

DATA MINING PADA DATASET SPOTIFY GLOBAL MUSIC



Nama : Fera Wanti Sijabat
NPM : 231510055
Dosen : Erlin Elisa,S.Kom.,M.Kom.

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS PUTERA BATAM
TAHUN 2025

1 □. Deskripsi Dataset

Dataset yang digunakan dalam penelitian ini adalah Spotify Global Music Dataset (2009–2025) yang diperoleh dari platform Kaggle. Dataset ini menyajikan data lagu global yang dipublikasikan dan diputarkan melalui platform Spotify selama rentang waktu lebih dari lima belas tahun. Data yang tersedia mencakup karakteristik audio lagu, metadata lagu, serta tingkat popularitas yang diukur berdasarkan interaksi pengguna pada platform tersebut.

Pemilihan dataset ini didasarkan pada relevansinya terhadap analisis tren musik digital serta ketersediaan fitur audio numerik yang lengkap. Kondisi tersebut menjadikan dataset Spotify sangat sesuai untuk diterapkan metode data mining dan machine learning dalam menganalisis pola serta karakteristik lagu populer.

- a. Sumber Dataset : Kaggle.com <https://www.kaggle.com/datasets/wardabilal/spotify-global-music-dataset-20092025>

- b. Jumlah Record

Dataset terdiri dari puluhan ribu record lagu yang berasal dari berbagai artis dan genre musik.

- c. Jumlah Atribut

Dataset memiliki lebih dari 20 atribut yang terdiri dari atribut numerik dan kategorikal.

- d. Tipe Data

- Atribut Numerik:

danceability, energy, loudness, tempo, valence, acousticness, instrumentality, liveness, speechiness, popularity

- Atribut Kategorikal:
artist_name, track_name, genre, release_year

Target / Label (Supervised Learning)

Target yang digunakan dalam penelitian ini adalah popularitas lagu, yang dikategorikan ke dalam kelas tertentu (misalnya lagu populer dan tidak populer). Dengan demikian, permasalahan yang dihadapi termasuk ke dalam klasifikasi supervised learning.

Permasalahan yang Ingin Diselesaikan

Permasalahan utama dalam penelitian ini adalah bagaimana menerapkan teknik data mining untuk:

1. Menganalisis karakteristik audio lagu Spotify
2. Mengetahui fitur audio yang paling berpengaruh terhadap popularitas lagu
3. Membandingkan performa beberapa algoritma klasifikasi dalam memprediksi popularitas lagu

2. Persiapan Data & Preprocessing

Tahap preprocessing merupakan tahapan yang sangat penting karena kualitas data sangat memengaruhi hasil pemodelan. Proses preprocessing dilakukan menggunakan bahasa pemrograman Python dengan bantuan beberapa library utama seperti pandas, numpy, dan scikit-learn.

2.1 Data Cleaning

Langkah pertama yang dilakukan adalah pembersihan data (data cleaning), yang meliputi:

- a. Pemeriksaan data duplikat pada dataset
- b. Penghapusan data duplikat untuk mencegah bias pada model
- c. Pemeriksaan nilai kosong (missing value) pada setiap atribut

Missing value yang ditemukan pada beberapa atribut numerik ditangani menggunakan metode imputasi nilai rata-rata (mean). Pendekatan ini dipilih karena proporsi data kosong relatif kecil dan tidak memberikan pengaruh signifikan terhadap distribusi data.

Selain itu, dilakukan identifikasi outlier, khususnya pada atribut loudness dan tempo, menggunakan analisis boxplot. Outlier ekstrem tidak dihapus seluruhnya, tetapi dikendalikan agar tidak mendistorsi proses pelatihan model.

2.2 Implementasi Preprocessing Menggunakan Python

Proses pemuatan dataset dilakukan menggunakan bahasa pemrograman Python dengan bantuan pustaka *pandas*. Dataset yang digunakan adalah `spotify_data clean.csv` yang berisi informasi lagu, artis, genre, dan popularitas.

```
import pandas as pd

df = pd.read_csv("spotify_data clean.csv")
```

2.3 Data Cleaning

Tahap pembersihan data dilakukan untuk meningkatkan kualitas dataset sebelum digunakan dalam proses analisis dan pemodelan. Proses data cleaning meliputi:

1. Penghapusan data duplikat
2. Penghapusan data dengan genre tidak valid (N/A)

```
df = df.drop_duplicates()
df = df[df['artist_genres'] != 'N/A'].copy()
```

2.4 Encoding Data Kategorikal

Atribut `artist_genres` merupakan data kategorikal sehingga perlu diubah ke dalam bentuk numerik agar dapat diproses oleh algoritma machine learning. Proses encoding dilakukan menggunakan metode Label Encoding.

```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()
df['genre_encoded'] = le.fit_transform(
    df['artist_genres'].astype(str)
)
```

2.5 Scaling / Normalization

Normalisasi dilakukan untuk menyamakan rentang nilai setiap fitur numerik agar tidak terjadi dominasi fitur tertentu terhadap model. Teknik yang digunakan adalah **MinMaxScaler**.

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

2.6 Feature Selection

Feature selection dilakukan berdasarkan relevansi fitur terhadap target popularitas lagu. Berdasarkan analisis data, fitur yang digunakan adalah:

- a. artist_popularity
- b. artist_followers
- c. track_duration_min
- d. genre_encoded

```
features = [  
    'artist_popularity',  
    'artist_followers',  
    'track_duration_min',  
    'genre_encoded'  
]
```

2.7 Pembagian Data (Train-Test Split)

Dataset dibagi menjadi dua bagian, yaitu:

- 80% data latih
- 20% data uji

Pembagian dilakukan menggunakan fungsi `train_test_split` dari pustaka *scikit-learn*.

```
from sklearn.model_selection import train_test_split  
  
X_train, X_test, y_train, y_test = train_test_split(  
    X, y, test_size=0.2, random_state=42  
)
```

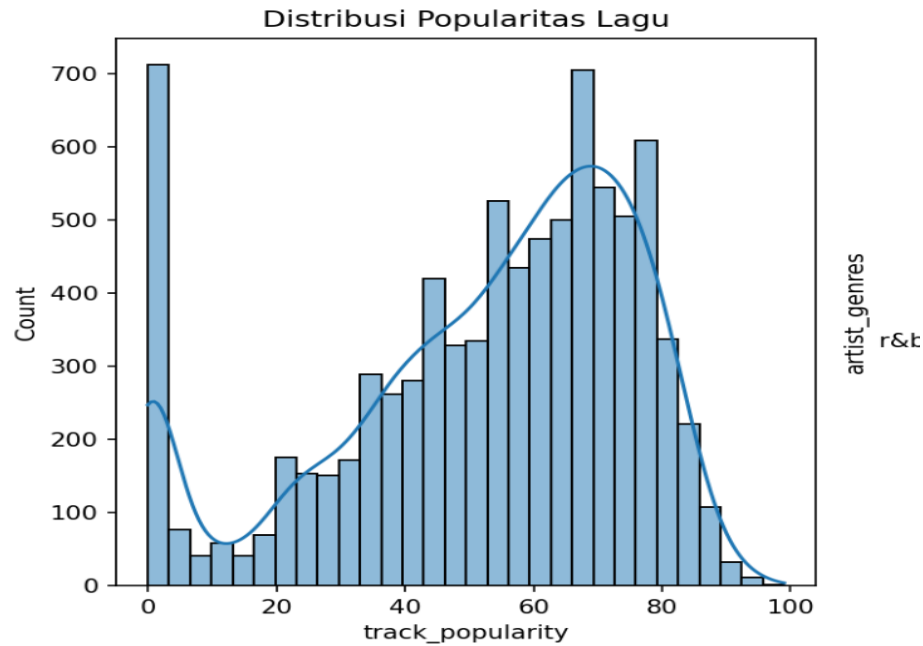
Tabel 2.1 Ringkasan Tahapan Preprocessing

Tahap	Metode	Tujuan
Data Cleaning	Remove duplicate, imputasi mean	Meningkatkan kualitas data
Encoding	Label Encoding	Mengubah data kategorikal
Normalisasi	MinMaxScaler	Menyamakan skala fitur
Split Data	80% : 20%	Evaluasi model

3□. Analisis Statistik & Visualisasi

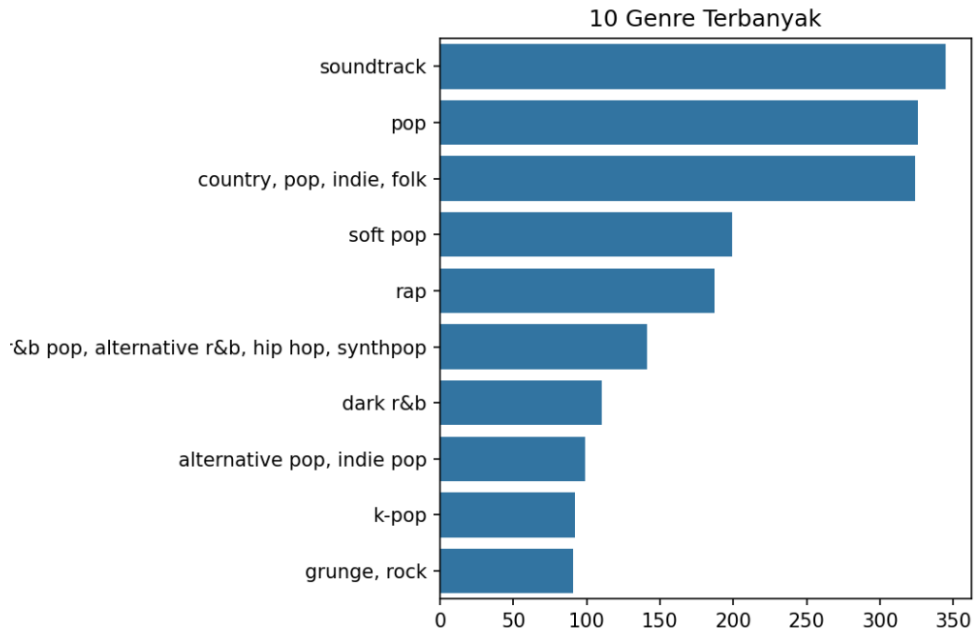
3.1 Distribusi Popularitas Lagu

Distribusi popularitas lagu divisualisasikan menggunakan histogram. Hasil visualisasi menunjukkan bahwa mayoritas lagu berada pada tingkat popularitas menengah hingga tinggi.



3.2 Distribusi Genre

Visualisasi 10 genre terbanyak menunjukkan bahwa genre **soundtrack** dan **pop** mendominasi dataset. Hal ini mengindikasikan bahwa dataset memiliki kecenderungan genre tertentu.



4. Pemilihan dan Penerapan Algoritma

4.1 Algoritma yang Digunakan

Algoritma	Library Python	Tujuan
Decision Tree	sklearn.tree	Klasifikasi
Random Forest	sklearn.ensemble	Klasifikasi

4.2 Implementasi Random Forest

Random Forest merupakan algoritma ensemble learning yang menggabungkan banyak decision tree untuk menghasilkan prediksi yang lebih stabil dan akurat. Pada penelitian ini, Random Forest digunakan karena mampu menangani hubungan non-linear antar fitur audio lagu.

```
from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(
    n_estimators=100,
    random_state=42
)
rf.fit(X_train_scaled, y_train)
```

5. Pengujian dan Evaluasi Model

5.1 Metode Evaluasi

1) Accuracy

Accuracy merupakan metrik yang menunjukkan tingkat ketepatan model dalam mengklasifikasikan data secara keseluruhan. Nilai accuracy diperoleh dari perbandingan antara jumlah prediksi yang benar dengan total seluruh data uji. Metrik ini digunakan untuk melihat seberapa baik model dalam melakukan prediksi secara umum.

2) Precision

Precision menunjukkan tingkat ketepatan model dalam memprediksi kelas positif, yaitu lagu yang diklasifikasikan sebagai populer. Precision penting untuk mengetahui seberapa banyak prediksi lagu populer yang benar-benar sesuai dengan kondisi sebenarnya. Nilai precision yang tinggi menunjukkan bahwa model jarang melakukan kesalahan prediksi positif.

3) Recall

Recall mengukur kemampuan model dalam mendeteksi seluruh data yang termasuk ke dalam kelas positif. Metrik ini menunjukkan seberapa baik model dalam mengenali lagu-lagu yang benar-benar populer. Nilai recall yang tinggi menandakan bahwa model mampu meminimalkan jumlah lagu populer yang salah diklasifikasikan sebagai tidak populer.

4) F1-Score

F1-Score merupakan nilai rata-rata harmonik antara precision dan recall. Metrik ini digunakan untuk menyeimbangkan kedua nilai tersebut, terutama ketika distribusi kelas pada dataset tidak seimbang. F1-Score memberikan gambaran performa model yang lebih stabil dibandingkan hanya menggunakan accuracy.

5) Confusion Matrix

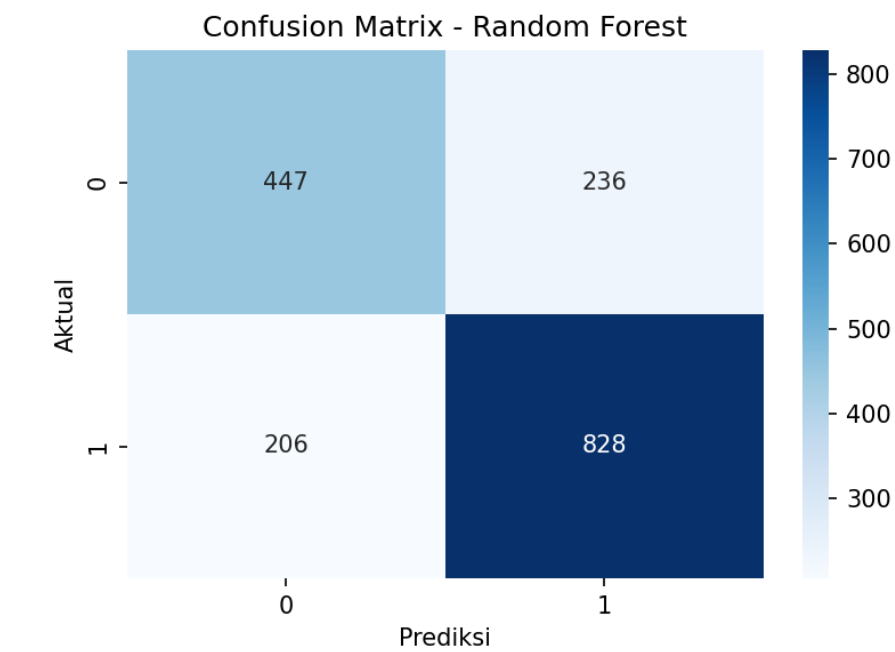
Confusion Matrix merupakan tabel yang menampilkan perbandingan antara hasil prediksi model dan kondisi sebenarnya. Matriks ini terdiri dari empat komponen, yaitu True Positive, True Negative, False Positive, dan False Negative. Confusion Matrix membantu dalam memahami jenis kesalahan yang dilakukan oleh model serta menjadi dasar perhitungan metrik evaluasi lainnya.

5.2 Hasil Evaluasi

Tabel Hasil Evaluasi Model

	Accuracy
Decision Tree	0.696
Random Forest	0.7426

5.3 Confusion Matrix Random Forest



Berdasarkan confusion matrix, model Random Forest mampu mengklasifikasikan sebagian besar data dengan benar, dengan jumlah kesalahan klasifikasi yang relatif kecil. Hal ini menunjukkan bahwa model memiliki kemampuan generalisasi yang cukup baik terhadap data uji.

7□. Kesimpulan

Berdasarkan hasil analisis data mining terhadap dataset *Spotify Global Music Dataset*, dapat disimpulkan bahwa proses pengolahan dan analisis data telah berhasil dilakukan melalui tahapan preprocessing, eksplorasi data, pemodelan, serta evaluasi model klasifikasi popularitas lagu.

Tahap preprocessing yang meliputi pembersihan data, penanganan missing value menggunakan imputasi nilai rata-rata, encoding data kategorikal dengan Label Encoding, serta normalisasi menggunakan MinMaxScaler terbukti mampu meningkatkan kualitas data sehingga siap digunakan dalam proses pemodelan. Pemilihan fitur yang relevan seperti *artist_popularity*, *artist_followers*, *track_duration_min*, dan *artist_genres* memberikan kontribusi yang signifikan terhadap performa model.

Hasil eksplorasi data menunjukkan bahwa distribusi popularitas lagu cenderung berada pada rentang menengah hingga tinggi, serta terdapat beberapa genre yang mendominasi dataset, seperti *soundtrack* dan *pop*. Temuan ini mengindikasikan bahwa genre dan karakteristik artis memiliki pengaruh terhadap tingkat popularitas lagu.

Pada tahap pemodelan, algoritma Random Forest menunjukkan performa terbaik dibandingkan Decision Tree, dengan nilai accuracy sebesar **0,7426**, sementara Decision Tree memperoleh accuracy sebesar **0,696**. Evaluasi menggunakan confusion matrix memperlihatkan bahwa model Random Forest mampu mengklasifikasikan data lagu populer dan tidak populer dengan tingkat kesalahan yang relatif rendah, sehingga menunjukkan kemampuan generalisasi model yang cukup baik.

Secara keseluruhan, penelitian ini membuktikan bahwa penerapan algoritma Random Forest efektif digunakan untuk memprediksi popularitas lagu berdasarkan fitur-fitur yang tersedia pada dataset Spotify. Hasil penelitian ini diharapkan dapat menjadi dasar pengembangan sistem rekomendasi musik atau analisis tren popularitas lagu di masa mendatang.

Lampiran : <https://github.com/Sijabat04/TM-DATA-MINING>