

Зелинский Даниил Михайлович, ИУ5-25М

1 вариант

О наборе данных `music_genre.csv`

Данный набор данных содержит информацию о музыкальных треках различных жанров. Каждая строка представляет собой один трек с его аудио- и мета-характеристиками, полученными с платформы Spotify. Датасет используется для задач классификации жанров и анализа аудиопараметров.

Общее количество признаков: 18

Типы признаков: числовые, категориальные

Объем: ~50 000 строк

Структура данных:

Название столбца	Описание
<code>instance_id</code>	Уникальный идентификатор трека
<code>artist_name</code>	Имя исполнителя
<code>track_name</code>	Название трека
<code>popularity</code>	Популярность трека (от 0 до 100)
<code>acousticness</code>	Насколько трек акустический (от 0 до 1)
<code>danceability</code>	Танцевальность трека (от 0 до 1)
<code>duration_ms</code>	Длительность трека в миллисекундах
<code>energy</code>	Энергичность трека (от 0 до 1)
<code>instrumentalness</code>	Вероятность, что трек инструментальный (от 0 до 1)
<code>key</code>	Музыкальный ключ (например, C#, F#)
<code>liveness</code>	Вероятность живой записи (от 0 до 1)
<code>loudness</code>	Громкость в децибелах (обычно отрицательное значение)
<code>mode</code>	Лад трека: мажор

Название столбца	Описание
	(Major) или минор (Minor)
speechiness	Наличие речи в треке (от 0 до 1)
tempo	Темп (ударов в минуту)
obtained_date	Дата получения записи
valence	Эмоциональная окраска трека (от 0 до 1)
music_genre	Жанр музыки (целевой признак)

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

file_path = "music_genre.csv"
data = pd.read_csv(file_path)

# --- ЗАДАЧА №1 ---
# Count (frequency) encoding для категориального признака 'music_genre'
count_encoding = data['music_genre'].value_counts().to_dict()
data['music_genre_count_encoded'] =
data['music_genre'].map(count_encoding)

print("Пример после Count Encoding:")
print(data[['music_genre', 'music_genre_count_encoded']].head())

Пример после Count Encoding:
  music_genre  music_genre_count_encoded
0  Electronic                5000.0
1  Electronic                5000.0
2  Electronic                5000.0
3  Electronic                5000.0
4  Electronic                5000.0

# --- ЗАДАЧА №21 ---
# Масштабирование по медиане для числового признака 'popularity'
median_value = data['popularity'].median()
data['popularity_scaled_median'] = data['popularity'] / median_value

print("\nМедианное значение popularity:", median_value)
print(data[['popularity', 'popularity_scaled_median']].head())

Медианное значение popularity: 45.0
  popularity  popularity_scaled_median
0         27.0                0.600000
```

1	31.0	0.688889
2	28.0	0.622222
3	34.0	0.755556
4	32.0	0.711111

ДОПОЛНИТЕЛЬНОЕ ЗАДАНИЕ

Построение pairplot по числовым признакам с hue='music_genre'

```
numeric_cols = [  
    'acousticness', 'danceability', 'duration_ms',  
    'energy', 'instrumentalness', 'valence'  
]
```

Приведение колонок к числовому типу и удаление некорректных значений

```
for col in numeric_cols:  
    data[col] = pd.to_numeric(data[col], errors='coerce')
```

```
pairplot_data = data[numeric_cols + ['music_genre']].replace([np.inf,  
-np.inf], np.nan).dropna()
```

```
sampled = pairplot_data.sample(n=1000, random_state=42)
```

```
sns.set_context("talk", font_scale=1.4)
```

```
sns.set_style("whitegrid")
```

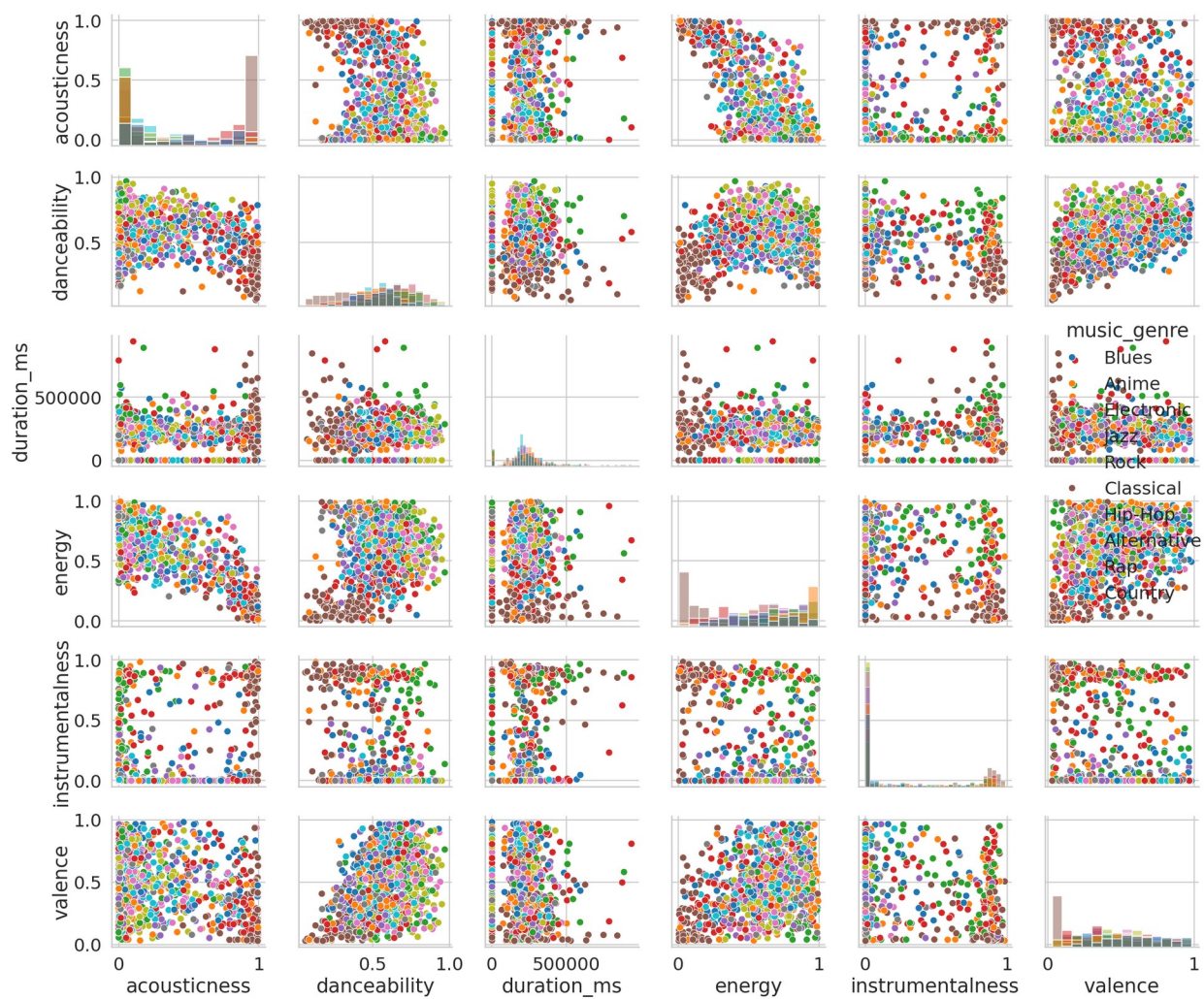
```
sns.pairplot(sampled, hue='music_genre', diag_kind='hist', height=3.2)
```

```
plt.suptitle("Pairplot по числовым признакам и жанрам", y=1.02)
```

```
plt.tight_layout()
```

```
plt.show()
```

Pairplot по числовым признакам и жанрам



Вывод

1. Count Encoding успешно выполнен для категориального признака 'music_genre'.
2. Масштабирование по медиане выполнено для числового признака 'popularity'.
3. Построен pairplot, показывающий взаимосвязи между числовыми признаками с учетом музыкального жанра.