Analyse Musicale Spotify

Classification des Genres, Prédiction de Popularité & Recommandation

Faycal Raghibi, Guerouaoui Ilyas

Février 2026

1 Introduction

Les plateformes de streaming musical telles que Spotify hébergent des millions de titres couvrant une grande variété de genres. La classification automatique des genres, la prédiction de popularité et la recommandation basée sur le contenu sont des tâches fondamentales qui améliorent l'expérience utilisateur et la curation de la plateforme. Ce rapport présente la méthodologie et les résultats d'un pipeline d'apprentissage automatique construit autour d'un jeu de données Spotify contenant des descripteurs audio numériques extraits de l'API Web Spotify.

Le projet aborde trois objectifs interconnectés :

1. **Classification des genres** — attribuer l'une des étiquettes de genre à chaque morceau en fonction de ses caractéristiques audio.
2. **Prédiction de popularité** — estimer le score continu de popularité d'un morceau par régression.
3. **Recommandation basée sur le contenu** — suggérer des morceaux similaires en mesurant la proximité dans l'espace des caractéristiques.

Toutes les expériences ont été implémentées en Python à l'aide de Scikit-Learn, NumPy, Pandas, Matplotlib et Seaborn. La suite de ce rapport est organisée comme suit : la Section **??** décrit les jeux de données ; la Section **??** détaille le pipeline de prétraitement ; les Sections **??** et **??** présentent respectivement les tâches de classification et de régression ; la Section **??** introduit l'approche de recommandation ; la Section **??** décrit l'expérience bonus de prédiction conforme ; et la Section **??** conclut par une discussion.

2 Description des Données & Analyse Exploratoire

2.1 Jeux de Données Disponibles

Quatre fichiers CSV ont été fournis :

Fichier	Lignes	Colonnes	Objectif
`spotify_dataset_train.csv`	25 493	17	Entraînement (incl. `genre`)
`spotify_dataset_test.csv`	2 834	16	Test (sans étiquette `genre`)
`spotify_dataset_subset.csv`	—	—	Sous-ensemble pour la régression de popularité
`recommendation_spotify.csv`	—	—	Pool pour le système de recommandation

TABLE 1 – Aperçu des jeux de données du projet.

2.2 Dictionnaire des Caractéristiques

Chaque morceau est décrit par les caractéristiques listées dans le Tableau **??**.

Caractéristique	Type	Description
`track_id`	chaîne	Identifiant unique Spotify
`track_name`	chaîne	Nom du morceau
`artists`	chaîne	Artiste(s) interprète(s)
`release_date`	chaîne	Date de sortie (année extraite)
`acousticness`	flottant	Mesure de confiance de la qualité acoustique
`danceability`	flottant	Aptitude à la danse
`duration_ms`	entier	Durée du morceau en millisecondes
`energy`	flottant	Mesure d'intensité perceptive
`instrumentalness`	flottant	Probabilité que le morceau soit instrumental
`key`	entier	Tonalité musicale (0–11, classe de hauteur)
`liveness`	flottant	Probabilité de performance en direct
`loudness`	flottant	Intensité sonore globale en dB
`mode`	entier	Modalité (0 = mineur, 1 = majeur)
`speechiness`	flottant	Présence de paroles
`tempo`	flottant	BPM estimé
`valence`	flottant	Positivité musicale
`popularity`	entier	Score de popularité (0–100)
`explicit`	entier	Indicateur de contenu explicite
`genre`	chaîne	Étiquette de genre (jeu d'entraînement uniquement)

TABLE 2 – Dictionnaire des caractéristiques des jeux de données Spotify.

2.3 Observations Exploratoires

Une exploration initiale du jeu d'entraînement a révélé plusieurs caractéristiques qui ont influencé la conception du pipeline :

— **Déséquilibre des classes** — La distribution des genres est très inégale. Certains genres (par ex. *pop*, *rock*) dominent le jeu de données tandis que d'autres sont significativement sous-représentés. Cela a motivé l'utilisation de poids équilibrés par classe dans le classificateur.
— **Valeurs manquantes** — Une faible proportion d'entrées contient des valeurs numériques manquantes, nécessitant une imputation.
— **Échelles des caractéristiques** — Les caractéristiques couvrent des plages très différentes (par ex. `duration_ms` $\sim 10^5$ vs. `acousticness` $\in [0, 1]$), rendant la standardisation nécessaire.
— **Dimensionnalité** — Une visualisation ACP des caractéristiques standardisées a montré un chevauchement substantiel entre les genres dans les deux premières composantes principales, laissant présager une tâche de classification difficile.

3 Pipeline de Prétraitement

Les étapes de prétraitement suivantes ont été appliquées de manière cohérente aux données d'entraînement et de test :

1. **Extraction de l'année.** La colonne `release_date` a été analysée pour extraire une caractéristique numérique `year`, fournissant un signal temporel compact.
2. **Sélection des colonnes.** Les identifiants non prédictifs (`track_id`, `track_name`, `artists`) et la chaîne brute `release_date` ont été supprimés.
3. **Imputation des valeurs manquantes.** Les valeurs manquantes restantes dans les colonnes numériques ont été remplies à l'aide de `SimpleImputer` avec une stratégie par la *moyenne*.
4. **Standardisation des caractéristiques.** Toutes les caractéristiques numériques ont été centrées et mises à l'échelle à variance unitaire via `StandardScaler`, ajusté sur le jeu d'entraînement et appliqué aux deux sous-ensembles.

5. **Encodage catégoriel.** La caractéristique catégorielle binaire `explicit` a été conservée telle quelle (déjà encodée en 0/1). Un encodage one-hot a été appliqué si nécessaire pour tout signal catégoriel supplémentaire.

Après le prétraitement, la matrice de caractéristiques contenait les colonnes suivantes utilisées pour la modélisation : `acousticness`, `danceability`, `duration_ms`, `energy`, `instrumentalness`, `key`, `liveness`, `loudness`, `mode`, `speechiness`, `tempo`, `valence`, `popularity`, `explicit` et `year`.

4 Classification des Genres

4.1 Sélection du Modèle

Un **Classificateur Random Forest** a été choisi pour la prédiction multi-classe des genres. Les forêts aléatoires agrègent de nombreux arbres de décision décorrélés via l'agrégation bootstrap, offrant une robustesse contre le surapprentissage et la capacité de gérer des types de caractéristiques mixtes. Les hyperparamètres suivants ont été définis :

— `n_estimators` = 100 — nombre d'arbres dans l'ensemble.
— `class_weight` = `balanced` — ajuste automatiquement les poids des échantillons de manière inversement proportionnelle aux fréquences des classes, atténuant l'effet du déséquilibre des classes.
— `random_state` = 42 — graine fixe pour la reproductibilité.

4.2 Protocole d'Évaluation

Le modèle a été évalué par **validation croisée stratifiée à 5 plis** sur le jeu d'entraînement. La métrique principale est le **score F1 micro-moyenné**, qui calcule la précision et le rappel globaux sur toutes les classes et est équivalent à l'exactitude lorsque chaque échantillon se voit attribuer exactement une étiquette.

4.3 Résultats

Métrique	Valeur
F1 CV (micro) moyenne	0.4396
F1 CV (micro) écart-type	± 0.0132

TABLE 3 – Résultats de la validation croisée pour la classification des genres.

Un F1 micro-moyenné d'environ **0.44** reflète la difficulté inhérente de la tâche : de nombreux genres partagent des profils audio similaires, et l'espace des caractéristiques (descripteurs audio purement numériques) ne capture pas nécessairement les différences stylistiques de haut niveau. La pondération équilibrée des classes permet de s'assurer que les genres minoritaires ne sont pas systématiquement ignorés, mais la séparabilité globale reste limitée.

4.4 Prédiction sur le Jeu de Test

Après la validation croisée, le classificateur a été ré-entraîné sur l'ensemble du jeu d'entraînement et utilisé pour prédire les étiquettes de genre des 2 834 morceaux non étiquetés du jeu de test. Les prédictions ont été exportées dans `submission.csv` au format requis (`track_id`, `genre`).

5 Prédiction de Popularité

5.1 Définition de la Tâche

La colonne `popularity` (entier dans $[0, 100]$) sert de cible de régression. Cette tâche utilise le fichier dédié `spotify_dataset_subset.csv`, qui se concentre sur une sélection de morceaux.

5.2 Modèle

Un **Régresseur Random Forest** a été utilisé, reprenant la philosophie d'ensemble employée pour la classification. Les régresseurs à base d'arbres de décision sont bien adaptés pour capturer les relations non linéaires entre les caractéristiques audio et la popularité sans nécessiter d'ingénierie de caractéristiques explicite.

5.3 Résultats

Métrique	**Valeur**
Erreur Quadratique Moyenne (MSE)	517.99
Coefficient de Détermination (R^2)	0.2126

TABLE 4 – Performance de la régression de popularité.

Un R^2 d'environ **0.21** indique que les caractéristiques audio seules expliquent approximativement 21% de la variance de la popularité. Ceci est attendu car la popularité est fortement influencée par des facteurs externes — notoriété de l'artiste, marketing, placement dans les playlists, tendances temporelles — qui ne sont pas capturés par les descripteurs audio. Le MSE de 517.99 correspond à une erreur quadratique moyenne d'environ 22.8 points de popularité sur l'échelle de 0 à 100.

6 Recommandation Basée sur le Contenu

6.1 Approche

Une stratégie de **filtrage basé sur le contenu** a été implémentée en utilisant la **similarité cosinus** sur les vecteurs de caractéristiques audio standardisés. Étant donné un morceau requête, le système récupère les k morceaux les plus similaires du pool `recommendation_spotify.csv` en classant les similarités cosinus par paires.

6.2 Pipeline

1. Charger et prétraiter le jeu de données de recommandation (même pipeline que la Section **??**).
2. Calculer la matrice de similarité cosinus entre toutes les paires de morceaux.
3. Pour un morceau requête donné, trier les scores de similarité par ordre décroissant et retourner les k plus proches voisins (en excluant le morceau requête lui-même).

La similarité cosinus est un choix naturel dans ce contexte car elle mesure la proximité angulaire entre les vecteurs de caractéristiques, la rendant invariante aux changements d'échelle uniformes — une propriété importante lorsque les caractéristiques ont été standardisées mais peuvent encore présenter des plages dynamiques différentes selon le sous-ensemble.

7 Bonus : Prédiction Conforme

En tant qu'expérience supplémentaire, un cadre de **prédiction conforme** a été appliqué pour quantifier l'incertitude de prédiction sur la tâche de classification des genres.

7.1 Méthode

1. Un classificateur **K plus proches voisins (KNN)** a été entraîné sur le jeu d'entraînement après prétraitement standard.
2. Un **jeu de calibration** de 1 500 échantillons a été réservé à partir des données d'entraînement pour calculer les scores de non-conformité.
3. Pour chaque instance de test, le prédicteur conforme produit un **ensemble de prédiction** — un sous-ensemble d'étiquettes de genre garanti de contenir la vraie étiquette avec une probabilité d'au moins $1 - \alpha$, où α est le niveau de signification spécifié par l'utilisateur.

7.2 Résultats

α	**Couverture (%)**	**Taille moy. de l'ensemble**	**Ensembles vides (%)**
0.01	99.88	16.74	0.00
0.05	98.32	13.96	0.00
0.10	96.93	12.36	0.00
0.20	93.82	10.19	0.00

TABLE 5 – Résultats de la prédiction conforme à différents niveaux de signification.

Le prédicteur conforme atteint les garanties de couverture souhaitées à chaque niveau de signification. Cependant, les tailles moyennes des ensembles de prédiction sont importantes (par ex. ≈ 14 genres à $\alpha = 0.05$), reflétant le pouvoir discriminant limité des caractéristiques audio pour une séparation fine des genres. Cela corrobore le score F1 modeste observé dans la Section **??**.

8 Discussion & Conclusion

Ce projet a démontré un pipeline d'apprentissage automatique de bout en bout pour l'analyse de morceaux Spotify, couvrant la classification, la régression et la recommandation. Plusieurs observations méritent discussion :

— **Limitations des caractéristiques.** Les caractéristiques audio purement numériques fournies par l'API Spotify capturent des propriétés timbrales, rythmiques et harmoniques de bas niveau mais ne peuvent pas encoder des attributs sémantiques ou culturels de plus haut niveau. L'incorporation de métadonnées textuelles (paroles, biographie de l'artiste) ou de représentations spectrales (par ex. spectrogrammes Mel) pourrait améliorer significativement la classification des genres.

— **Déséquilibre des classes.** Malgré l'utilisation de poids équilibrés par classe, les genres sous-représentés restent difficiles à classifier. Des techniques telles que le suréchantillonnage (SMOTE) ou la classification hiérarchique (regroupement de genres apparentés) pourraient apporter des améliorations.

— **Modélisation de la popularité.** Un R^2 de 0.21 confirme que la popularité n'est que faiblement liée aux propriétés audio intrinsèques. Un ensemble de caractéristiques plus riche incluant des métriques de réseaux sociaux, le calendrier de sortie et l'exposition dans les playlists serait nécessaire pour une prévision pratique de la popularité.

— **Recommandation.** Le système de recommandation basé sur la similarité cosinus fournit une référence simple mais efficace. Des approches hybrides combinant filtrage basé sur le contenu et filtrage collaboratif pourraient offrir des recommandations plus diversifiées et précises.
— **Quantification de l'incertitude.** L'expérience de prédiction conforme illustre que les garanties de couverture statistique se font au prix d'ensembles de prédiction volumineux lorsque le classificateur sous-jacent a une précision modérée. Améliorer le classificateur de base resserrerait directement les intervalles conformes.

Dans l'ensemble, la famille Random Forest s'est avérée être un choix fiable et interprétable pour la classification et la régression sur des caractéristiques audio tabulaires. Les travaux futurs pourraient explorer les arbres à gradient boosté (XGBoost, LightGBM), les embeddings neuronaux ou la fusion multimodale pour améliorer davantage les performances.