UNIVERSITÉ TOULOUSE III PAUL SABATIER
Université de Toulouse
Faculté Sciences et Ingénierie

Master 1 Intelligence Articielle

# Mini-Projet : classification de battements cardiaques


***Auteurs :***
DJIGUINE Mamady
DIALLO Mohamed

***Chargé du cours :***
Mme MOUYSSET Sandrine
*Encadrant TP :*
MR PELLEGRINI Thomas


17 décembre 2023

# Table des matières

# 1 Introduction

Le projet vise à classifier les bruits de battements cardiaques à partir d'enregistrements de deux sources distinctes : (A) collectés via une application de smartphone auprès du grand public et (B) recueillis dans le cadre d'un essai clinique avec l'utilisation d'un stéthoscope numérique. Les enregistrements ont été transformés en Mel-Frequency Cepstral Coefficients (MFCC) pour extraire le contenu fréquentiel.

## 1.1 Chargement des données

A chaque enregistrement, 20 coefficients MFCC sont calculés en réalisant la moyenne sur chaque fenêtre de 10ms. DataMFCC.csv regroupe tous les enregistrements des 2 dispositifs. Le fichier source DataMFCC.csv sur lequel vous travaillez est la conversion des enregistrements audio en matrice de paramètres appélés MFCC (Mel Frequency Cepstral Coefficient) en utilisant la librairie python librosa. Ces paramètres permettent d'extraire au mieux le contenu vocal fréquenciel du signal audio. La matrice de données est composée d'autant de vecteurs lignes que de fichiers audio. Le nombre de colonnes correspond à la dimension du vecteur moyen représentatif des MFCC : ici 20.(tiré de l'enoncé)

## 1.2 Prétraitement des Données

Aucune réduction de dimension (PCA) n'a été appliquée, mais une normalisation des données a été effectuée pour garantir une mise à l'échelle uniforme entre les différentes sources.

## 1.3 Évaluation des Performances

Les performances de chaque modèle ont été évaluées à l'aide de métriques telles que l'accuracy, le rapport de classification et la matrice de confusion. Ces mesures permettent de comprendre la capacité de chaque modèle à classifier correctement les différentes classes de battements cardiaques.

# 2 Méthodes d'apprentissage utilisées

## 2.1 Méthodes Supervisées :

Les méthodes supervisées nécessitent l'utilisation d'un ensemble d'apprentissage étiqueté pour former le modèle. En ce qui concerne nos choix de méthodes supervisées, nous avons choisi d'utiliser les approches suivantes :

- **l'approches SVM (Machine Vecteur Support)**
- **l'approches Random Forest**
- **l'approches k-NN (k plus proches voisins)**

## 2.2 Méthodes Non Supervisées :

Dans le cadre des méthodes non supervisées, nous avons opté pour l'utilisation de K-Means. Les méthodes non supervisées visent à regrouper les données sans la connaissance des classes réelles.

# 3 Etude sur les paramètres inhérents à la méthode supervisée et non supervisée

## 3.1 Machine Vecteur Support (SVM) :

Le modèle SVM a été entraîné avec deux noyaux différents : linéaire et RBF. Le noyau linéaire est efficace lorsque la relation entre les caractéristiques et les classes est linéaire, tandis que le noyau RBF est plus flexible pour capturer des relations non linéaires.

### 3.1.1 Description des Parametres :

L'étude des paramètres pour le SVM implique généralement l'ajustement de deux paramètres principaux :

1. **C (paramètre de régularisation) :**
   — Des valeurs élevées de C entraînent un modèle plus complexe, qui pourrait potentiellement surajuster les données d'entraînement.
   — Des valeurs faibles de C conduisent à un modèle plus tolérant envers les erreurs d'entraînement, mais pourraient manquer de généralisation.
2. **Gamma (pour le noyau RBF) :**
   — Un gamma élevé signifie que le modèle accorde plus de poids aux exemples d'entraînement les plus proches, créant des frontières de décision plus complexes.
   — Un gamma faible permet à l'influence des exemples d'entraînement de s'étendre sur une plus grande distance, conduisant à des frontières de décision plus lisses.

### 3.1.2 Méthodologie :

Pour déterminer les meilleurs hyperparamètres du modèle SVM, nous avons utilisé une approche de recherche systématique avec GridSearchCV. Cette technique explore différentes combinaisons de noyaux (linéaire, RBF), de valeurs de régularisation (C), et de paramètres du noyau RBF (gamma). L'objectif était de trouver la combinaison qui maximise les performances du modèle en termes de précision, de rappel et de score F1.

Le GridSearchCV a parcouru une grille prédéfinie de valeurs pour C et gamma, évaluant chaque combinaison à l'aide d'une validation croisée. Pour le noyau RBF, cela a permis de rechercher la meilleure combinaison entre la souplesse du noyau et la régularisation. Les paramètres optimaux ont été sélectionnés en fonction des scores moyens obtenus lors de la validation croisée.

```
# Définition des valeurs à tester pour C et gamma
param_grid = [ {'kernel':['rbf'],'gamma': [0.001,0.1, 1, 10, 100],'C': [0.001,0.1, 1, 10, 100]},
               {'kernel': ['linear'],'C': [0.001,0.1, 1, 10, 100]}
             ]
```

```
# Approche 1: On va entrainer le modele avec les données filtré de  A(X_train_f_setA)
# et tester avec les données de test de B (X_test_setB)
# On y procède comme suit

from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score
from sklearn.metrics import f1_score
import seaborn as sns


# Création d'un modèle SVM
svm_model_1 = SVC()

# Utilisation de GridSearchCV pour trouver les meilleurs paramètres
grid_search_1 = GridSearchCV(svm_model_1, param_grid, cv=5, scoring='accuracy')
grid_search_1.fit(X_train_f_setA, y_train_f_setA)

# Affichage des meilleurs paramètres trouvés
print("Meilleurs paramètres trouvés:", grid_search_1.best_params_)
# Prédiction avec le modèle entrainé avec A ajusté sur l'ensemble de test de B (X_test_setB)
y_predicted_svm_B = grid_search_1.predict(X_test_setB)

# Affichage des performances
print("Rapport de classification : SVM A--->B")
print(classification_report(y_test_setB, y_predicted_svm_B))
```

FIGURE 1 – Exemple de choix des meilleurs Hyperparamètres

Cette approche a assuré une exploration exhaustive de l'espace des hyperparamètres, nous permettant de choisir les valeurs optimales pour construire un modèle SVM performant et bien généralisé.

### 3.1.3 Analyse du Resultat :

Le rapport de classification pour le SVM avec noyau linéaire montre comment chaque classe a été prédite par le modèle. La précision, le rappel et le score F1 pour chaque classe fournissent une mesure détaillée des performances.

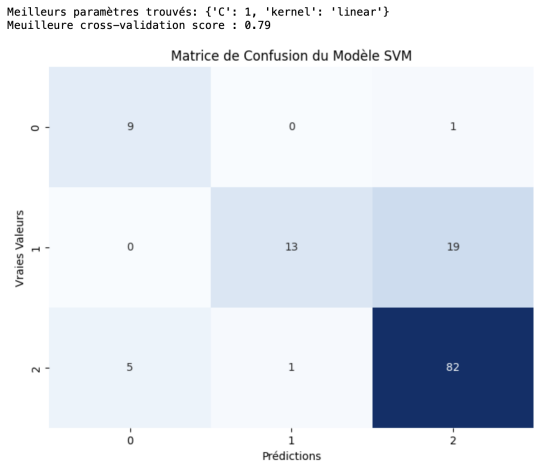


FIGURE 2 – Matrice de Confusion du SVM

Rapport de classification :

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.64 | 0.90 | 0.75 | 10 |
| 1 | 0.93 | 0.41 | 0.57 | 32 |
| 2 | 0.80 | 0.93 | 0.86 | 88 |
| accuracy | | | 0.80 | 130 |
| macro avg | 0.79 | 0.75 | 0.73 | 130 |
| weighted avg | 0.82 | 0.80 | 0.78 | 130 |

FIGURE 3 – Rapport de la classification

## 3.2 l'approches Random Forest

Le modèle KNN a été utilisé pour classer un point en fonction de la majorité des classes de ses voisins les plus proches. L'ajustement du nombre de voisins (k) a été réalisé pour trouver le meilleur équilibre entre biais et variance.

### 3.2.1 Description des Parametres :

L'étude des paramètres pour Random Forest se concentre généralement sur :

1. **Nombre d'arbres :**
   — Un nombre d'arbres élevé peut améliorer la robustesse du modèle, mais cela peut également augmenter le temps d'entraînement.
   — Un nombre trop faible d'arbres peut conduire à un surajustement.
2. **Profondeur maximale de l'arbre (profondeux maximale) :**
   — Contrôle la profondeur maximale de chaque arbre.
   — Des valeurs plus élevées peuvent conduire à un surajustement, tandis que des valeurs trop basses peuvent sous-ajuster le modèle.
3. **Nombre minimal d'échantillons par feuille (min sample leaf) :**
   — Spécifie le nombre minimal d'échantillons requis pour former une feuille.
   — Des valeurs plus élevées peuvent prévenir le surajustement.

L'étude de ces paramètres pour chaque méthode implique généralement la recherche de la combinaison optimale par validation croisée ou des méthodes similaires.

### 3.2.2 Méthodologie :

Pour déterminer les meilleurs hyperparamètres du modèle random forest, nous avons utilisé une approche de recherche systématique avec GridSearchCV. Cette technique explore différentes combinaisons. L'objectif était de trouver la combinaison qui maximise les performances du modèle en termes de précision, de rappel et de score F1.

### 3.2.3 Analyse du Resultat :

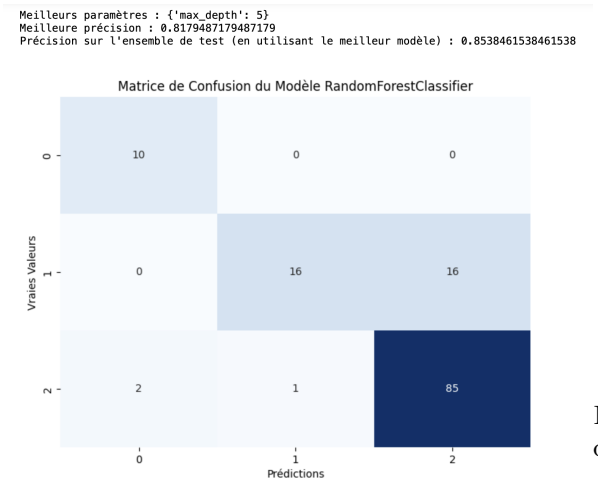


FIGURE 4 – Matrice de Confusion de l'arbre de decision

```
Rapport de classification sur les données de test:
              precision    recall  f1-score   support

           0       0.83      1.00      0.91        10
           1       0.94      0.50      0.65        32
           2       0.84      0.97      0.90        88

    accuracy                           0.85       130
   macro avg       0.87      0.82      0.82       130
weighted avg       0.87      0.85      0.84       130
```

FIGURE 5 – Rapport de la classification de l'arbre de decision

## 3.3 K Plus Proche Voisins (Kppv ou KNN)

Le modèle Random Forest, basé sur un ensemble d'arbres de décision, offre robustesse et généralisation. En ajustant le nombre d'arbres, il est possible de contrôler la complexité du modèle.

### 3.3.1 Description des Paramètres :

Pour le modèle KNN, le paramètre principal à ajuster est :

1. **Nombre de voisins (k) :**
   — Un k faible peut rendre le modèle sensible au bruit et aux fluctuations.
   — Un k élevé peut rendre le modèle plus stable mais peut aussi introduire du lissage excessif.

### 3.3.2 Méthodologie :

Nous avons mené une étude en faisant varier le nombre de voisins (k) de 1 à 10. Pour chaque valeur de k, nous avons entraîné un modèle k-NN sur l'ensemble d'apprentissage et évalué sa performance sur l'ensemble de test en utilisant l'accuracy comme métrique de performance.

### 3.3.3 Analyse du resultat :

Le rapport de classification pour le modèle KNN donne une vision détaillée de la performance du modèle pour chaque classe. La sensibilité du modèle aux variations du nombre de voisins (k) est illustrée, montrant comment le modèle évolue en fonction de ce paramètre.

## 3.4 Méthode Non Supervisée (Kmeans) :

L'objectif de cette étude était d'analyser comment la variation du nombre de clusters (k) dans l'algorithme K-Means affecte la performance de la méthode non supervisée pour la détection de sons de beatbox. Le choix de k détermine combien de clusters sont créés pour regrouper les données.

### 3.4.1 Description des Paramètres :

Le nombre de clusters (k) dans l'algorithme K-Means contrôle le nombre de regroupements effectués sur les données. Un k trop faible peut ne pas suffisamment capturer la structure des données, tandis qu'un k trop élevé peut entraîner une sur-segmentation.Dans le cas de la méthode non supervisée, K-Means, le paramètre clé était le nombre de clusters.

Rapport de classificcation 2

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.71 | 1.00 | 0.83 | 10 |
| 1 | 0.77 | 0.53 | 0.63 | 32 |
| 2 | 0.84 | 0.90 | 0.87 | 88 |
| accuracy | | | 0.82 | 130 |
| macro avg | 0.78 | 0.81 | 0.78 | 130 |
| weighted avg | 0.81 | 0.82 | 0.81 | 130 |

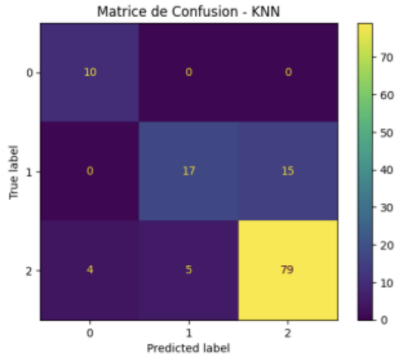


FIGURE 6 – Matrice de Confusion de l'arbre de decision

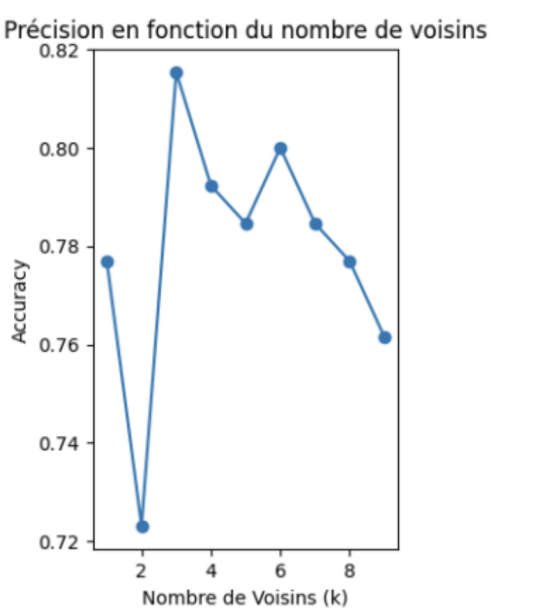


FIGURE 7 – Rapport de la classification de l'arbre de decision

### 3.4.2 Méthodologie :

Nous avobs utilisé pour cela l'inertie.L'inertie, dans le contexte de l'algorithme K-Means, est une mesure qui évalue la compacité des clusters formés par l'algorithme. Plus précisément, l'inertie est la somme des carrés des distances entre chaque point de données et le centroïde de son cluster attribué. L'objectif de K-Means est de minimiser cette somme des carrés, ce qui se traduit par la création de clusters compacts et bien délimités.

### 3.4.3 Resultat :

L'analyse de l'inertie dans le cadre de l'algorithme K-Means a fourni des insights essentiels sur la qualité et la compacité des clusters formés. Voici un résumé des principaux résultats :

Précision du modèle K-means sur les données d'entraînement : 0.7102564102564103

Précision du modèle K-means sur les données de test : 0.7076923076923077

Rapport de classification sur les données de test :

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 1.00 | 0.20 | 0.33 | 10 |
| 1 | 0.50 | 0.34 | 0.41 | 32 |
| 2 | 0.75 | 0.90 | 0.81 | 88 |
| accuracy | | | 0.71 | 130 |
| macro avg | 0.75 | 0.48 | 0.52 | 130 |
| weighted avg | 0.70 | 0.71 | 0.68 | 130 |

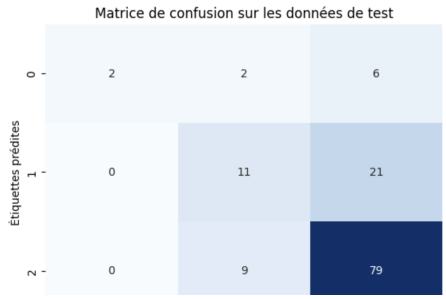


FIGURE 8 – Matrice de Confusion du kmeans

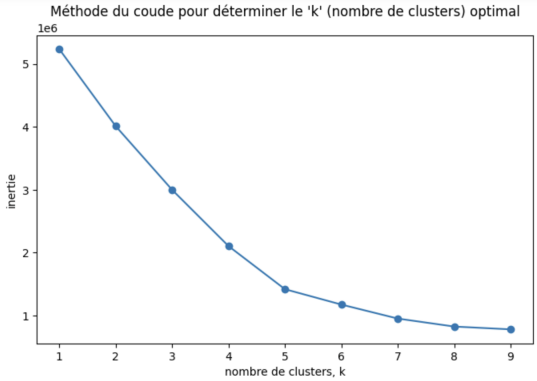


FIGURE 9 – Courbe de l'iniertie en fonction du nombre de cluster

**Remarque 1.** *De ces deux Interprétations de la matrice de confusion on peut déduire directement les sons qui ont été correctement prédits par notre modèle sans erreur de confusion avec d'autres .*

# 4 Notre Propre Etude

## 4.1 Étude Comparative entre les Ensembles A et B après Filtrage :

Après avoir filtré l'ensemble A pour retirer les éléments( c'est a dire Artifact) qui n'existaient pas dans l'ensemble B, nous avons mené une étude comparative en utilisant le modèle Support Vector Classifier (SVC). Cette approche visait à évaluer comment le modèle généralisait les caractéristiques extraites des battements cardiaques entre les deux ensembles de données provenant de sources distinctes.

## 4.2 Évaluation du Modèle Entraîné sur A et testé sur l'ensemble de Test B :

Après avoir entraîné le modèle sur l'ensemble A, nous avons évalué ses performances sur l'ensemble de test B. Cette approche permet de comprendre comment le modèle généralise ses capacités de classification lorsqu'il est confronté à des données provenant d'une source différente.

### 4.2.1 Méthodologie :

1. **Entraînement du Modèle sur A :**
   — Le modèle a été entraîné en utilisant l'ensemble A pour apprendre les motifs et les caractéristiques spécifiques aux battements cardiaques de cette source.
2. **Évaluation sur l'ensemble de Test B :**
   — Le modèle préalablement entraîné sur A a été testé sur l'ensemble B, qui n'a pas été utilisé pendant la phase d'entraînement. Cela permet d'évaluer la capacité du modèle à généraliser et à classifier efficacement les battements cardiaques provenant d'une source différente.

## 4.3 Résultats :

Les résultats de cette évaluation sur l'ensemble de test B fournissent des informations cruciales sur la capacité du modèle à s'adapter à des données de sources différentes. Les métriques de performance, telles que l'accuracy, le rappel et le score F1, reflètent la capacité du modèle à maintenir des performances élevées dans un contexte où il n'a pas été spécifiquement entraîné.

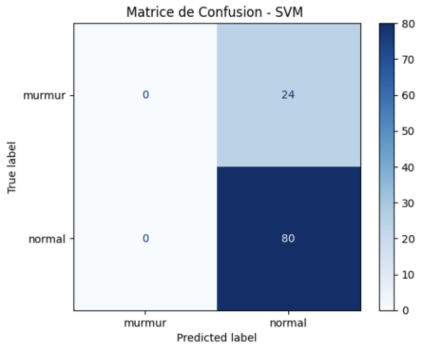


FIGURE 10 – Matrice de Confusion de la comparaison entre A et B

```
Meilleurs paramètres trouvés: {'C': 1, 'gamma': 0.001, 'kernel': 'rbf'}
Rapport de classification : SVM A--->B
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        24
           1       0.77      1.00      0.87        80

    accuracy                           0.77       104
   macro avg       0.38      0.50      0.43       104
weighted avg       0.59      0.77      0.67       104
```

FIGURE 11 – Rapport de classification

## 4.4 Évaluation du Modèle Entraîné sur B et testé sur l'ensemble de Test A :

Par contre quand on a entraîné le modèle avec B et tester sur A cela nous a conduit a une contre performance du modèle en terme d'accuracy .

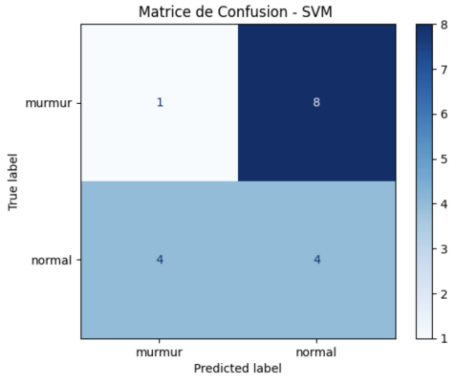


FIGURE 12 – Matrice de Confusion de la comparaison entre B et A

```
Meilleurs paramètres trouvés: {'C': 0.1, 'kernel': 'linear'}
Rapport de classification : SVM A--->B
              precision    recall  f1-score   support

           0       0.20      0.11      0.14         9
           1       0.33      0.50      0.40         8

    accuracy                           0.29        17
   macro avg       0.27      0.31      0.27        17
weighted avg       0.26      0.29      0.26        17
```

FIGURE 13 – Rapport de classification

**Remarque 2.** *Ce comportement étrange du modèle pourrait etre due a la provenance des deux type de données, il pourrait y avoir des différences inhérentes entre les sources de données (par exemple, la qualité des enregistrements, les conditions environnementales) qui affectent la performance du modèle.*

## 4.5 Etude sur les données les jeux de données Entre eux :

Dans cette partie, nous avons fait une étude sur le jeux de données A et sur le jeux de données B séparement afin de déterminer la performance du modèle sur des données ayant même les caractéristiques mais provenant de deux source différentes. (voir les deux resultat ci dessous)

### 4.5.1 Jeu de données A :

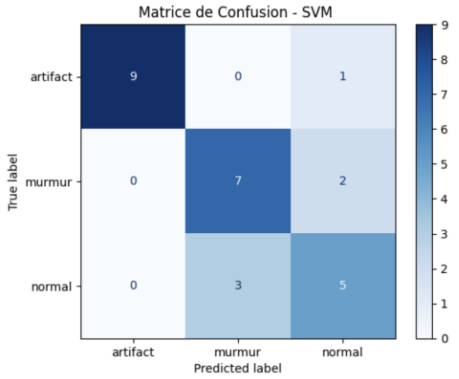


FIGURE 14 – Matrice de Confusion de l'etude sur A

```
Meilleurs paramètres trouvés: {'C': 0.001, 'kernel': 'linear'}
Rapport de classification :
              precision    recall  f1-score   support

           0       1.00      0.90      0.95        10
           1       0.70      0.78      0.74         9
           2       0.62      0.62      0.62         8

    accuracy                           0.78        27
   macro avg       0.78      0.77      0.77        27
weighted avg       0.79      0.78      0.78        27
```

FIGURE 15 – Rapport de classification

### 4.5.2 Jeu de données B :

**Remarque 3.** *Le but de cette étude sur chaque type de données de manière isolé était de bien comprendre que le modèle reagit bien sur des données provenant de la même source, ce qui est bien visible sur les valeurs des précisions sur les rapport des figure 13 de A et figure 15 de B*

```
Meilleurs paramètres trouvés: {'C': 0.1, 'kernel': 'linear'}
Rapport de classification :
              precision    recall  f1-score   support

           0       0.71      0.42      0.53        24
           1       0.84      0.95      0.89        80

    accuracy                           0.83       104
   macro avg       0.78      0.68      0.71       104
weighted avg       0.81      0.83      0.81       104
```

FIGURE 16 – Matrice de Confusion de l'etude sur B

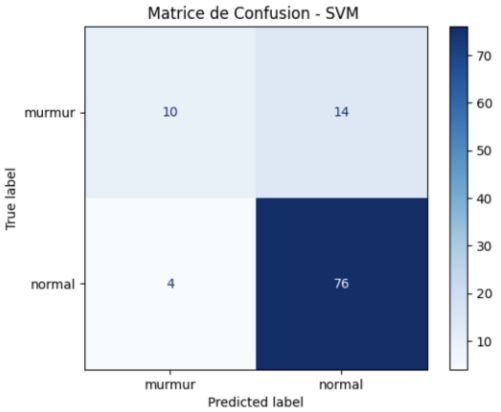


FIGURE 17 – Rapport de classification

# 5 Conclusion

De cette étude sur les battements cardiaques on en en déduit que :

— Les trois méthodes supervisées et la méthode non supervisée ont montré des performances prometteuses pour la classification des battements cardiaques sur les données fusionnées .
— Le SVM avec noyau RBF ou linéaire dépend vraiment des caractéristique des données et leurs distributions.
— KNN a bien adapté sa sensibilité aux données.
— Random Forest a démontré une forte robustesse.
— La provenance des données peut parfois impacté sur la performance du modèle.

## 5.1 Bilan de nos travaux :

Travailler un sujet aussi passionnant que la classification de battements cardiaque nous a appris beaucoup de choses dans ce vaste domaine qui est l'apprentissage supervisé et non supervisé allant d'une expérience d'implémentation et d'analyses de nombreuses approches de classifications. Il était très agréable de pouvoir mettre en pratique les enseignements vues durant les scéances en amphi et TD/ Nous retenons également de ce projet, la nécessité d'une bonne préparation, l'importance de la recherche la réalisation d'un d'une prédiction.

## 5.2 Améliorations Possibles :

Nous pouvons continuer ce projet de nombreuses façons :

— Nous pourrions explorer d'autres techniques de determination moins couteux que le Gridsearch et meme temps .
— Nous pourrions également comparer des méthodes supervisées et non supervisées entre elles après reduction de dimension ACP.

# Références

[1] sklearn https ://scikit-learn.org

[2] Numpy *https ://numpy.org*