

# **Cardiotocographie fœtale**

MOALI Sarah  
AMMAR-KHODJA Lilia  
Mokrani Manel  
LAFDHAL Ahmed

## **I. Définition de la problématique**

L'objectif de cette étude est d'explorer la possibilité d'utiliser des techniques d'apprentissage supervisé pour améliorer la précision et la rapidité du diagnostic médical des examens de cardiocotogrammes fœtaux. Le projet vise à déterminer si ces techniques peuvent aider à classer les examens en fonction de leur état de santé de manière plus efficace que les méthodes traditionnelles basées sur l'expertise humaine.

## **II. Traitement des données**

### **1. Préparation des données**

Pour ce projet, nous avons utilisé une base de données provenant de la plateforme [Kaggle](#), qui contient des enregistrements de cardiocotogrammes fœtaux avec des informations diagnostiques. Cette base de données a été mise à disposition par des chercheurs de l'Université de Porto, au Portugal.

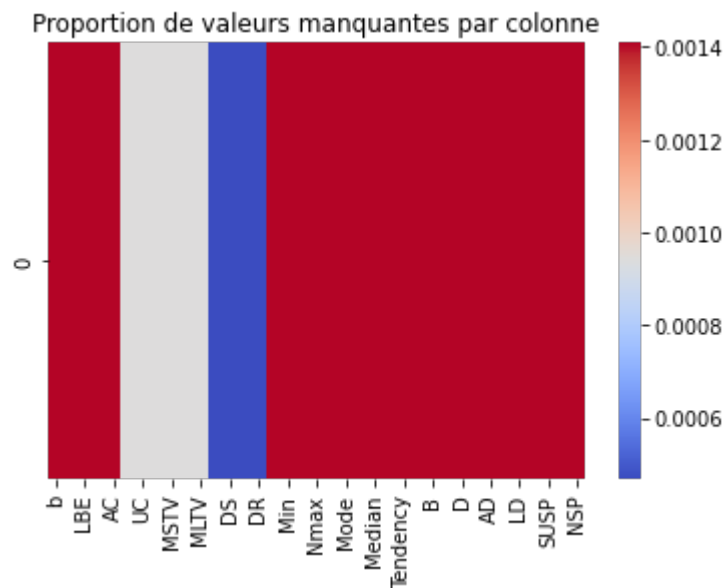
Elle se compose de 2126 enregistrements de cardiocotogrammes fœtaux (CTG), qui ont été collectés et traités automatiquement. Chaque enregistrement de CTG est associé à un ensemble de caractéristiques diagnostiques mesurées par des experts obstétriciens. Ces caractéristiques ont été utilisées pour classer chaque enregistrement de CTG en fonction d'un modèle morphologique (A, B, C, etc.) ainsi qu'en fonction de l'état fœtal (N, S, P). voici une explication plus détaillée de chaque colonne:

- FileName : Nom du fichier de l'examen CTG
- Date : Date de l'examen
- b : Instant de début
- e : Instant de fin
- LBE : Valeur de référence (expert médical)
- LB : Valeur de référence (SisPorto)
- AC : Accélérations (SisPorto)
- FM : Mouvements fœtaux (SisPorto)
- UC : Contractions utérines (SisPorto)
- ASTV : Pourcentage de temps avec une variabilité à court terme anormale (SisPorto)
- mSTV : Valeur moyenne de la variabilité à court terme (SisPorto)
- ALTV : Pourcentage de temps avec une variabilité à long terme anormale (SisPorto)
- mLTV : Valeur moyenne de la variabilité à long terme (SisPorto)

- DL : Décélérations légères
- DS : Décélérations sévères
- DP : Décélérations prolongées
- DR : Décélérations répétitives
- Width : Largeur de l'histogramme
- Min : Fréquence basse de l'histogramme
- Max : Fréquence haute de l'histogramme
- Nmax : Nombre de pics dans l'histogramme
- Nzeros : Nombre de zéros dans l'histogramme
- Mode : Mode de l'histogramme
- Mean : Moyenne de l'histogramme
- Median : Médiane de l'histogramme
- Variance : Variance de l'histogramme
- Tendency : Tendence de l'histogramme : -1 = asymétrique à gauche ; 0 = symétrique ; 1 = asymétrique à droite
- A : Sommeil calme
- B : Sommeil paradoxal
- C : Vigilance calme
- D : Vigilance active
- SH : Modèle de changement (A ou Susp avec changements)
- AD : Modèle accélérateur/décélérateur (situation de stress)
- DE : Modèle décélératif (stimulation vagale)
- LD : Modèle largement décélératif
- FS : Modèle plat-sinusoïdal (état pathologique)
- SUSP : Modèle suspect
- CLASS : Code de classe (1 à 10) pour les classes A à SUSP
- NSP : État fœtal normal = 1 ; Suspect = 2 ; Pathologique = 3

## 1. Traitement des valeurs manquantes

Les valeurs manquantes sont courantes dans les ensembles de données réels et peuvent affecter la qualité des résultats de l'analyse. Dans ce projet, nous avons travaillé avec un ensemble de données médicales contenant 2126 enregistrements de cardiocotogrammes fœtaux (CTG), qui ont été collectés et traités automatiquement. Notre ensemble de données a des valeurs manquantes dans certaines des caractéristiques diagnostiques mesurées par des experts obstétriciens.



**Figure 1 :** Proportion de valeurs manquantes par colonne

la figure 1 représente une heatmap de la proportion de valeurs manquantes par colonne permet de visualiser les colonnes qui ont le plus de valeurs manquantes. Les colonnes avec une forte proportion de valeurs manquantes auront une couleur plus rouge, tandis que les colonnes avec moins de valeurs manquantes auront une couleur plus bleue.

Après avoir examiné les données manquantes dans notre ensemble de données, nous avons choisi de supprimer les enregistrements avec des valeurs manquantes. La raison de cette décision est que notre ensemble de données contient 2126 lignes, donc la suppression de valeurs manquantes ne nous fait pas perdre beaucoup de données. De plus, la suppression de ces enregistrements avec des valeurs manquantes est justifiée car les données médicales doivent être précises et complètes.

### III. Analyse des données graphique

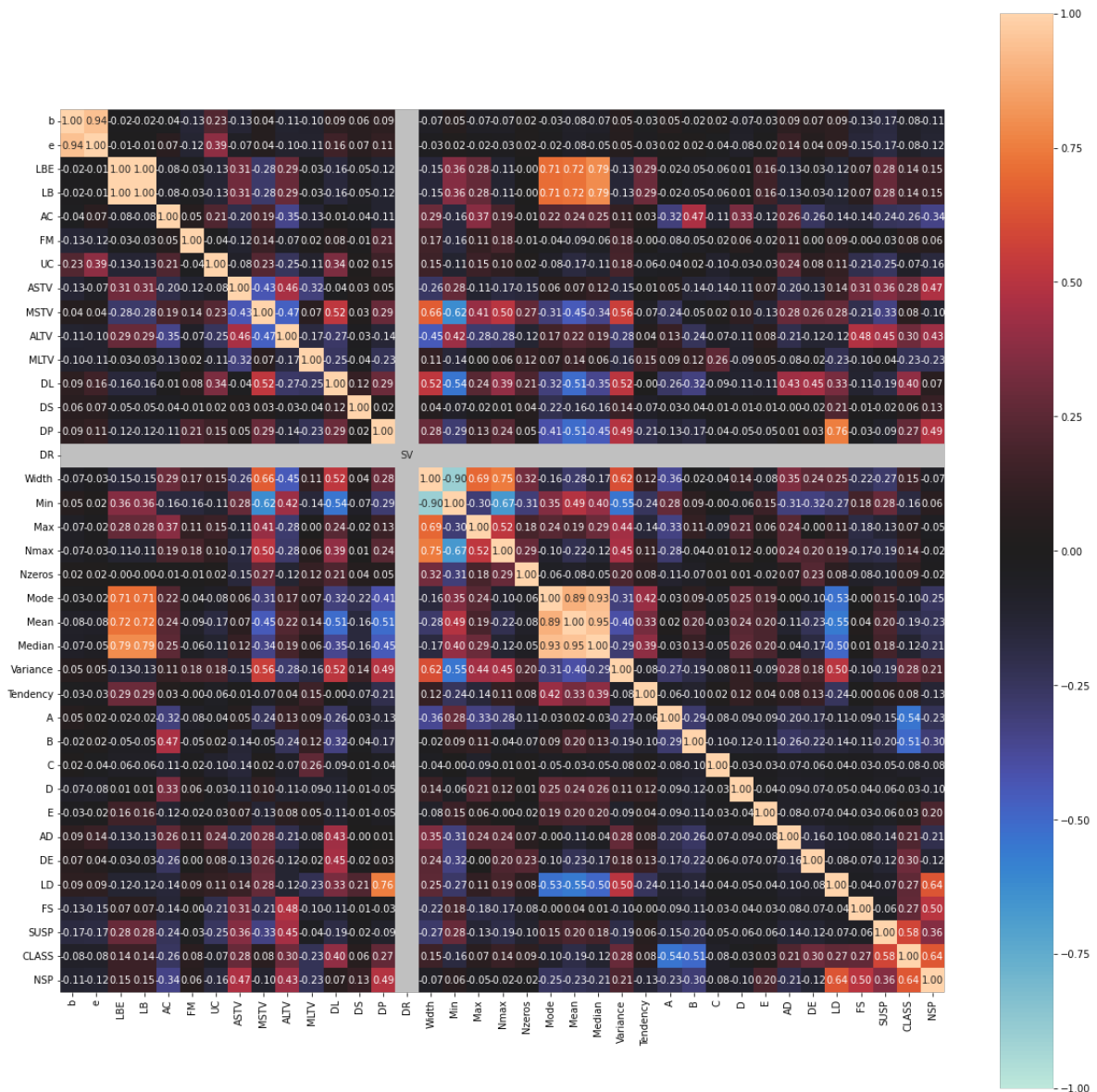


Figure 2 : la matrice de corrélation

La figure 2 représente une matrice de corrélation entre différentes caractéristiques (ou variables) d'un ensemble de données. Chaque caractéristique est représentée à la fois en tant que ligne et en tant que colonne.

Les valeurs dans la matrice indiquent la force de la corrélation entre chaque paire de caractéristiques. Les valeurs sont comprises entre -1 et 1. Une valeur de 1 indique une corrélation positive parfaite (c'est-à-dire que les deux caractéristiques varient exactement de la même manière), une valeur de -1 indique une corrélation négative parfaite (les deux caractéristiques varient en sens opposé), et une valeur de 0 indique qu'il n'y a pas de corrélation entre les deux caractéristiques.

Par exemple, la corrélation entre la caractéristique ASTV et la caractéristique MSTV est de -0,430705, ce qui suggère une corrélation négative modérée entre ces deux

caractéristiques. Cela signifie que lorsque la caractéristique ASTV augmente, la caractéristique MSTV tend à diminuer, et vice versa.

Cette matrice de corrélation est souvent utilisée pour identifier les caractéristiques les plus importantes ou les plus pertinentes pour une analyse ultérieure, ainsi que pour détecter les relations et les modèles dans les données.

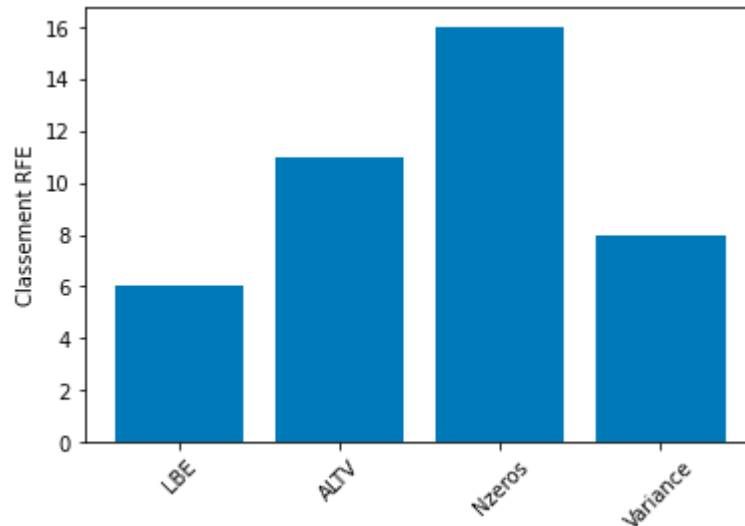
## **IV. Apprentissage de nos modèles**

### **1. SVM (Support Vector Machine):(Sarah)**

Dans cette partie, nous allons étudier l'utilisation des machines à vecteurs de support (SVM) pour la classification des enregistrements CTG en fonction de leur état fœtal. Les SVM sont un type de modèle d'apprentissage supervisé qui peut être utilisé pour la classification binaire et multiclasse. L'objectif des SVM est de trouver un hyperplan dans un espace de grande dimension qui sépare les différentes classes de manière optimale.

Pour entraîner notre modèle SVM, nous avons séparé notre jeu de données en deux parties : une partie pour l'entraînement et une partie pour la validation. Nous avons utilisé la méthode GridSearchCV de scikit-learn, qui permet d'ajuster les hyperparamètres du modèle en utilisant la technique de validation croisée. Cette méthode nous a permis d'explorer différentes combinaisons de valeurs pour les paramètres du modèle et de trouver les valeurs optimales qui ont produit les meilleures performances.

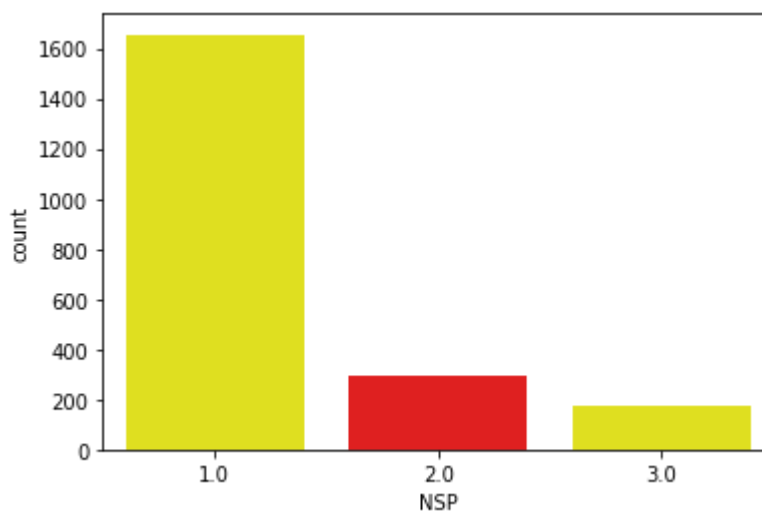
Nous avons construit deux modèles de classification SVM pour prédire l'état fœtal à partir des données CTG : le premier modèle utilise toutes les variables disponibles dans le jeu de données, tandis que le deuxième modèle utilise une sélection de variables pertinentes obtenues à l'aide de la technique RFE.



**Figure 3 :** Classement RFE des variables sélectionnées

Après avoir sélectionné les variables pertinentes à l'aide de l'algorithme RFE, nous avons obtenu les noms des variables sélectionnées. Nous avons ensuite créé un diagramme en barres pour visualiser les variables sélectionnées. (figure 3) le diagramme montre que les variables les plus importantes sont 'LBE', 'ALTV', 'Nzeros' et 'Variance', dans cet ordre. Ces résultats sont cohérents avec la littérature médicale existante, qui suggère que ces variables ont un impact significatif sur la détection de cardiocardiographie fœtale.

En termes de performance, les deux modèles ont produit des résultats similaires en termes d'exactitude (accuracy) avec des scores respectifs de 0,97 et 0,99.



**Figure 4 :** Répartition des classes dans l'ensemble de données NSP

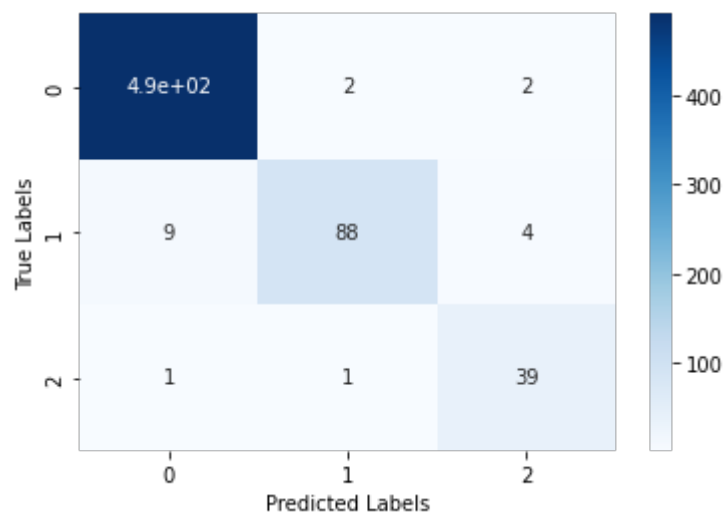
D'après la figure 4 dans notre ensemble de données, la classe 1 est beaucoup plus grande que les classes 2 et 3. Cela signifie que si nous évaluons notre modèle uniquement sur la base de l'exactitude (accuracy), il peut être trompeur car même un modèle qui prédit constamment la classe 1 aura une bonne exactitude en raison de la prédominance de cette classe.

En effet, le choix de mesures d'évaluation telles que la précision, le rappel et le score F1 est directement lié à la nature des données médicales, où la détection des cas positifs (par exemple, les états fœtaux anormaux) est plus critique que la précision globale. Le rappel est donc une mesure importante pour évaluer la performance des modèles de classification dans les données médicales.

Le premier modèle a produit un rappel de 0,97, ce qui signifie que le modèle a correctement identifié 97% des états fœtaux anormaux dans l'ensemble de données. Le deuxième modèle a produit un rappel de 0,99, ce qui est une amélioration significative par rapport au premier modèle.

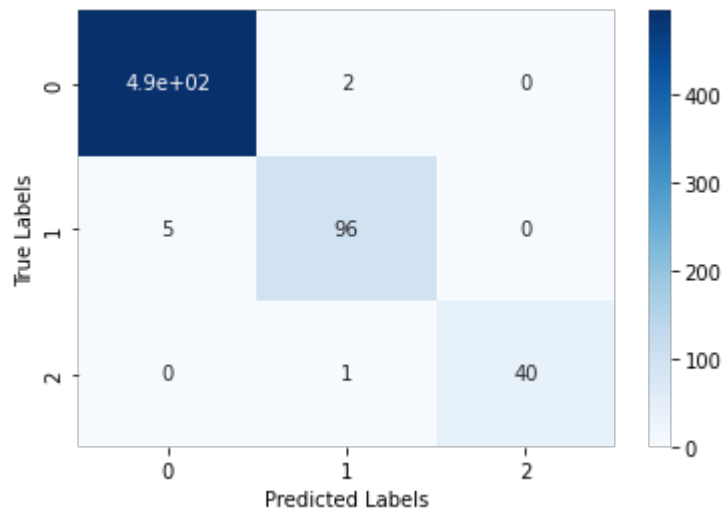
En ce qui concerne la précision (precision), le premier modèle a produit une précision de 0,97, tandis que le deuxième modèle a produit une précision de 0,99. La précision mesure la capacité du modèle à ne pas classer à tort des cas négatifs comme positifs.

Le score F1 est une mesure combinée de la précision et du rappel, donnant une moyenne pondérée de ces deux scores. Dans notre cas, le premier modèle a produit un score F1 de 0,97, tandis que le deuxième modèle a produit un score F1 de 0,99.



**Figure 5 :** Matrice de confusion du modèle de classification de base





**Figure 6:** Matrice de confusion du modèle de classification avec sélection de variables

En examinant les résultats de la matrice de confusion pour les deux modèles, comme on le voit dans la figure 6 et 5 nous pouvons voir que le deuxième modèle (avec sélection de variables) a produit moins de faux positifs et de faux négatifs que le premier modèle. Cela suggère que la sélection de variables a été efficace pour améliorer la performance du modèle.

Cependant, il est important de noter que même le deuxième modèle a produit quelques faux négatifs, ce qui pourrait avoir des conséquences graves pour le patient si cela se produisait en pratique. Ces cas devraient être examinés de plus près pour comprendre pourquoi le modèle les a mal classés et comment améliorer sa performance à l'avenir.

**En conclusion**, nos résultats montrent que la classification SVM est une méthode efficace pour prédire l'état fœtal à partir des données CTG. La sélection de variables pertinentes peut considérablement améliorer les performances des modèles de classification SVM en réduisant les dimensions des données et en éliminant les variables non pertinentes.

Cependant, il est important de noter que la nature des données médicales nécessite une attention particulière à la performance de détection des cas positifs (états fœtaux anormaux). Ainsi, le rappel est une mesure plus pertinente à considérer que l'exactitude globale du modèle pour s'assurer que les cas d'états fœtaux anormaux ne sont pas manquants.

En outre, les résultats de nos modèles de classification SVM montrent une faible incidence de faux positifs et de faux négatifs, ce qui suggère que ces modèles pourraient être utiles en pratique clinique. Cependant, des études supplémentaires sont nécessaires pour valider la performance des modèles sur des données de patients supplémentaires et pour examiner comment ces modèles peuvent être

utilisés de manière appropriée en pratique clinique pour aider à améliorer les résultats de santé maternelle et fœtale.

## 2. Régression logistique :(Lilia)

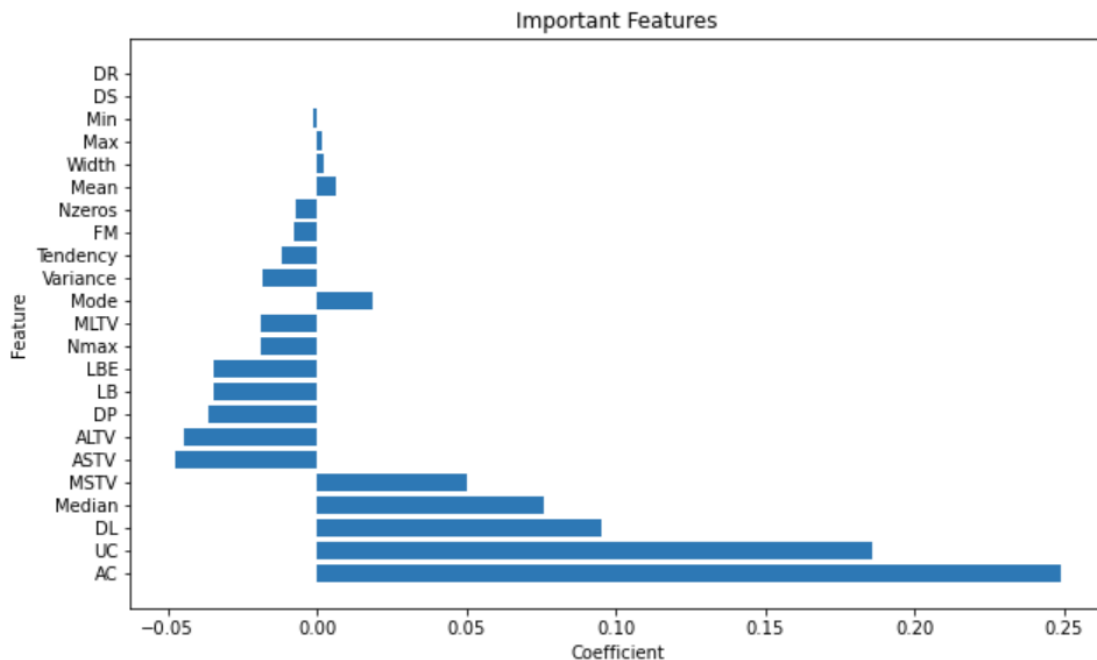
La régression logistique est adaptée à notre problématique vu que c'est une multi classification en se basant sur des variables numériques. L'implémentation est faite en python en utilisant le package **Sklearn**.

On utilise 70% de nos données pour la détermination des paramètres W et 30% pour le test.

Les coefficients de régression dans un modèle de régression logistique indiquent la force et la direction de l'association entre chaque variable indépendante et la variable cible. Les coefficients positifs indiquent une association positive, tandis que les coefficients négatifs indiquent une association négative. Les coefficients plus élevés indiquent une association plus forte.

Les caractéristiques les plus importantes peuvent être identifiées en examinant les coefficients de régression les plus élevés en valeur absolue.

### 2.1 Régression logistique non pénalisée



**Figure 7 :**les coefficients de la régression

En examinant ce graphique on peut déduire les top 10 caractéristiques.

- AC:0.248823

- UC:0.185633
- DL:0.095469
- Median :0.075973
- MSTV:0.050354
- ASTV:-0.047647
- ALTV:-0.044471
- DP:-0.036425
- LB:-0.034781
- LBE:-0.034781

Le premier modèle a produit les performances suivantes :

Confusion Matrix

```
[[466  20  10]
 [ 38  56   7]
 [   3   5  33]]
```

Balanced Accuracy 0.7662832077857669

Recall Accuracy Score~TP 0.7662832077857669

Precision Score ~ Ratio of TP 0.7568300581975795

F1 Score 0.7566239011902023

Un recall de 0.76 pour une classification à 3 classes peut indiquer que le modèle est capable d'identifier correctement 76 % des true exemples positifs. Cela peut être considéré comme un bon résultat, mais cela dépend également de la gravité de la maladie ou de l'état du patient.

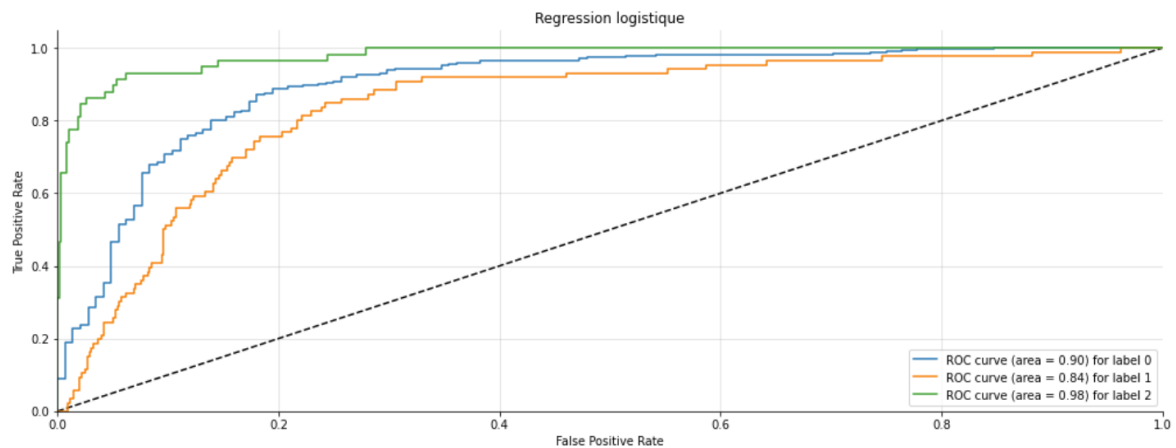
Une précision de 0,76 pour une classification à 3 classes peut indiquer que le modèle est capable d'identifier correctement 76 % des exemples positifs

Le F1-score est une mesure de la performance d'un modèle de classification qui tient compte à la fois de la précision et du rappel. Il représente la moyenne harmonique de ces deux mesures .

Un F1-score de 0.75 peut être considéré comme une performance raisonnable du modèle de classification à 3 classes dans le domaine médical, mais cela doit être interprété en tenant compte de plusieurs autres facteurs importants.

Dans cette problématique on priorise la bonne classification de la classe 3 qui représente une pathologie ,pour cela on utilise AUC .

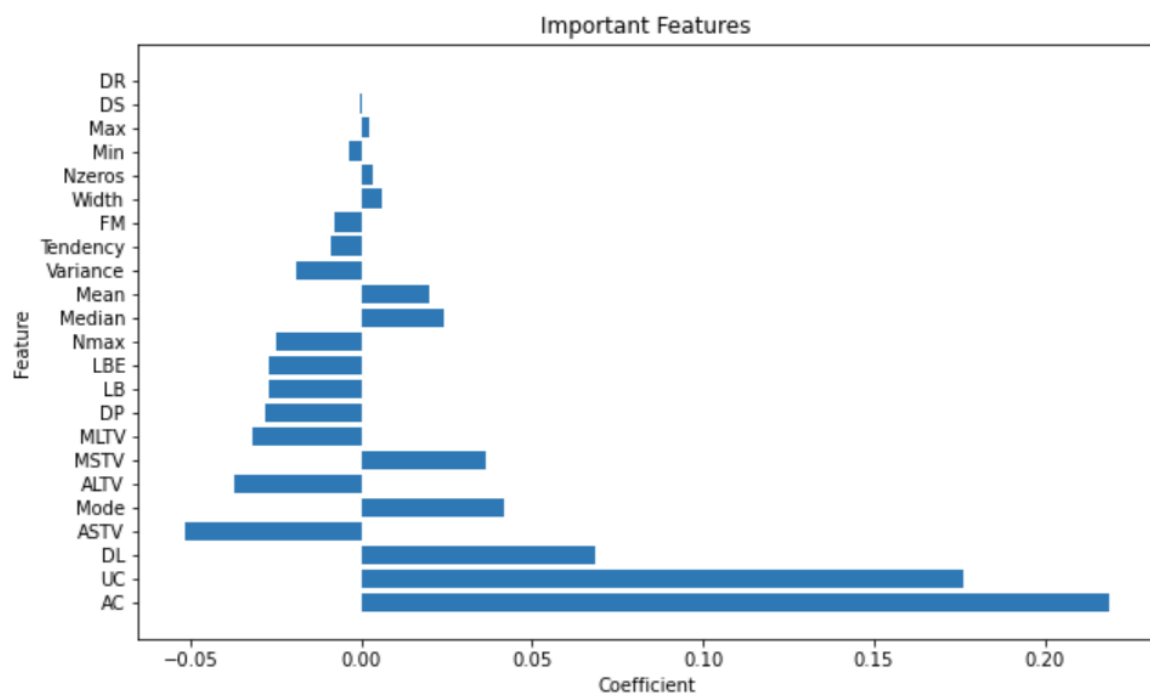
L'aire sous la courbe ROC (AUC) OVA est une mesure de performance utilisée pour évaluer la performance d'un modèle de classification multi-classes utilisant la méthode OVA. Elle est calculée en combinant les courbes ROC pour chaque classe individuelle. Pour chaque classe, le modèle binaire entraîné contre cette classe est utilisé pour prédire si l'exemple appartient ou non à cette classe. Ensuite, la courbe ROC est tracée en utilisant les scores de ces prédictions pour cette classe spécifique.



On remarque que ce modèle détecte très bien la classe 3 (label 2 sur le graphique) on a un AUC de 0.98

## 2.2 Régression logistique ridge

on régularise notre régression logistique avec norme  $l_2$  et on a eu les résultats suivants .



**Figure 8:** les coefficients de la régression ridge

En examinant ce graphique on peut déduire les top 10 caractéristiques.

- AC:0.218671
- UC:0.175956
- DL:0.068566
- ASTV:-0.051711
- Mode :0.041956
- ALTV:-0.037325
- MSTV:0.036673
- MLTV:-0.031543
- DP:-0.028260
- LB: -0.027017

En comparant avec la régression logistique non pénalisée y a quelques caractéristiques en commun mais avec une importance différente .

Le deuxième modèle a produit les performances suivantes :

Confusion Matrix

```
[[466  20  10]
 [ 38  56   7]
 [   3   5  33]]
```

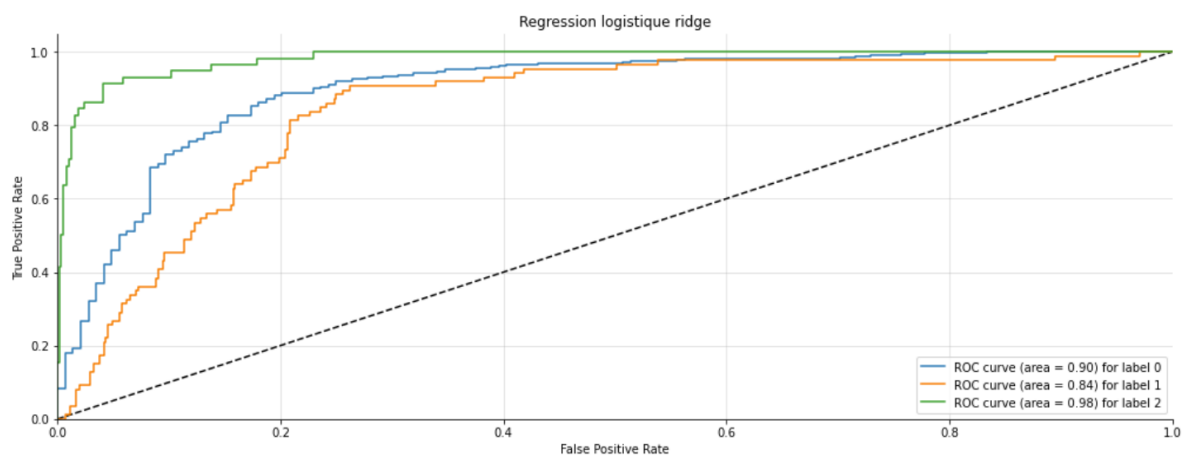
Balanced Accuracy 0.7662832077857669

Recall Accuracy Score~TP 0.7662832077857669

Precision Score ~ Ratio of TP 0.7568300581975795

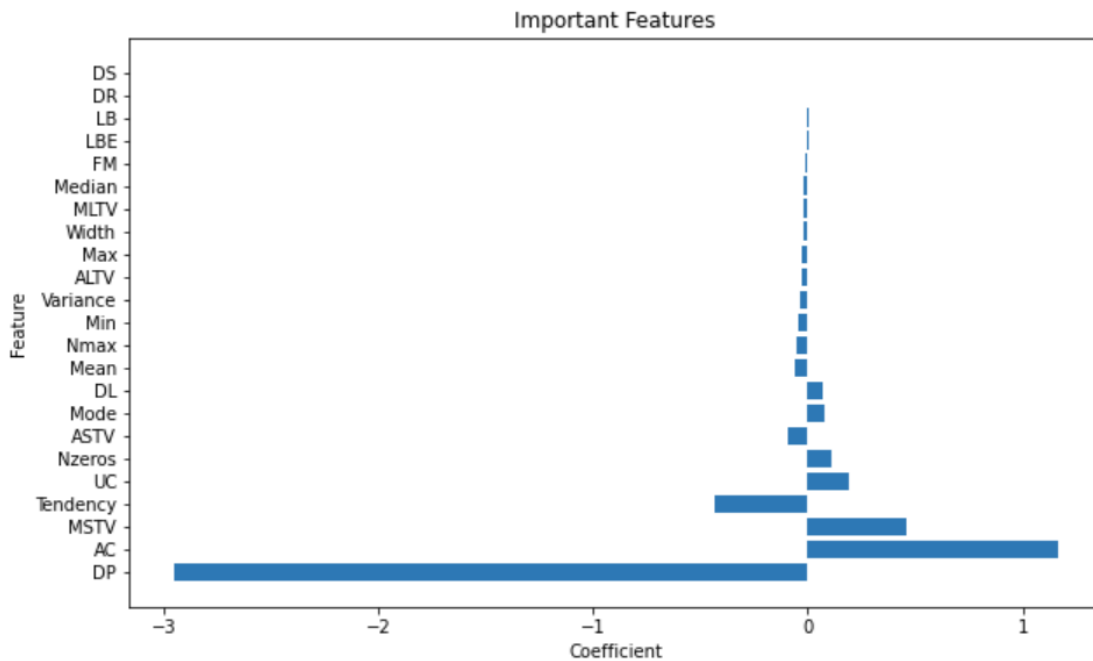
F1 Score 0.7566239011902023

On a eu les même performances que le premier modèle malgré que ce modèle ne donne pas les même importances aux caractéristiques , idem pour ROC



## 2.2 Régression logistique Lasso:

on régularise notre régression logistique avec norme l1 et on a eu les résultats suivants



**Figure 9** :les coefficients de la régression lasso

En examinant ce graphique on peut déduire les top 10 caractéristiques.

- DP : -2.956064
- AC : 1.164626
- MSTV : 0.460363
- Tendency: -0.438049
- UC : 0.189881
- Nzeros: 0.107981
- ASTV : -0.089469
- Mode : 0.076904
- DL : 0.071182
- Mean: -0.056971

En comparant avec les autres modèles y a quelques caractéristiques en commun mais avec une importance différente .

Le troisième modèle a produit les performances suivantes :

Confusion Matrix

```
[[471  23   2]
 [ 27  61  13]
 [   3   4  34]]
```

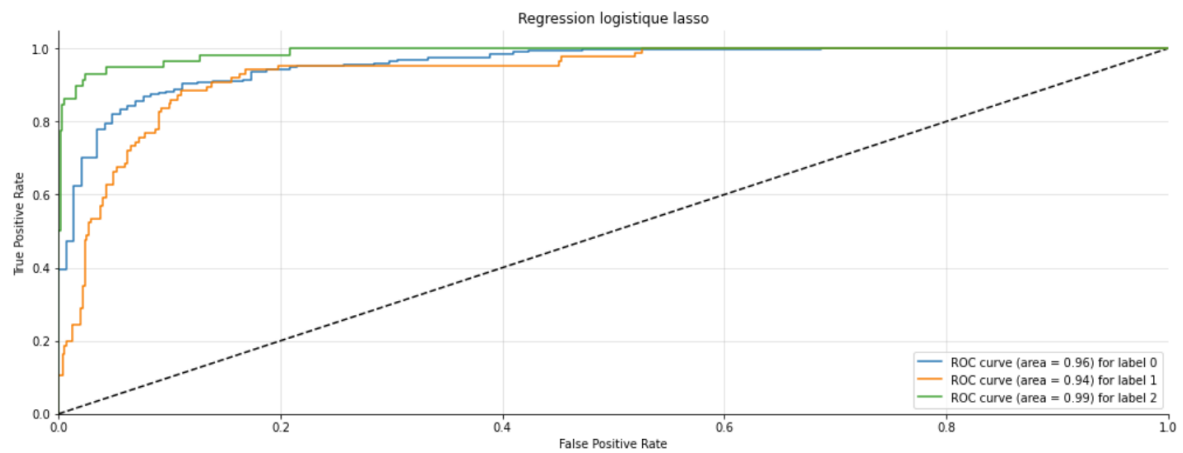
Balanced Accuracy 0.7942751543053598

Recall Accuracy Score~TP 0.7942751543053598

Precision Score ~ Ratio of TP 0.7757263765604226

F1 Score 0.7819642348562442

Ce modèle donne les meilleures performances par rapport aux autres modèles ,on le conforme avec ROC .



AUC de 0.99 pour la classe 3 qui représente la pathologie ce qui est parfait !

## 2.4 L'ensemble learning

L'ensemble learning avec des régressions logistiques peut être réalisé en utilisant des techniques telles que le bagging, et le boosting et le stacking.

On définit un modèle de base, qui est une régression logistique lasso vu qu' elle produit les meilleures performances . Ensuite, nous avons créé un modèle de sac (bagging) en utilisant le modèle de base et en spécifiant le nombre d'estimateurs et une graine aléatoire pour la reproductibilité.

Le nombre d'estimateurs a été défini avec une cross validation et on a 150 estimateurs .

Ce que donne les résultats suivant :

**Confusion Matrix**

```
[[470  21   3]
 [ 31  54   1]
 [   2   8  48]]
```

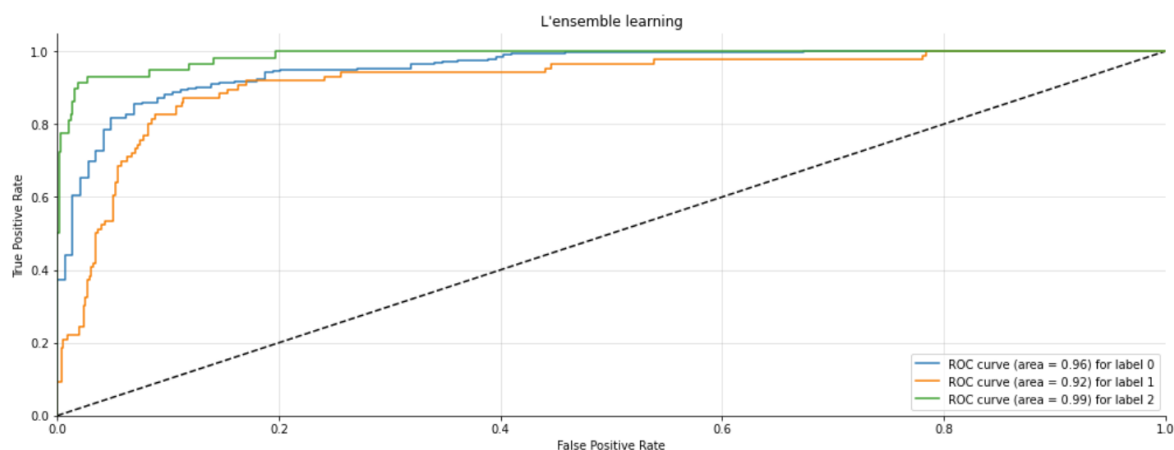
**Balanced Accuracy 0.8023033958964403**

**Recall Accuracy Score~TP 0.8023033958964403**

**Precision Score ~ Ratio of TP 0.8360243236288172**

**F1 Score 0.8182030042071705**

On remarque une amélioration pour toutes les métriques et concernant ROC on a eu ceci



Toujours une parfaite détection de la classe qui représente la pathologie en premier lieu puis la classe 1 Normal et enfin la classe 2 Suspect .

Ce serait mieux si notre modèle détectait la classe Suspect plus efficacement que la classe Normale .



**En conclusion**, nos résultats montrent que la classification RL est une méthode efficace pour prédire l'état fœtal à partir des données CTG malgré qu'on a un jeu de données déséquilibré où la classe état fœtal représente la minorité .

Avec la pénalisation avec la norme l1 on a pu améliorer les résultats et pour encore les améliorer on a combiné 150 régressions.

Ce modèle assure que les cas d'états fœtaux anormaux ne sont pas ratés.

### 3. Random Forest & Decision Tree (Manel)

Afin de sélectionner les variables les plus pertinentes pour l'entraînement des modèles

**Random Forest** et **Decision Tree**, une analyse de l'importance des caractéristiques

**(Feature importance)** est effectuée. Cette méthode permet de sélectionner les

caractéristiques les plus pertinentes à inclure dans le modèle et donc réduire la

dimensionnalité du vecteur caractéristique. Toute variable avec un score d'importance de 0 ou moins est abandonnée.

```
Nzeros : importance : 0.0
DS : importance : 0.0
DR : importance : 0.0
Nmax : importance : 0.0006750037692308345
Max : importance : 0.001270074277581104
Tendency : importance : 0.0013407871135868803
FM : importance : 0.001946439965477985
DL : importance : 0.004755408661196924
UC : importance : 0.013262015883010798
MLTV : importance : 0.01564026698752758
Width : importance : 0.02181174647156448
Min : importance : 0.02322290603055076
LBE : importance : 0.028462053036451107
Mode : importance : 0.03551260185121311
LB : importance : 0.036937814835363694
Variance : importance : 0.03957388491270836
Median : importance : 0.04323590888687447
DP : importance : 0.07775598239277838
AC : importance : 0.08103725976362756
Mean : importance : 0.1132810385369712
ALTV : importance : 0.1280730642896806
ASTV : importance : 0.15510000134301422
MSTV : importance : 0.17710574099159018
```

**Figure 9** : Les scores de l'importance des features

D'après la figure 9 , Les variables suivantes sont donc supprimées : ['Nzeros', 'DS', 'DR'], et on se retrouve donc avec 19 variables indépendantes (features) au lieu des 41 initiales.

### 3.1 Random Forest

Dans cette partie, nous allons étudier l'utilisation du Random Forest pour la classification des enregistrements CTG en fonction de leur état foetal.

La Random Forest ou Forêt Aléatoire est une méthode d'apprentissage supervisé qui combine plusieurs arbres de décision pour améliorer la précision de la prédiction. Chaque arbre de décision dans la forêt aléatoire est construit sur un échantillon aléatoire du jeu de données et utilise un sous-ensemble aléatoire des attributs.

Nous avons entraîné une forêt aléatoire sur notre jeu de données après l'étude de la feature importance pour déterminer les variables indépendantes les plus pertinentes pour la classification que nous voulons effectuer. Après quoi, nous avons effectué une grid search de scikit-learn afin d'ajuster au mieux les hyperparamètres du modèle Random Forest tout en comparant les résultats de l'accuracy et la précision à chaque étape avec la précédente grâce au concept de cross validation ou validation croisée.

Nous avons donc construit deux modèles de classification RF pour prédire l'état foetal à partir des données CTG : le premier modèle utilise toutes les variables disponibles dans le jeu de données, tandis que le deuxième modèle utilise une sélection de variables pertinentes obtenues à l'aide de la technique Feature Importance.

En termes de performance, les deux modèles ont produit des résultats similaires en termes d'exactitude (accuracy) avec des scores respectifs de 0,84 et 0,95.

La cross validation, avant la feature importance a produit une accuracy moyenne de 0.84 avec une précision moyenne de 0.86 contre une accuracy moyenne de 0.86 et une précision moyenne de 0.85.

Nous avons donc choisi le modèle avec la sélection de variables pertinentes, et nous lui avons appliqué la grid search qui a permis de déterminer le meilleur modèle avec les meilleurs Hyperparamètres pour la Random Forest. On note une accuracy moyenne de 0.95 et une précision moyenne de 0.92.

Le meilleur modèle a produit un rappel de 0,88, ce qui signifie que le modèle a correctement identifié 88% des états fœtaux anormaux dans l'ensemble de données. Le deuxième modèle.

En ce qui concerne la précision (precision), le meilleur modèle a produit une précision de 0,93,

Le score F1 est une mesure combinée de la précision et du rappel, donnant une moyenne pondérée de ces deux scores. Dans notre cas, le modèle a produit un score F1 de 0,91.

### 3.2 Decision Tree

Dans cette partie, nous allons étudier l'utilisation du Decision Tree pour la classification des enregistrements CTG en fonction de leur état fœtal.

L'arbre de décision ou le Decision Tree est un modèle d'apprentissage automatique qui est utilisé pour la classification et la prédiction. Il s'agit d'un modèle qui se présente sous la forme d'un arbre, où chaque nœud représente une caractéristique (ou variable) et chaque branche représente une règle de décision basée sur cette caractéristique. Le processus de construction d'un arbre de décision commence par la sélection de la caractéristique la plus importante pour la classification du jeu de données. Cette caractéristique est utilisée pour diviser les données en deux groupes distincts. Le processus est répété pour chaque groupe jusqu'à ce que toutes les données soient classées.

Nous avons entraîné un arbre de décision sur notre jeu de données après l'étude de la feature importance pour déterminer les variables indépendantes les plus pertinentes pour la classification que nous voulons effectuer. Après quoi, nous avons effectué une grid search de scikit-learn afin d'ajuster au mieux les hyperparamètres du modèle Decision Tree (DT) tout en comparant les résultats de l'accuracy et la précision à chaque étape avec la précédente grâce au concept de cross validation ou validation croisée.

Nous avons donc construit directement le modèle de classification DT pour prédire l'état fœtal à partir des données CTG sur la sélection de variables pertinentes obtenues à l'aide de la technique Feature Importance.

En termes de performance, le modèle a produit des résultats satisfaisants en termes d'exactitude (accuracy) de précision avec des scores respectifs de 0,92 et 0,88.

Nous avons ensuite effectué la grid search qui a permis de déterminer le meilleur modèle avec les meilleurs Hyperparamètres pour la Decision Tree. On note une accuracy moyenne de 0.92 et une précision moyenne de 0.90.

Le modèle a produit un rappel de 0,83, ce qui signifie que le modèle a correctement identifié 83% des états fœtaux anormaux dans l'ensemble de données. Le deuxième modèle.

En ce qui concerne la précision (precision), le modèle a produit une précision de 0,89.

Le score F1 est une mesure combinée de la précision et du rappel, donnant une moyenne pondérée de ces deux scores. Dans notre cas, le modèle a produit un score F1 de 0,86.

**En conclusion**, nos résultats montrent que les classifications RF et DT sont toutes deux efficaces pour prédire l'état fœtal à partir des données CTG. La sélection de variables pertinentes peut considérablement améliorer les performances des

modèles de classification RF et DT en réduisant les dimensions des données et en éliminant les variables non pertinentes.

Cependant, nous remarquons que le Modèle Random Forest est plus adapté à notre étude étant donné qu'il produit de meilleurs résultats que la Decision Tree en termes d'accuracy, de précision, de rappel et même en termes de Score F1.

#### 4. ANN (Ahmed)

On a choisi d'utiliser le modèle ANN pour sa capacité à réaliser des classifications précises. Pour l'implémentation de ce modèle, nous avons utilisé la bibliothèque scikit-learn, qui offre une interface simple et efficace pour la création de modèles de réseau de neurones. Afin d'optimiser les performances du modèle, nous avons utilisé la méthode de grid search pour choisir les meilleurs hyperparamètres.

Pour la recherche des meilleurs hyperparamètres, nous avons défini une grille de paramètres à tester. Cette grille comprenait plusieurs combinaisons de valeurs pour les hyperparamètres suivants :

- `hidden_layer_sizes` : nous avons testé plusieurs configurations de couches cachées, y compris des architectures à une ou plusieurs couches, avec des nombres variables de neurones dans chaque couche.

```
[ (10, 30, 10), (20, ), (300, 100), (50, 50), (200, 100), (100, 100), (50, 100),  
(100, 200) ]
```

- `activation` : nous avons testé deux fonctions d'activation couramment utilisées dans les réseaux de neurones, à savoir la fonction tangente hyperbolique (`tanh`) et la fonction de rectification linéaire (`ReLU`).
- `solver` : nous avons testé trois algorithmes de descente de gradient différents pour l'optimisation du réseau de neurones, y compris `lbfgs`, `sgd` et `adam`.
- `alpha` : nous avons testé trois valeurs différentes pour le paramètre de régularisation `alpha`, qui contrôle la complexité du modèle 0.0001, 0.05, 0.1.
- `learning_rate` : nous avons testé deux stratégies différentes pour la mise à jour des poids du réseau de neurones pendant l'apprentissage, à savoir une stratégie de taux d'apprentissage constant et une stratégie adaptative basée sur l'erreur de validation.

Après avoir effectué une recherche exhaustive des hyperparamètres en utilisant la méthode `GridSearchCV`, nous avons identifié les meilleurs hyperparamètres pour notre modèle ANN.

Les hyperparamètres optimaux étaient les suivants :

- `activation` : `tanh`
- `alpha` : 0.1
- `hidden_layer_sizes` : (300, 100)
- `learning_rate` : `adaptive`
- `solver` : `adam`

Les meilleurs scores obtenus pour notre modèle ANN, en utilisant les hyperparamètres optimaux identifiés par la méthode GridSearchCV, sont les suivants :

- Précision (precision) : 0.828
- Rappel (recall) : 0.834
- F1-score (f1) : 0.806
- Exactitude (accuracy) : 0.910

## 5. Conclusion générale

	Accuracy	F1score	Recall	Precision
random forest	0.952978	0.906540	0.882247	0.934856
arbre de decision	0.926332	0.865162	0.838886	0.896652
svm	0.987461	0.987405	0.987461	0.987413
ANN	0.909688	0.806077	0.834094	0.828329
RL	0.743756	0.757576	0.743756	0.775113
Ridge	0.766283	0.756624	0.766283	0.756830
Lasso	0.794275	0.781964	0.794275	0.775726
EnsembleofRL	0.802303	0.818203	0.802303	0.836024

Selon les différents résultats obtenus, il est clair que le modèle le plus performant pour notre cas est celui du SVM, en termes des quatre critères que nous avons fixés : L'accuracy, la précision, le score F1 et le recall.

## 6.Synthèses :

**Lilia:** j'ai pas rencontré des difficultés particulières en appliquant la régression logistique sur cette problématique cependant souvent lorsque on applique des modèles de Machine Learning sur des données déséquilibrées, cela peut poser des problèmes car le modèle peut apprendre à prédire la classe majoritaire de manière prédominante, en ignorant la classe minoritaire. Cela peut conduire à une performance de prédiction biaisée et peu fiable pour la classe minoritaire, c'est étonnant que nos modèles arrivent à détecter la classe minoritaire qui représente la pathologie .

**Manel:** J'ai choisi d'appliquer la Random Forest et l'Arbre de Décision car la différence entre ces deux méthodes de classification m'étaient floues et je n'arrivais pas à faire la différence entre les deux concepts. Après les avoir appliqué sur les enregistrements de cardiogrammes fœtaux, j'ai non seulement pu éclaircir ce point mais aussi pu découvrir l'importance de la réduction de la dimensionnalité du vecteur caractéristique pour un problème de classification. Je n'ai cependant pas rencontré de difficultés majeures.

**Ahmed:** En utilisant les réseaux de neurones pour la classification, j'ai réussi à améliorer mes compétences en apprentissage automatique. En explorant les différentes architectures de réseau et les paramètres avec la bibliothèque sklearn, j'ai découvert de nouvelles fonctionnalités qui m'ont aidé à optimiser les performances de notre modèle de classification. Cette expérience m'a également permis de perfectionner mes compétences en résolution de problèmes liés à l'apprentissage automatique, et ce, sans rencontrer de difficultés majeures.

**Sarah:**

J'ai choisi d'utiliser le SVM pour la classification de données médicales. J'ai découvert que cette méthode était très efficace pour classer les données en fonction de leur caractéristique. J'ai dû rencontrer un défi majeur lorsque j'ai réalisé que mes données étaient déséquilibrées, le fait d'avoir un grand nombre de données pour une classe et un petit nombre de données pour l'autre classe. J'ai ensuite appris qu'il était important de prendre en compte les déséquilibres de classe lors de l'entraînement des modèles de classification.

J'ai également rencontré des problèmes de faux positifs et de faux négatifs. J'ai appris à évaluer la performance de mon modèle à l'aide de la matrice de confusion et à ajuster les paramètres du modèle pour minimiser les faux positifs et les faux négatifs.

Cette expérience m'a permis de mieux comprendre l'importance de la gestion des déséquilibres de classe et des faux positifs et négatifs dans les modèles de classification, et que dans le contexte médical il faut souvent minimiser les faux négatifs pour éviter les conséquences graves pour la santé des patients et minimiser aussi les faux positifs pour ne pas avoir des traitements inutiles pour les patients, j'ai appris aussi que l'accuracy n'est pas toujours la meilleure mesure de performance quand on a des déséquilibres de classe.