OPTIMISATION AVEC GridSearchCV

Un facteur important dans les performances de ces modèles sont leurs hyperparamètres, une fois que nous avons défini des valeurs appropriées pour ces hyperparamètres, les performances d'un modèle peuvent s'améliorer considérablement. Par exemple, n_neighbors (nombre de voisins) et metric (la métrique à utiliser) sont les hyper-paramètres de KNN.

Tout d'abord, comprenons ce qu'est la méthode GridSearchCV? Il s'agit, tout simplement, du processus de réglage des hyper-paramètres afin de déterminer les valeurs optimales pour un modèle donné. Comme mentionné ci-dessus, les performances d'un modèle dépendent de manière significative de la valeur des hyper-paramètres.

Ainsi, après avoir réglé les hyperparamètres de nos modèles, les performances de ces derniers se résument comme suit: 96% de précision avec RandomForest, 94% avec KNN, 92% avec SVM et 94% avec le réseau de neurones.

Nous pouvons maintenant sauvegarder nos modèles que nous utiliserons dans notre application Web.

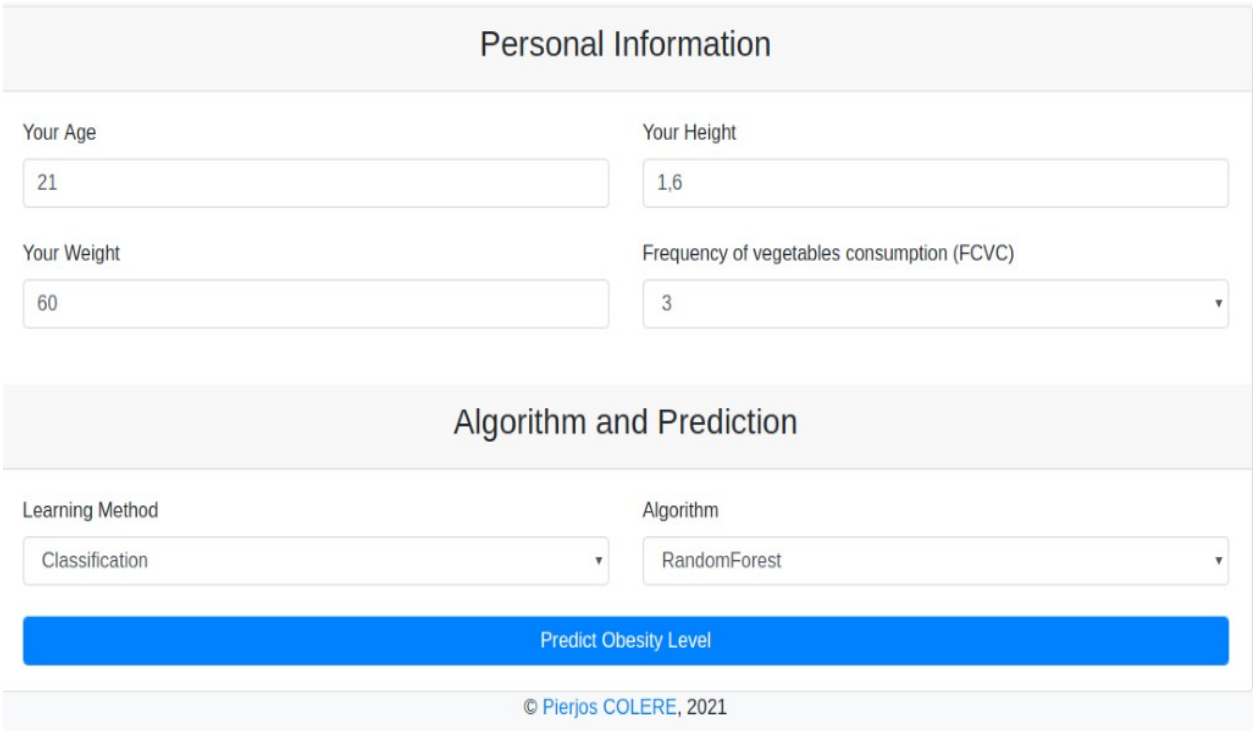
CHAPITRE 5 : APPLICATION WEB (DJANGO)

Après avoir obtenu nos modèles performants, nous avons développé une application Web dont le nom est Dont Be Obese. Un grand nombre de gens ne maîtrisent pas les notions de Machine Learning, encore moins, n'arrivent pas à interpréter les résultats des modèles de ce domaine. Alors mettre en place un dispositif leur permettant de s'y connaître serait primordial. Ainsi, cette application permet aux utilisateurs de connaître leur niveau de santé (normal, obèse ou en insuffisance de poids) et donne également la possibilité de télécharger les résultats obtenus au format PDF.

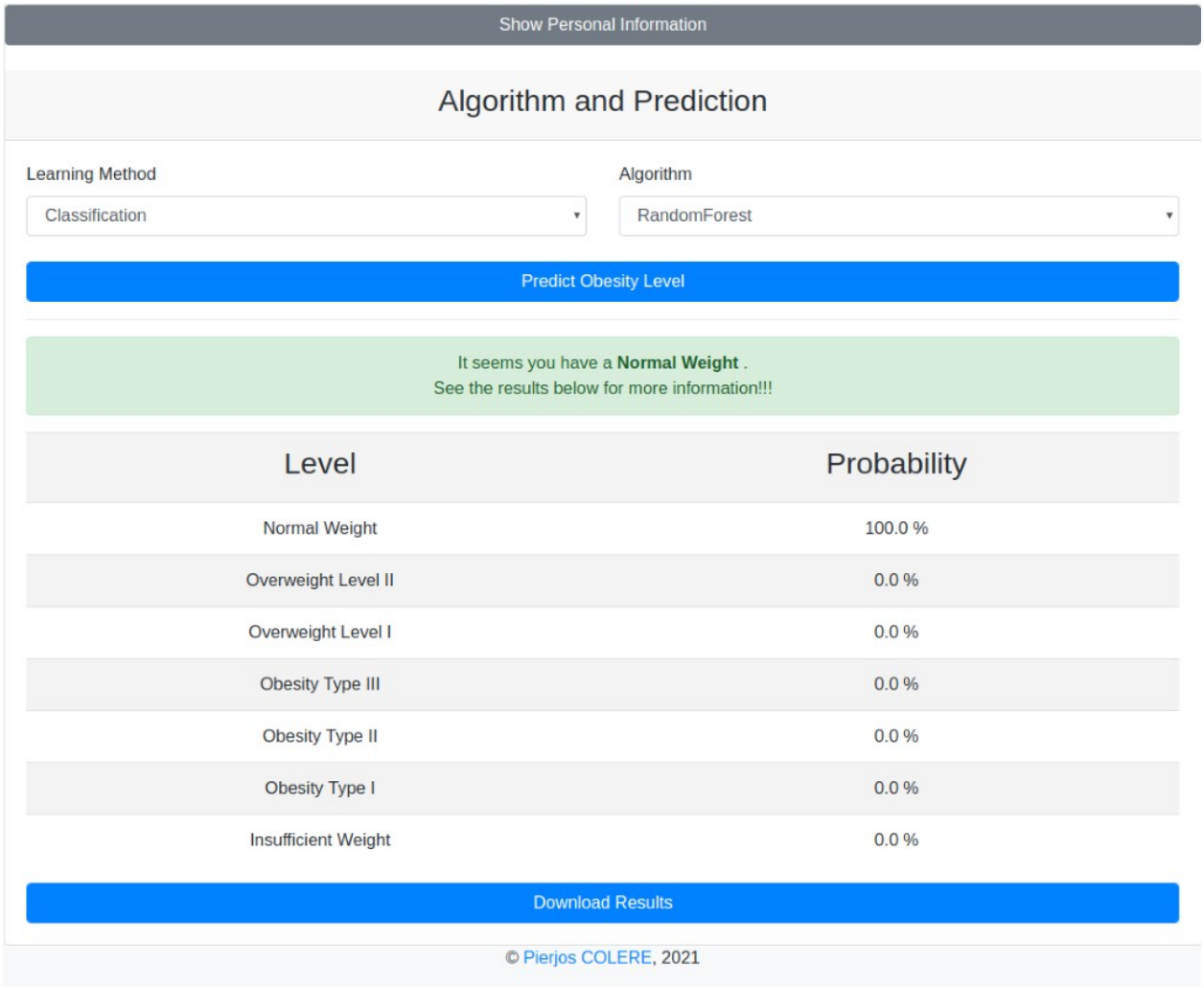
Pour ce faire, nous avons utilisé nos connaissances en Django. Qu'est-ce que Django ?

Django est un Framework Python de haut niveau, permettant un développement rapide de sites internet, sécurisés, et maintenables. Créé par des développeurs expérimentés, Django prend en charge la plupart des tracas du développement web, nous pouvons donc nous concentrer sur l'écriture de notre application sans avoir besoin de réinventer la roue. Il est gratuit, open source, a une communauté active, une bonne documentation, et plusieurs options pour du support gratuit ou non.

Voici l'application Dont Be Obese en image :



Demande de saisie d’informations



Résultats de la prédiction

Pour consulter notre site et vérifier votre niveau d’obésité alors veuillez cliquer sur ce lien dontbeobese.herokuapp.com.

CONCLUSION

Cet humble rapport présente le résultat d'un travail soutenu et assidu qui s'inscrit dans le cadre de la réalisation d'un projet du module Machine Learning.

Nous avons, dans un premier temps, compris les données utilisées issues de UCI Machine Learning Repository et analysé profondément ces données afin de dégager des hypothèses sur les causes de l'obésité.

Ensuite, après avoir pré-traité les données, nous avons développé et optimisé à travers GridSearch nos modèles de classification (KNN, SVM, RandomForest et Réseau de neurones).

Enfin, au lieu que nos modèles soient abstraits pour un large nombre de personnes (n'étant pas du domaine), nous avons mis en place une application web leur permettant de connaître leur niveau d'obésité et de télécharger les résultats afin de suivre leur état de santé et respecter l'hygiène alimentaire si possible.

Grâce à ce projet, nous avons pu renforcer nos connaissances en Machine Learning (Apprentissage Machine) et en développement Web avec Django. A travers des méthodes de travail et des outils, ce projet nous a permis de nous immerger dans un univers professionnel. Il est vrai que générer des modèles de Machine Learning ou créer une application et respecter un cahier des charges rend un projet intéressant et professionnel mais il y a aussi toutes les démarches qui ne sont pas visibles et qui rendent enrichissante une telle expérience : s'organiser sur les plans personnels, gérer les imprévus, respecter des délais pour ne pas retarder tout le projet. En somme, le projet nous a apporté une idée sur l'organisation dans le monde du travail et nous permettra de nous adapter plus facilement lors de notre stage de fin de formation pour ne citer que cela.

BIBLIOGRAPHIE

Les cours et TP de Machine Learning de Ikram CHAIRI et Hasnae ZEROUAOUI

https://archive.ics.uci.edu

https://scikit-learn.org

https://www.juripredis.com

https://www.tensorflow.org

https://www.javatpoint.com/machine-learning-random-forest-algorithm

https://www.djangoproject.com

https://dontbeobese.herokuapp.com

Mon site : pierjos-colere-website.web.app