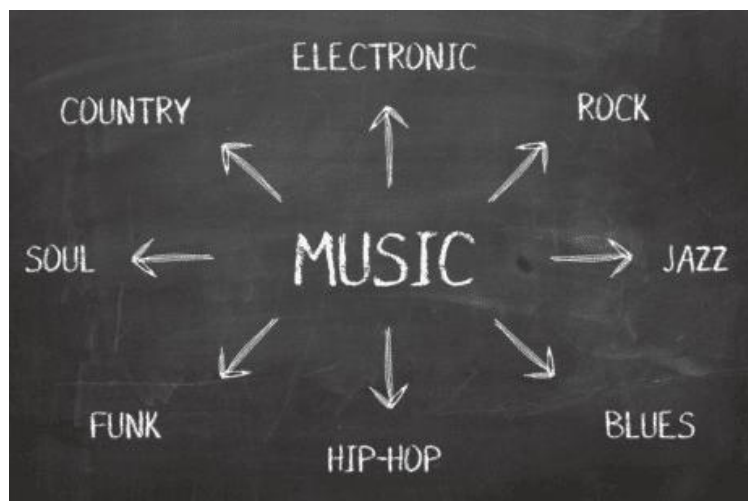


Polytech Anancy

Rapport de projet

DATA 832

Mise en place d'un workflow d'apprentissage supervisée avec scikit-learn dans la reconnaissance de genre musicaux



DEGOUEY Corentin

BERCIER Thomas

04/04/2025

Table des matières

Introduction	2
Préparation des données	2
Mesures de performances	3
Modèles de classifications sans utiliser de PCA	3
Arbre de décisions :	3
KNN	4
Random Forest	5
Réseau de Neurones MLP	6
Comparaison	7
Modèles de classification en utilisant une PCA.....	8
Conclusion	9

Introduction

Ce projet vise à classifier des morceaux musicaux en genres à l'aide de méthodes d'apprentissage supervisé. Nous utilisons le dataset GTZAN, composé de 1 000 extraits répartis en 10 genres (rock, jazz, blues, reggae, etc.). L'objectif est de comparer plusieurs modèles (arbres de décision, kNN, Random Forest, réseau de neurones) en optimisant leurs hyperparamètres et en évaluant leurs performances. Afin de réaliser cette classification nous utilisons Python et les bibliothèques suivantes :

- Pandas afin d'effectuer des opérations sur les dataframes
- Matplotlib afin de visualiser certains résultats
- Sklearn qui implémente directement les différents algorithmes utiles au projet
- Seaborn pour les visualisations des matrices de confusions

Le dataset GTZAN disponible sur Kaggle fournit, en plus des morceaux de musiques, leurs analyses spectrales et les features des morceaux (MFCC, chroma, etc.) directement extraites. Nous utilisons donc ses features pour l'entraînement des modèles.

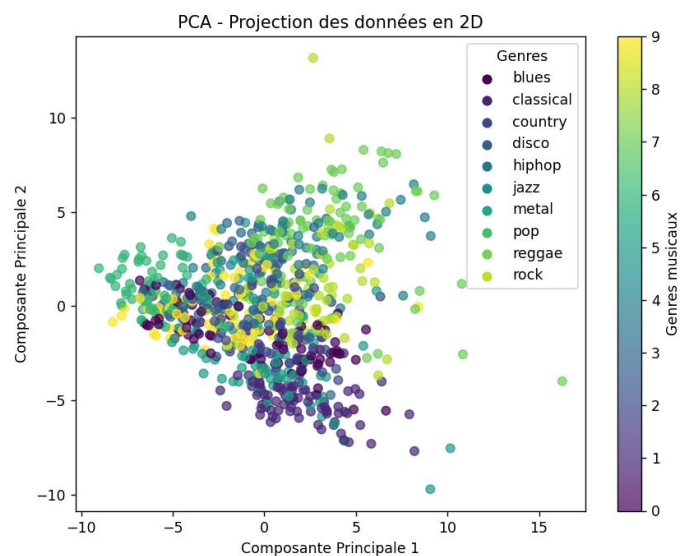
Préparation des données

Afin d'analyser les données nous les séparons pour l'entraînement (80%) et le test (20%).

Nous normalisons également les données en utilisant la fonction `StandardScaler` qui soustrait la moyenne et divise par l'écart type. La normalisation permet d'éviter que certaines features dominent l'apprentissage ce qui améliorent la performance des modèles.

Le dataset comprenant plus de 58 features nous allons appliquer une PCA à l'ensemble de données dans le but de réduire la complexité tout en gardant l'information pertinente. Les tests des différents modèles seront faits avec et sans PCA pour déterminer son influence. La variance expliquée des données est de 95% afin que la réduction de dimension conserve assez d'informations, ce qui conserve finalement 33 features.

Afin d'illustrer l'effet de la réduction de dimension, les données projetées sur les deux premières composantes principales ont été visualisées en 2D. Cette projection montre comment les différents genres musicaux sont distribués dans l'espace réduit, facilitant ainsi la compréhension de la structure des données et des relations entre les classes.



La visualisation permet de constater que certains genres sont bien séparés (blues et reggae), tandis que d'autres se chevauchent davantage (reggae et rock), ce qui suggère que certains genres peuvent être plus difficiles à classifier.

Mesures de performances

Afin de mesurer les performances de nos modèles nous utiliserons des matrices de confusions qui permettront une analyse visuelle des résultats. Nous analyserons également la précision d'une classe, son recall (rappel) et son f1 score.

- Précision = parmi les éléments prédits comme positifs, combien sont réellement positifs
- Rappel = parmi les éléments réellement positifs, combien ont été correctement prédits comme positifs

Le F1-score est la moyenne harmonique de la précision et du rappel :

$$F1 = 2 * \frac{precision * rappel}{precision + rappel}$$

Dans les matrices de confusions les genres: Blues (classe 0), Country (classe 1), Country (classe 2), Disco (classe 3), HipHop (classe 4), Jazz (classe 5), Metal (classe 6), Pop (classe 7), Reggae (classe 8), Rock (classe 9).

Modèles de classifications sans utiliser de PCA

Pour chaque modèle nous allons effectuer une recherche des meilleurs hyperparamètres en utilisant GridSearch ainsi qu'une validation croisée. GridSearch va pour les différents paramètres donnés, entraîner et évaluer le modèle. Chaque combinaison de paramètres va subir une validation croisée et la meilleure combinaison (selon un critère) va être retournée. Nous avons choisi comme critère la précision du modèle puisque notre but est d'avoir le modèle le plus précis possible.

La validation croisée va diviser les données en k sous-ensembles et entraîner le modèle sur les k-1 premiers sous-ensembles et tester celui-ci sur le dernier. Chaque sous-ensemble subit un entraînement et un test puis une moyenne de précision est ressortie. L'utilisation de la validation croisée permet d'avoir une évaluation plus fiable du modèle si les données sont mal réparties dans le dataset. Dans notre cas nous choisissons de diviser les données en 5 sous-ensembles qui semblent être un bon compromis pour avoir un biais faible mais un temps de calcul acceptable.

Arbre de décisions :

Choix des hyperparamètres :

- criterion: ['gini', 'entropy']
 - Gini est rapide, tandis qu'Entropy peut mieux gérer des classes déséquilibrées.
- max_depth: [None, 10, 20, 30, 40, 50]
 - Contrôle la profondeur de l'arbre. Trop profond = sur-apprentissage, trop peu profond = sous-apprentissage.
- min_samples_split: [2, 5, 10]
 - Nombre minimal d'échantillons pour diviser un nœud. Plus grand = moins de sur-apprentissage.

- min_samples_leaf: [1, 2, 4]
 - Nombre minimal d'échantillons dans une feuille. Plus grand = arbres plus simples et moins sujets au sur-apprentissage.

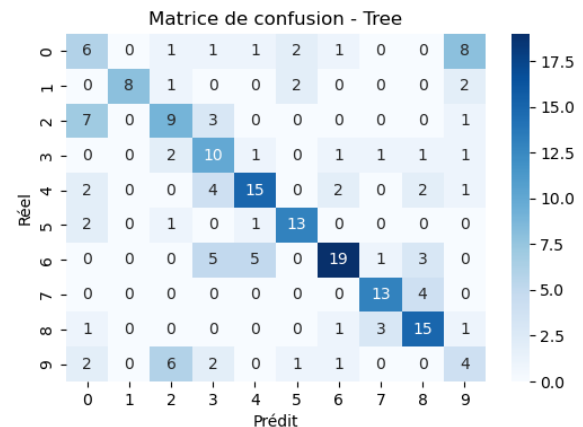
Après validation croisée les meilleurs paramètres sont :

criterion: entropy, max_depth: 10, min_samples_leaf: 1, min_samples_split: 5

Résultats :

La précision de l'arbre après tuning des hyperparamètres est de **56%** ce qui est correct.

	precision	recall	f1-score	support
blues	0.30	0.30	0.30	20
classical	1.00	0.62	0.76	13
country	0.45	0.45	0.45	20
disco	0.40	0.59	0.48	17
hiphop	0.65	0.58	0.61	26
jazz	0.72	0.76	0.74	17
metal	0.76	0.58	0.66	33
pop	0.72	0.76	0.74	17
reggae	0.60	0.71	0.65	21
rock	0.22	0.25	0.24	16



Grâce au rapport de classification nous observons :

- Blues : Faible performance avec une précision et un recall de 0.30. Le modèle confond souvent le blues avec rock.
- Classical : Précision parfaite de 1.00, mais un recall de 0.62, indiquant qu'il reconnaît bien les morceaux classiques mais en rate certains.
- Rock : Très faible précision (0.22) et recall (0.25), le modèle peine à classer correctement le rock, avec beaucoup d'erreurs, notamment avec blues.
- Jazz, Hiphop, Pop, Metal : Bonnes performances avec des scores F1 entre 0.61 et 0.74, montrant une classification équilibrée pour ces genres.

KNN

Choix des hyperparamètres :

- n_neighbors : Tester différentes valeurs (3, 5, 7, 10, 15, 30) permet de trouver un équilibre entre sur-apprentissage (avec peu de voisins) et sous-apprentissage (avec trop de voisins).
- weights : Le choix entre 'uniform' (poids égal pour tous les voisins) et 'distance' (les voisins plus proches ont plus d'influence) aide à mieux capturer les relations locales dans les données.
- metric : Tester les métriques 'euclidean', 'manhattan' et 'minkowski' permet d'évaluer quelle distance est la plus adaptée aux caractéristiques des données (en fonction de leur distribution et de leur échelle).

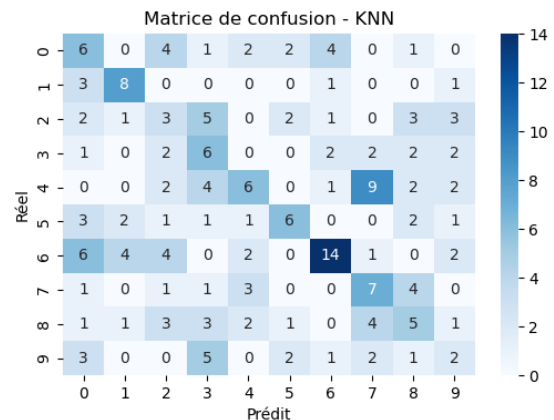
Après validation croisée les meilleurs paramètres sont :

metric: manhattan, n_neighbors: 7, weights: distance

Résultats :

Après tuning des hyperparamètres la précision est de **31,5%** ce qui est faible.

	precision	recall	f1-score	support
blues	0.23	0.30	0.26	20
classical	0.50	0.62	0.55	13
country	0.15	0.15	0.15	20
disco	0.23	0.35	0.28	17
hiphop	0.38	0.23	0.29	26
jazz	0.46	0.35	0.40	17
metal	0.58	0.42	0.49	33
pop	0.28	0.41	0.33	17
reggae	0.25	0.24	0.24	21
rock	0.14	0.12	0.13	16



Le modèle kNN montre des performances faibles sur la majorité des genres musicaux ce qui s'explique par la mauvaise précision du modèle. Classical est le mieux classé avec une précision de 0.50 et un recall de 0.62, tandis que Rock et Blues sont particulièrement mal classés, avec des précisions respectives de 0.14 et 0.23. Des genres comme Country, HipHop, et Disco montrent également de faibles résultats.

La matrice de confusion révèle que Blues et Rock sont souvent confondus avec d'autres genres, tandis que Classical est mieux identifié mais confondu avec Jazz. Les genres comme Metal et Jazz sont mieux séparés, mais des erreurs persistent.

Random Forest

L'algorithme de Random Forest permet de créer plusieurs arbres de décisions afin d'affiner les résultats et augmenter la précision. De ce fait de nombreux hyperparamètres sont identiques.

Choix des hyperparamètres :

- `n_estimators` (50, 100, 200) : Tester différents nombres d'arbres pour équilibrer performance et temps de calcul.
- `max_depth` (None, 10, 20, 30) : identique au modèle d'arbre de décisions.
- `min_samples_split` (2, 5, 10) : identique au modèle d'arbre de décisions.
- `min_samples_leaf` (1, 2, 4) : identique au modèle d'arbre de décisions.
- `bootstrap` (True, False) : permet de déterminer si chaque arbre est construit avec un sous-ensemble de données aléatoires. Cela permet de réduire la variance et éviter le sur-apprentissage.

Après validation croisée les meilleurs paramètres sont :

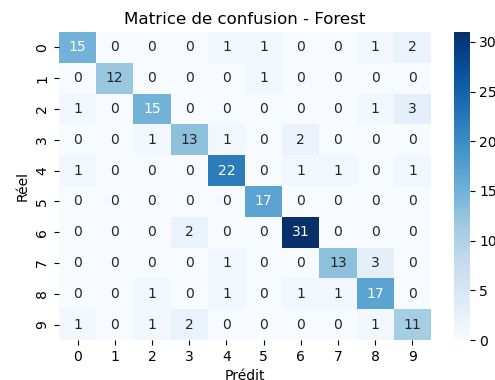
bootstrap: False, max_depth: 20, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 100

Résultats :

Après tuning des hyperparamètres la précision est de **83%** ce qui est une très bonne performance.

Classification Report (Random Forest):

	precision	recall	f1-score	support
blues	0.83	0.75	0.79	20
classical	1.00	0.92	0.96	13
country	0.83	0.75	0.79	20
disco	0.76	0.76	0.76	17
hiphop	0.85	0.85	0.85	26
jazz	0.89	1.00	0.94	17
metal	0.89	0.94	0.91	33
pop	0.87	0.76	0.81	17
reggae	0.74	0.81	0.77	21
rock	0.65	0.69	0.67	16



Le modèle montre des performances très bonnes sur la majorité des genres. Le genre Classical est parfaitement classifié (f1 score de 0,96). Les erreurs sont rares, mais il y a quelques confusions, surtout pour Rock et Reggae. La matrice de confusion montre que Classical et Jazz sont parfaitement identifiés (seulement 1 erreur), tandis que Rock et Blues présentent des erreurs de classification. Le Rock est confondu avec le Blues et le Country ce qui semble être une tendance avec tous les modèles.

Réseau de Neurones MLP

Le réseau MLP de scikit-learn est une architecture de réseau de neurones utilisée pour des tâches d'apprentissage supervisé. Un MLP est un type de perceptron multicouche, qui se compose de plusieurs couches de neurones connectées entre elles. Il est composé de couches d'entrées/sorties et de couches cachées. Les données passent de couche en couche où l'on applique un poids.

Choix des hyperparamètres :

- `hidden_layer_sizes` [(50,), (100,), (50, 50), (100, 100)]: Définit la topologie du réseau, (nombre de neurones par couche cachée). Tester différentes architectures permet d'explorer la complexité des réseaux plus simples. Permet d'ajuster la capacité d'apprentissage du modèle et à éviter l'overfitting.
- `activation` ['relu', 'tanh']: Les fonctions d'activation déterminent comment les sorties des neurones sont calculées. ReLU est plus efficace et plus rapide à entraîner. Tester ces deux fonctions permet d'évaluer leur impact sur la performance et la convergence du modèle.
- `solver` ['adam', 'sgd']: Définir l'algorithme utilisé pour minimiser la fonction de coût. Adam est efficace et gère les gradients à différentes échelles, SGD est plus rapide. Tester permet de comparer vitesse de convergence et capacité à trouver des minima globaux.
- `alpha` [0.0001, 0.001, 0.01]: Correspond à la force de la régularisation L2, qui aide à prévenir l'overfitting. Valeur faible = meilleure adaptation, Valeur haute = modèle plus général. Tester permet d'ajuster la compromission entre biais et variance.
- `learning_rate` ['constant', 'adaptive']: Contrôler la stratégie d'apprentissage. 'constant' = taux d'apprentissage fixe, 'adaptive' ajuster en fonction de la performance du modèle.

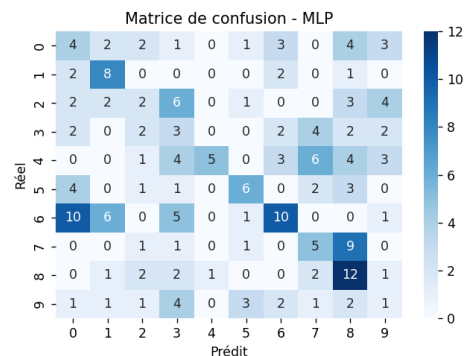
Après validation croisée les meilleurs paramètres sont : activation: tanh, alpha: 0.01, hidden_layer_sizes: (100, 100), learning_rate: constant, solver: sgd

Résultats :

Après tuning des hyperparamètres la précision est de **28%** ce qui est une très médiocre.

Classification Report (MLP):

	precision	recall	f1-score	support
blues	0.16	0.20	0.18	20
classical	0.40	0.62	0.48	13
country	0.17	0.10	0.12	20
disco	0.11	0.18	0.14	17
hiphop	0.83	0.19	0.31	26
jazz	0.46	0.35	0.40	17
metal	0.45	0.30	0.36	33
pop	0.25	0.29	0.27	17
reggae	0.30	0.57	0.39	21
rock	0.07	0.06	0.06	16

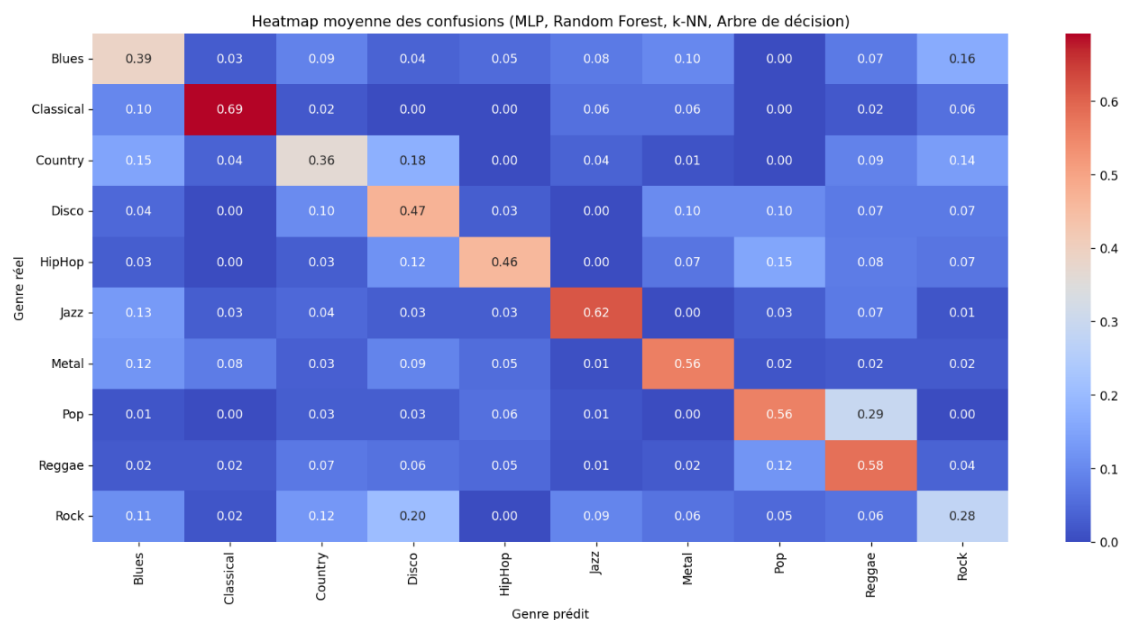


Les performances sont assez faibles pour plusieurs genres musicaux, notamment Blues, Country, Disco, et Rock. Bien que HipHop ait une bonne précision, son faible recall montre que beaucoup d'exemples sont mal classés. Les genres comme Classical, Jazz, Metal, et Reggae sont mieux classés, mais restent loin d'être parfaits. Le modèle a toujours beaucoup de mal à classer certains genres, notamment Rock et Blues.

Comparaison

	Tree	KNN	Random Forest	MLP
Précision	56%	31,5%	83%	28%
Vitesse de calcul	Rapide	Rapide	Lent	Très lent

Genre	Précision moyenne	Rappel moyen
Blues	0.38	0.39
Classical	0.73	0.7
Country	0.4	0.36
Disco	0.38	0.47
Hip-hop	0.68	0.46
Jazz	0.63	0.62
Metal	0.67	0.56
Pop	0.53	0.56
Reggae	0.47	0.58
Rock	0.27	0.28



Les modèles d'arbres de décision et de forêts aléatoires sont les plus efficaces pour classifier les genres musicaux, notamment pour HipHop, Jazz, Classical et Metal. Cependant, des genres comme Blues et Rock sont souvent mal classifiés et confondus. Les erreurs de classification entre certains genres peuvent être expliquées par leurs ressemblance rythmiques et harmoniques.

HipHop, Jazz, Metal et Classical sont bien classifiés grâce à des caractéristiques distinctes qui ne se retrouvent pas ailleurs comme les rythmes répétitifs du HipHop, les improvisations du Jazz, les tempéraments rapides et saturés du Metal, et les structures claires de la musique Classique. Blues et Rock partagent des rythmes simples, ce qui les rend difficiles à différencier, surtout lorsque des modèles comme le KNN et MLP ne captent pas bien les nuances subtiles de chaque genre. De même pour Disco, Country et Pop qui sont difficiles à différencier.

Modèles de classification en utilisant une PCA

Afin d'améliorer la performance du modèle MLP nous avons appliqué une PCA aux données. Nous avons également appliqué une PCA aux autres modèles mais la précision a systématiquement baissé.

Cela peut s'expliquer par plusieurs facteurs :

- Le KNN dépend beaucoup de la structure géométrique des données. La PCA modifiant cette structure le KNN perd beaucoup en précision
- Concernant les arbres de décisions et l'algorithme Random Forest, ces algorithmes exploitent mieux les caractéristiques locales et les relations linéaires qui sont supprimées ou diminuées par la PCA

En revanche, sur le réseau de neurones, la PCA réduit fortement le bruit ce qui permet de 'nettoyer' les données. La réduction de dimensions permet au modèle d'apprendre plus facilement puisqu'il a moins de paramètres à apprendre.

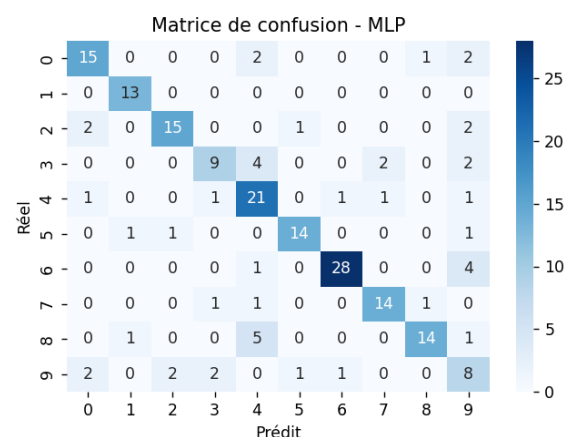
En appliquant de nouveau une validation croisée pour le tuning des hyperparamètres nous obtenons les choix suivants :

activation: tanh, alpha: 0.01, hidden_layer_sizes: (100, 100), learning_rate: constant, solver: sgd

Résultats:

	Avant PCA	Après PCA
Précision	28%	72%

	precision	recall	f1-score	support
blues	0.75	0.75	0.75	20
classical	0.87	1.00	0.93	13
country	0.83	0.75	0.79	20
disco	0.69	0.53	0.60	17
hiphop	0.62	0.81	0.70	26
jazz	0.88	0.82	0.85	17
metal	0.93	0.85	0.89	33
pop	0.82	0.82	0.82	17
reggae	0.88	0.67	0.76	21
rock	0.38	0.50	0.43	16



	F1 score avant PCA	F1 score après PCA
Blues	0.2	0.75
Classical	0.73	0.93
Country	0.2	0.79
Disco	0.3	0.6
HipHop	0.19	0.7
Jazz	0.46	0.85
Metal	0.44	0.89
Pop	0.5	0.82
Reggae	0.63	0.76
Rock	0.11	0.43

On constate une amélioration globale sur presque tous les genres. Tous les genres ont un rappel supérieur à 0,75 hormis le Disco et le Rock (encore assez faible). Les genres Classique, Jazz, Metal et Reggae sont bien détectés ce qui confirme les analyses précédentes. Une amélioration notable est la classification du Blues qui est bien mieux reconnu et surtout moins confondu avec le Rock.

La PCA a permis de transformer un MLP inefficace en un modèle performant, en facilitant la détection des structures pertinentes dans les données. La plupart des genres sont désormais bien classifiés, sauf Rock qui reste mal classifié. Ce résultat montre l'intérêt d'une réduction de dimensionnalité pour des modèles sensibles comme les réseaux de neurones.

Conclusion

Ce projet a permis d'évaluer et de comparer différentes approches d'apprentissage supervisé pour la classification des genres musicaux à partir du dataset GTZAN. Après avoir appliqué plusieurs modèles, à savoir les arbres de décision, kNN, Random Forest, et le réseau de neurones MLP, plusieurs conclusions peuvent être tirées quant à la facilité ou la difficulté de classer certains genres musicaux, ainsi que les genres fréquemment confondus.

Les genres musicaux tels que Classical, Jazz, HipHop et Metal ont présenté de bonnes performances de classification. Ces genres sont relativement faciles à différencier grâce à des caractéristiques distinctes.

À l'inverse, les genres Blues et Rock ont été plus difficiles à classifier en raison de leur similarité rythmique et harmonique. Les performances des modèles (KNN et MLP), ont été moins satisfaisantes pour ces genres. Ces deux genres sont souvent confondus, ce qui peut être attribué à leur simplicité structurelle. Malgré cela, l'application de la réduction de dimension via la PCA a permis d'améliorer la classification du Blues, qui est devenu mieux différencié du Rock.

Des genres comme Disco, Pop, Country et Reggae ont également présenté des difficultés de classification.

L'utilisation de la PCA a permis d'améliorer considérablement les performances du modèle MLP, en particulier pour les genres où des erreurs de classification étaient fréquentes. La PCA a contribué à éliminer le bruit et à simplifier la structure des données, ce qui a facilité l'apprentissage du modèle.

Les algorithmes de classification les plus adaptés à ce type de données sont donc la Random Forest et le réseau de neurones MLP (sur des données en dimensions réduites).