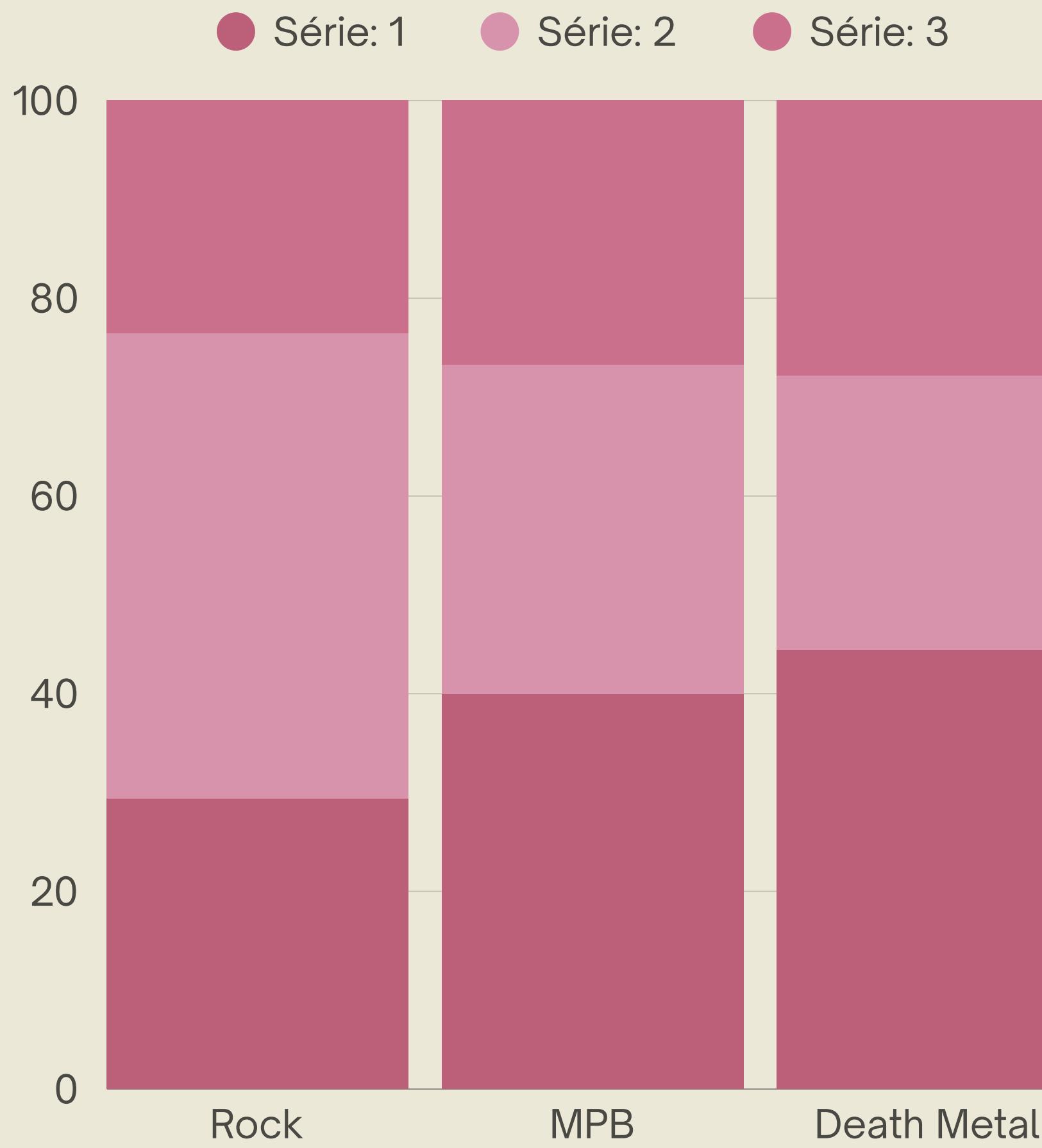


Introdução à Ciência de Dados

ANÁLISE DE FEATURES PARA A RECOMENDAÇÃO DE MÚSICAS POR GÊNERO

Carolina Barroco, Sofia Lahham,
Victória Marques



NOSSO OBJETIVO FINAL:



Compreender padrões entre as features de músicas, quando a diferenciação por gênero é importante, afim de testar a viabilidade de construir um sistema de recomendação inteiramente baseado em atributos da música e no seu respectivo gênero.

Portanto, uma proposta que visa diminuir infrações de privacidade do usuário, uma vez que se baseia integralmente em uma filtragem por conteúdo.

OBJETIVOS PARCIAIS:

(ORIENTAM AO OBJETIVO FINAL)

1. DEFINIÇÃO DO ESCOPO E PREPARAÇÃO DELE
2. ANÁLISES ESTATÍSTICAS POR FEATURE PARA INFERÊNCIAS
POR GÊNERO
3. ANÁLISE DA EFICÁCIA DO USO DE RELAÇÕES LINEARES
PARA O AGRUPAMENTO DE MÚSICAS
4. ANÁLISE DA CAPACIDADE DE CLASSIFICAÇÃO POR
GÊNERO ATRAVÉS DA DEFINIÇÃO DE THRESHOLDS, AFIM DA
VIABILIZAÇÃO DE AGRUPAMENTOS PRECISOS

OBJETIVOS PARCIAIS: (ORIENTAM AO OBJETIVO FINAL)

5. COMPARAÇÃO ENTRE ABORDAGENS
CONSIDERANDO A PUREZA DOS
AGRUPAMENTOS.

6. REALIZAÇÃO DE RECOMENDAÇÕES
CONSIDERANDO
GÊNEROS E PROXIMIDADES DE MÚSICAS
DIMENSIONALMENTE

1. DEFINIÇÃO DO ESCOPO E PREPARAÇÃO DELE

- O nosso conjunto de dados possui **3.000 faixas musicais**, classificadas em três gêneros: MPB, Rock e Death Metal.
- Cada uma das faixa é descrita por 17 atributos técnicos como *dançabilidade, energia, valência, acústica, tempo* e outros.

Info
3000 instances
17 features
Target with 3 values
4 meta attributes (0.0 % missing data)

Variables
 Show variable labels (if present)
 Visualize numeric values
 Color by instance classes

Selection
 Select full rows

>

	track_genre	track_id	artists	album_name	track_name	Number	Unnamed: 0	popularity	duration_ms	explicit	danceability	energy	key	loudness	mode
1	mpb	6g2BiiVQqY5v1...	Rodrigo Amara...	Tuyo (Narcos T...	Tuyo (Narcos T...	0	74000	66	151565	False	0.7650	0.497	1	-7806.000	0
2	mpb	6Dc2tCivms1s2...	Rodrigo Amara...	Tuyo (Narcos T...	Tuyo (Narcos T...	1	74001	65	89293	False	0.7200	0.422	1	-13338.000	0
3	mpb	76HOOCFt3IKV...	Djavan	Djavan "Ao Vivo"	Azul (Ao Vivo)	2	74002	49	259066	False	0.6320	0.651	7	-8658.000	0
4	mpb	4crctLJMJKeku...	Nicolas Candid...	Apaga a Luz (fe...	Apaga a Luz (fe...	3	74003	49	216446	False	0.6100	0.42	4	-10889.000	0
5	mpb	0XXwP0EmpOE...	Fábio Jr.	Fabio Jr.	Pareço um Me...	4	74004	50	273906	False	0.5800	0.33	7	-13703.000	1
6	mpb	4yDSMcUP1gL4...	Vanessa Da Mata	Caixinha de Mú...	Amado - Ao Vivo	5	74005	49	244426	False	0.3570	0.469	0	-11003.000	1
7	mpb	1KMBAzA8X44...	Lagum	MEMÓRIAS (de...	FESTA JOVEM	6	74006	49	112052	False	0.8190	0.438	11	-10611.000	1
8	mpb	1gDa6mXtwP3x...	Jorge Aragão	Ao vivo	Identidade - Ao...	7	74007	50	231866	False	0.5810	0.721	9	-9.860	0
9	mpb	4k2AjIWrulygEi...	Os Paralamas D...	Acústico (Live)	Tendo A Lua - ...	8	74008	51	200826	False	0.4920	0.794	5	-5823.000	1
10	mpb	5wSagCWfINwJ...	Pato Fu	Gol de Quem?	Sobre o Tempo	9	74009	52	207533	False	0.6330	0.551	11	-12294.000	1
11	mpb	6vf5x73BHel7hz...	Antônio Carlos ...	The Greatest Ja...	Look To The Sky	10	74010	0	137933	False	0.4990	0.215	11	-17369.000	0
12	mpb	3UzG9lNmUswP...	Tim Maia	Chill in Brazil	Acenda o farol	11	74011	0	192026	False	0.5740	0.883	9	-6686.000	0
13	mpb	0DIYusRuEUeZa...	Jorge Vercillo	Monalisa	Monalisa	12	74012	48	222732	False	0.7730	0.873	1	-4955.000	0
14	mpb	6OSs0dmcVqV...	Jota Quest	Acústico Jota Q...	Fácil - Acústico	13	74013	54	229540	False	0.5560	0.537	7	-10761.000	1
15	mpb	7dC7qps73xUo...	Tim Maia	Ressaqueirinha	Se eu lembro fa...	14	74014	0	230706	False	0.4460	0.752	9	-8252.000	0
16	mpb	6oly99hlunwn...	Gonzaguinha	Meus Momentos	Eu Apenas Que...	15	74015	50	155466	False	0.5590	0.505	8	-9446.000	1
17	mpb	7K0Nelb1gfXMj...	Jota Quest	Acústico Jota Q...	Amor Maior - A...	16	74016	51	235385	False	0.5820	0.538	2	-9395.000	0
18	mpb	3LmYMHpOQa...	Antônio Carlos ...	Chill in Brazil	Triste	17	74017	1	181600	False	0.5680	0.396	8	-13921.000	1

2. ANÁLISES ESTATÍSTICAS POR FEATURE PARA INFERÊNCIAS POR GÊNERO

O QUE OS NÚMEROS MOSTRAM:

- Para entender os dados, foram calculadas **média**, **mediana**, **desvio padrão**, **valor mínimo** e **máximo** para as principais variáveis.
- A análise foi feita separadamente para cada gênero musical.

Com isso, percebemos que:

- O **Death Metal** tem altíssima energia e pouca variação interna.
- O **MPB** se destaca por maior valência (músicas mais alegres) e *danceability*.
- O **Rock** possui valores medianos e maior diversidade sonora.

danceability						tempo					
track_genre	média	mediana	desvio padrão	mínimo	máximo	track_genre	média	mediana	desvio padrão	mínimo	máximo
death-metal	0,37	0,37	0,12	0,05	0,63	death-metal	111279,3	114977,5	45993,44	67,96	201018
mpb	0,57	0,58	0,13	0,22	0,93	mpb	108173,68	112838	45675,93	75,12	206641
rock	0,54	0,55	0,14	0,11	0,89	rock	115985,06	120041	44736,54	75,05	207478

energy						popularity					
track_genre	média	mediana	desvio padrão	mínimo	máximo	track_genre	média	mediana	desvio padrão	mínimo	máximo
death-metal	0,93	0,96	0,09	0,24	1	death-metal	32,17	25	14,13	0	73
mpb	0,58	0,59	0,2	0	0,97	mpb	40,79	42	8,99	0	66
rock	0,68	0,7	0,19	0,09	0,99	rock	19	0	32,54	0	96

valence					
track_genre	média	mediana	desvio padrão	mínimo	máximo
death-metal	0,25	0,22	0,16	0,02	0,93
mpb	0,56	0,55	0,23	0,07	0,98
rock	0,54	0,55	0,23	0,06	0,98

Tabela com as principais estatísticas por gênero para as variáveis energy, valence, danceability, tempo e popularity.

UM POUCO MAIS PERTO, ANALISAMOS QUE...

DEATH METAL :

- Tem a maior média de energia (0.93), com baixa variação (± 0.09) → músicas intensas e consistentes.
- Valência muito baixa (0.25) → músicas mais "sombrias".
- Dançabilidade também é menor (0.37) → menos dançantes.
- Popularidade média baixa (32), sendo a segunda maior do nosso dataset.

MPB:

- Valência mais alta (0.56) → músicas mais alegres.
- Dançabilidade também alta (0.57).
- Menor desvio padrão em popularidade (± 9) → ou seja, faixas com mais regularidade em popularidade, sendo a mais popular.

ROCK:

- Costuma ter valores medianos em quase tudo.
- Maior desvio padrão em popularidade (± 32) → ou seja, há músicas muito famosas e outras quase desconhecidas.
- Dançabilidade (0.54) e valência (0.54) mostram equilíbrio.

danceability						tempo					
track_genre	média	mediana	desvio padrão	mínimo	máximo	track_genre	média	mediana	desvio padrão	mínimo	máximo
death-metal	0,37	0,37	0,12	0,05	0,63	death-metal	111279,3	114977,5	45993,44	67,96	201018
mpb	0,57	0,58	0,13	0,22	0,93	mpb	108173,68	112838	45675,93	75,12	206641
rock	0,54	0,55	0,14	0,11	0,89	rock	115985,06	120041	44736,54	75,05	207478
energy						popularity					
track_genre	média	mediana	desvio padrão	mínimo	máximo	track_genre	média	mediana	desvio padrão	mínimo	máximo
death-metal	0,93	0,96	0,09	0,24	1	death-metal	32,17	25	14,13	0	73
mpb	0,58	0,59	0,2	0	0,97	mpb	40,79	42	8,99	0	66
rock	0,68	0,7	0,19	0,09	0,99	rock	19	0	32,54	0	96
valence											
track_genre	média	mediana	desvio padrão	mínimo	máximo	track_genre	média	mediana	desvio padrão	mínimo	máximo
death-metal	0,25	0,22	0,16	0,02	0,93	death-metal	32,17	25	14,13	0	73
mpb	0,56	0,55	0,23	0,07	0,98	mpb	40,79	42	8,99	0	66
rock	0,54	0,55	0,23	0,06	0,98	rock	19	0	32,54	0	96

DISTRIBUIÇÃO: ENERGY

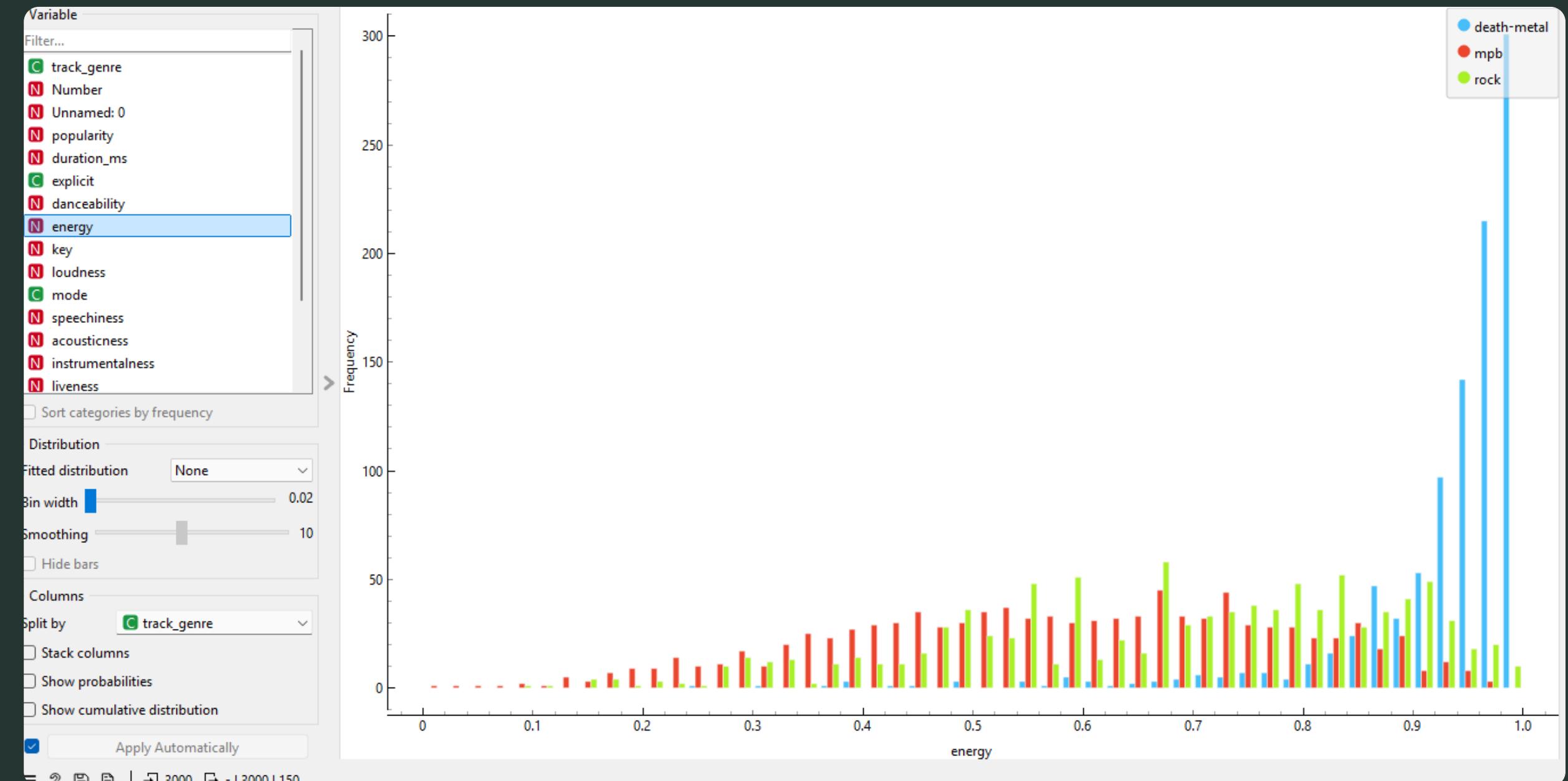
ENERGY POR GÊNERO

- Este gráfico nos mostra como a energia das músicas se distribui entre os três gêneros.

- O **Death Metal (azul)** apresenta uma concentração na faixa de energia próxima de 1.0, indicando que suas músicas são intensas e consistentes nesse aspecto.

- Já o **MPB (vermelho)** e o **Rock (verde)** possuem distribuições mais variadas, com faixas desde moderadas até intensas, o que mostra maior diversidade interna nesses estilos.

- Nessa imagem podemos observar a importância da variável energy para distinguir os gêneros, especialmente o Death Metal.

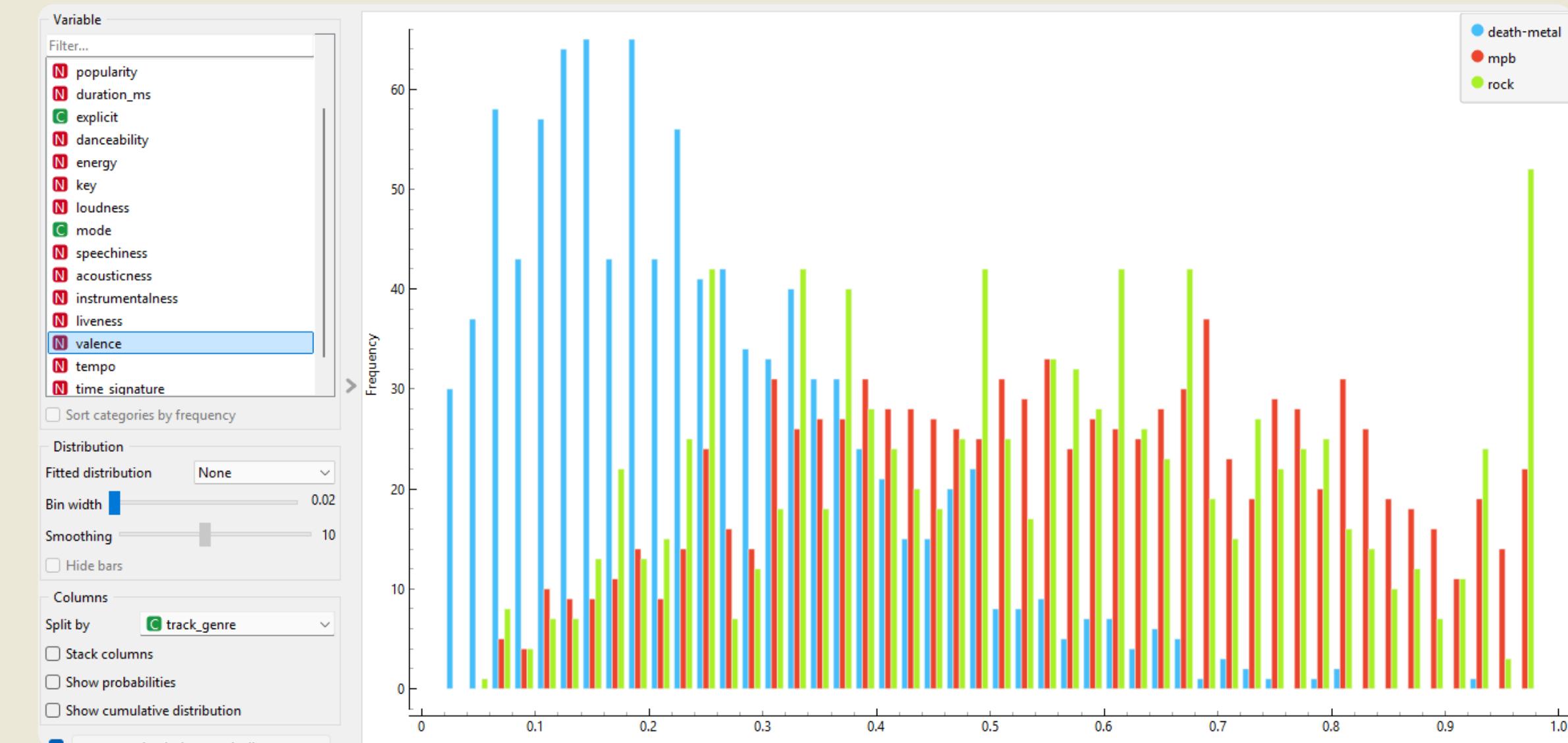


Death Metal domina os níveis máximos de energia, enquanto MPB e Rock variam entre intensidade baixa e moderada

DISTRIBUIÇÃO: VALÊNCIA

VALÊNCIA POR GÊNERO

- Já nesse gráfico observa-se que a valência – que indica o tom emocional da música – varia entre os gêneros.
- O **MPB (vermelho)** possui uma distribuição equilibrada, com destaque para valores médios e altos, indicando faixas mais alegres ou positivas.



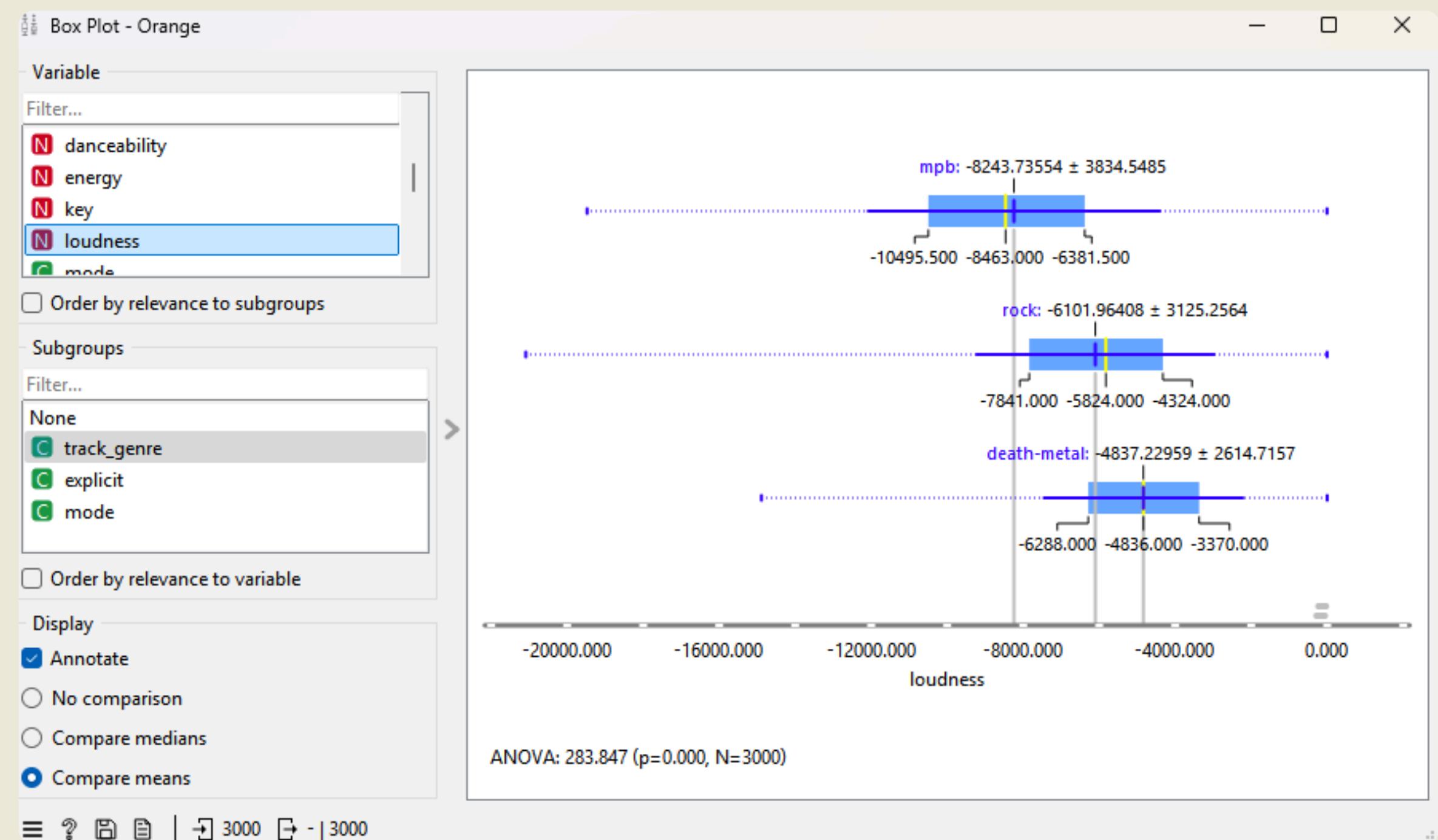
- O **Death Metal (azul)** se concentra nos níveis mais baixos de valência, refletindo músicas emocionalmente mais neutras ou sombrias.

- Já o **Rock (verde)** apresenta ampla variação, com faixas que vão desde melancólicas até muito alegres.
- Essa variável é essencial para compreender o clima emocional predominante em cada estilo.

BOX PLOT: LOUDNESS

VOLUME DAS FAIXAS POR GÊNERO

- O gráfico de box plot mostra a variação dos níveis de loudness (volume médio da faixa) por gênero musical.
- O **MPB** apresenta o volume mais baixo entre os três gêneros, com média próxima de -8.200 e ampla dispersão, indicando faixas mais suaves e variadas.
- O **Rock** possui volume mais alto que o MPB, com média em torno de -6.100, e também uma variação considerável.
- O **Death Metal** é o gênero com o volume mais alto (mais próximo de 0) e com menor variação, o que reflete a uniformidade e intensidade sonora típica desse estilo.
- Isso confirma que loudness é uma variável relevante para diferenciar os gêneros, especialmente o Death Metal.



MPB apresenta o menor volume médio, enquanto o Death Metal é o mais alto e uniforme.

CORRELAÇÕES

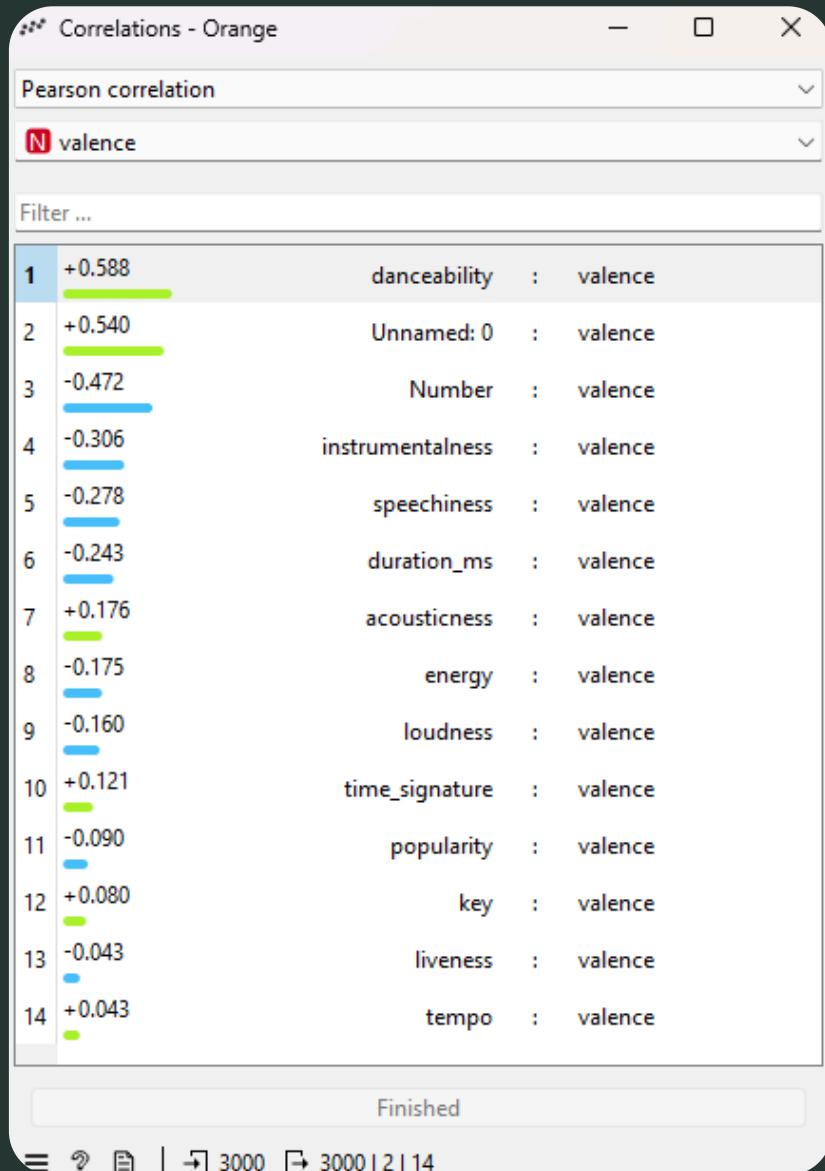
RELAÇÕES ENTRE VARIÁVEIS

PRINCIPAIS RELAÇÕES ENTRE ATRIBUTOS:

- **Acústico x Energia:** apresentam correlação negativa forte, o que significa que músicas com características mais acústicas tendem a ser menos energéticas – algo esperado em estilos mais calmos como MPB.
- **Energia x Volume:** mostram correlação positiva alta, indicando que quanto mais intensa (energética) a faixa, maior costuma ser seu volume. Essa relação é muito comum em gêneros como o Death Metal.
- **Conclusão:** o Death Metal concentra faixas com alta energia e volume, enquanto o MPB tende a ser mais acústico e menos intenso.

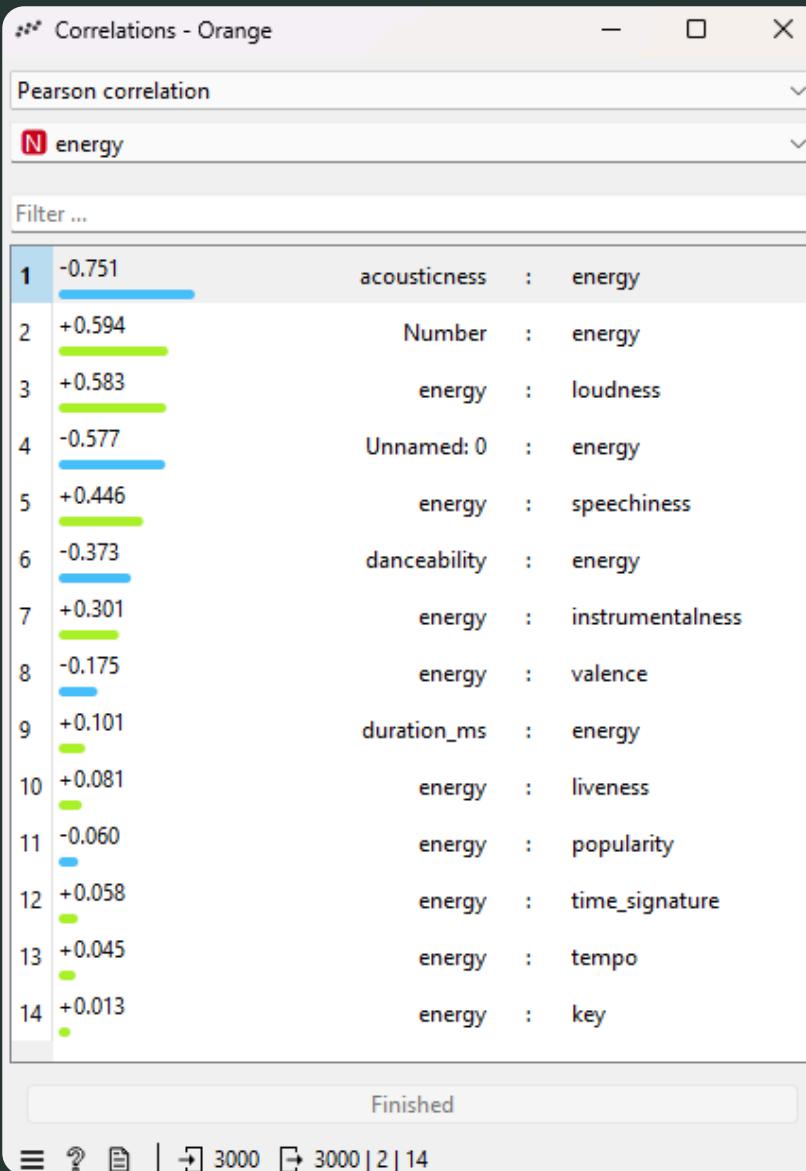
AMOSTRAS - CORRELAÇÃO

VALÂNCIA



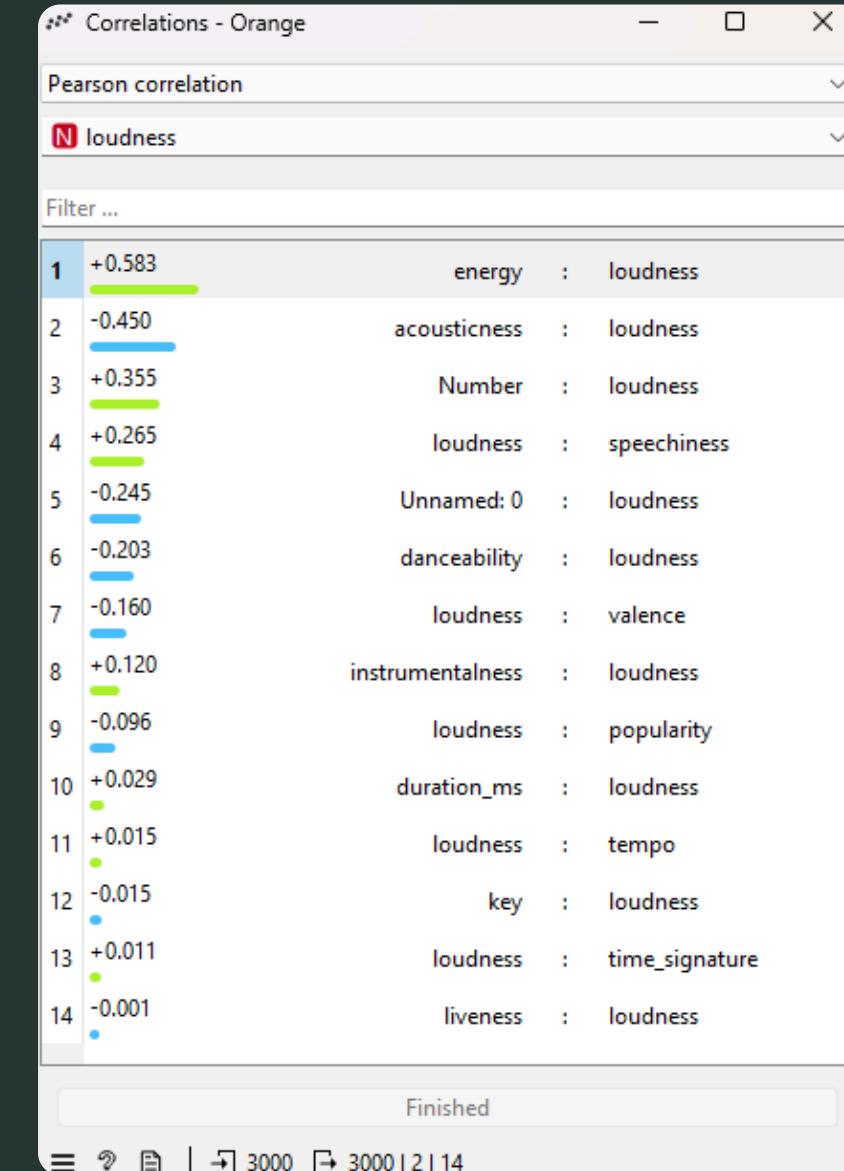
As músicas mais dançantes e instrumentais tendem a ter **maior valência**, ou seja, soam mais alegres. Já faixas com maior *speechiness* (fala) ou mais acústicas costumam ter **valência mais baixa**.

ENERGIA



Músicas com **maior energia** geralmente têm **menos** elementos acústicos e maior volume (*loudness*). A forte relação negativa com *acousticness* mostra que faixas intensas são pouco acústicas.

VOLUME



O **volume da música está fortemente ligado à sua energia**. Músicas mais acústicas, por outro lado, são menos altas. *Speechiness* também aparece como fator relevante, com correlação negativa moderada.

3. ANÁLISE DA EFICÁCIA DO USO DE MEDIDAS DE DISTÂNCIAS PARA O AGRUPAMENTO DE MÚSICAS

```

def hierarchical_clustering(df, iloc_lower, iloc_upper, num_clusters=106):
    X = df.iloc[:, iloc_lower:iloc_upper]

    scaler = MinMaxScaler()
    normalized_data = pd.DataFrame(scaler.fit_transform(X), columns=X.columns, index=X.index)

    Z = linkage(normalized_data, method='ward', metric='euclidean')

    num_clusters = 106
    clusters = fcluster(Z, num_clusters, criterion='maxclust')

    palette = sns.color_palette("Purples", num_clusters)
    cluster_to_color = {cluster: palette[i] for i, cluster in enumerate(np.unique(clusters))}

    row_colors = pd.Series(clusters, index=normalized_data.index).map(cluster_to_color)

    g = sns.clustermap(normalized_data,
                        row_linkage=Z,
                        col_cluster=False,
                        row_colors=row_colors,
                        cmap="magma",
                        figsize=(10, 80))

    legend_clusters = list(cluster_to_color.keys())
    legend_elements = [Patch(facecolor=cluster_to_color[cl], label=f'Cluster {cl}') for cl in legend_clusters]

    plt.gcf().legend(handles=legend_elements,
                      title='Cluster Colors',
                      loc='lower center',
                      ncol=7,
                      bbox_to_anchor=(0.5, -0.05))

    plt.show()

    result = X.copy()
    result['cluster'] = clusters

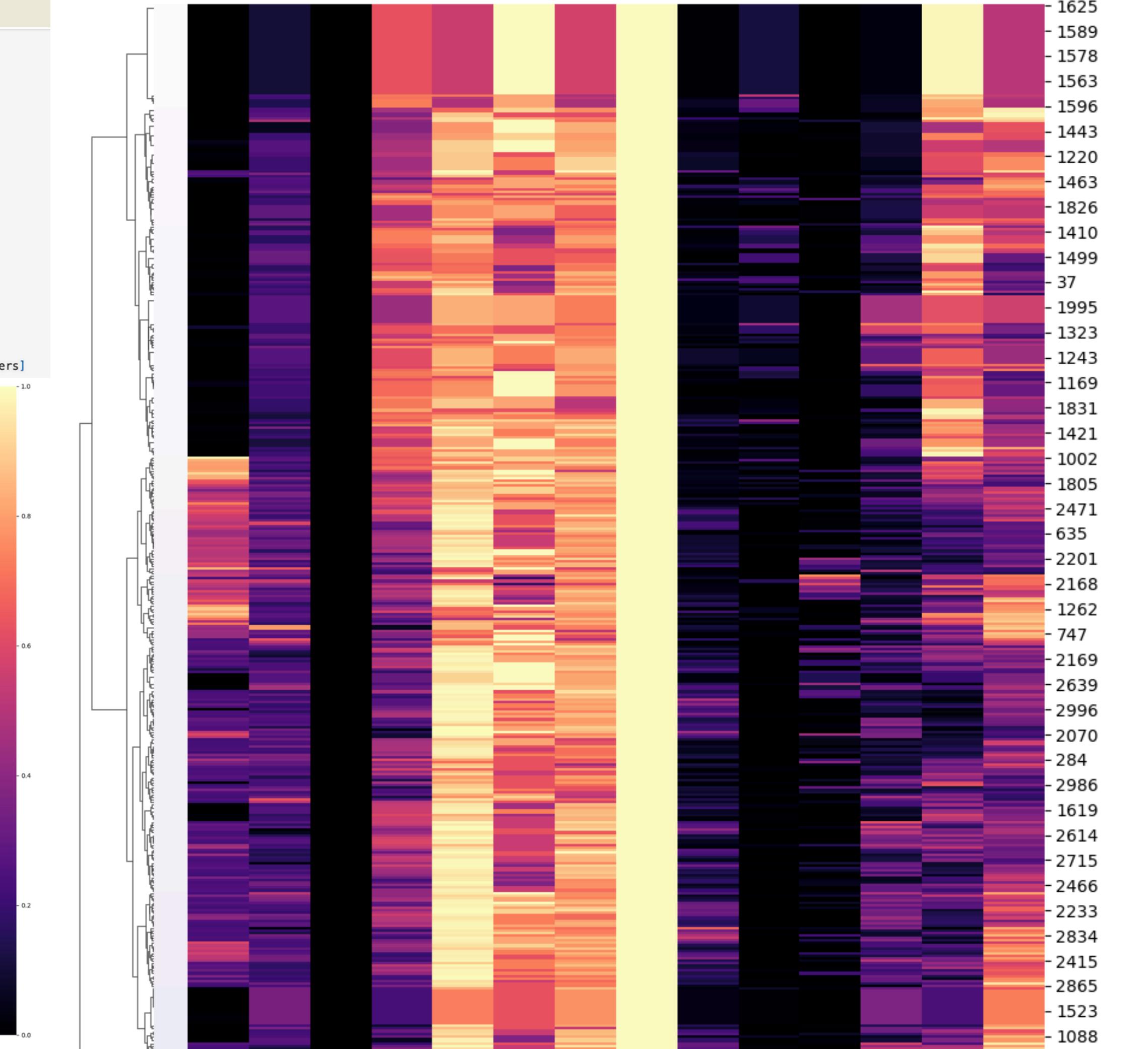
    # Step 9: Output result
    print(result.head())
    return result

result = hierarchical_clustering(df, 6, 20 )
print(result)

```

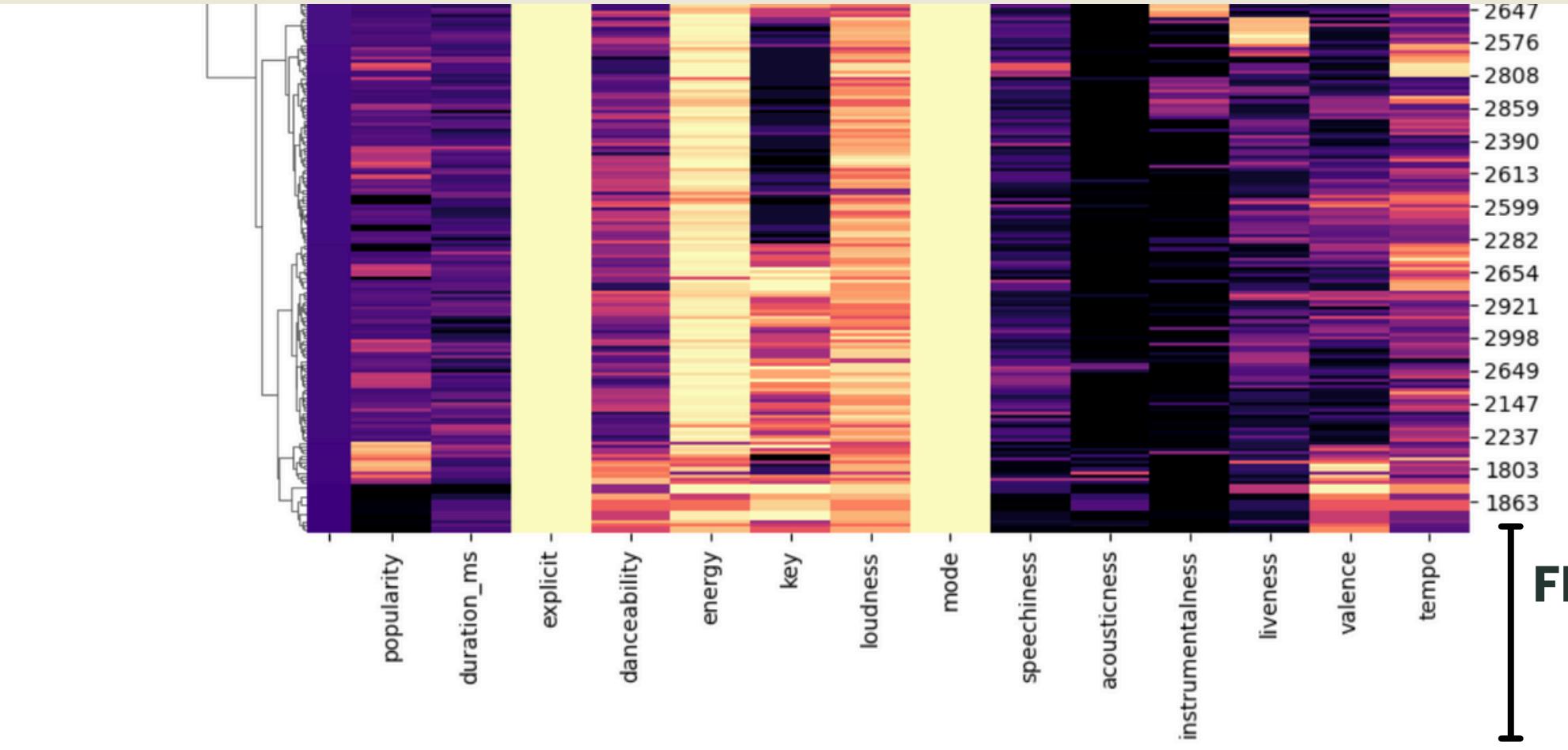
AGRUPAMENTO HIERARQUICO

CONSIDERANDO A DISTÂNCIA EUCLIDIANA



AGRUPAMENTO HIERARQUICO

CONSIDERANDO A DISTÂNCIA EUCLIDIANA



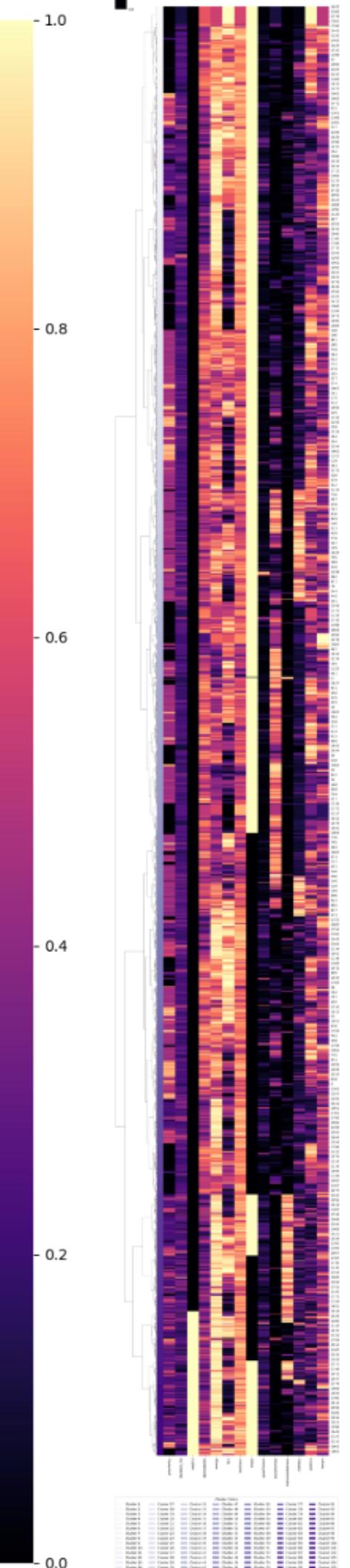
```
scaler = MinMaxScaler()  
normalized_data = pd.DataFrame(scaler.fit_transform(X),
```

NORMALIZAÇÃO: [0,1]

→ ID MÚSICA
FEATURES

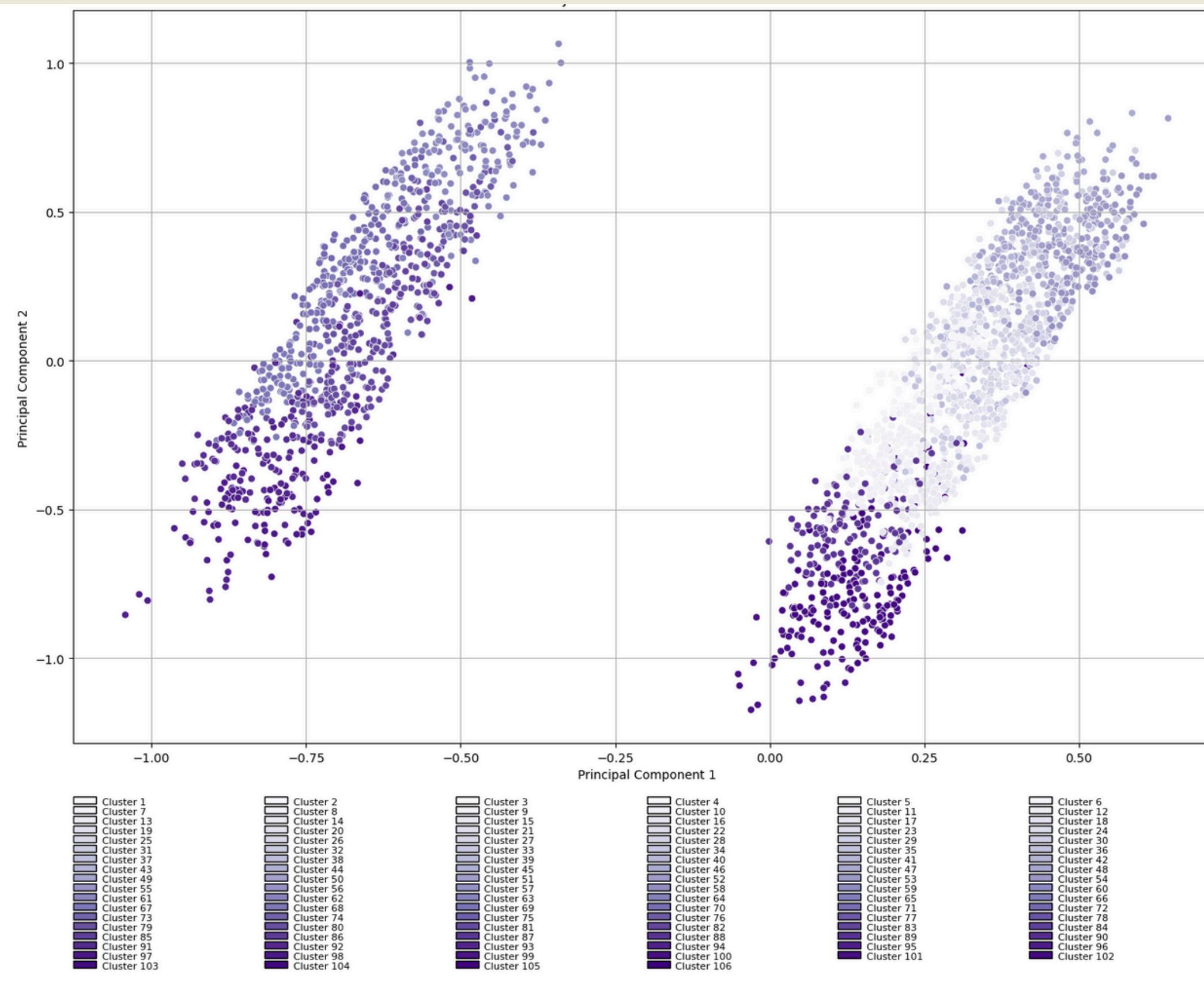
Cluster Colors																
Cluster 1	Cluster 17	Cluster 32	Cluster 47	Cluster 62	Cluster 77	Cluster 92										
Cluster 2	Cluster 18	Cluster 33	Cluster 48	Cluster 63	Cluster 78	Cluster 93										
Cluster 3	Cluster 19	Cluster 34	Cluster 49	Cluster 64	Cluster 79	Cluster 94										
Cluster 4	Cluster 20	Cluster 35	Cluster 50	Cluster 65	Cluster 80	Cluster 95										
Cluster 5	Cluster 21	Cluster 36	Cluster 51	Cluster 66	Cluster 81	Cluster 96										
Cluster 6	Cluster 22	Cluster 37	Cluster 52	Cluster 67	Cluster 82	Cluster 97										
Cluster 7	Cluster 23	Cluster 38	Cluster 53	Cluster 68	Cluster 83	Cluster 98										
Cluster 8	Cluster 24	Cluster 39	Cluster 54	Cluster 69	Cluster 84	Cluster 99										
Cluster 9	Cluster 25	Cluster 40	Cluster 55	Cluster 70	Cluster 85	Cluster 100										
Cluster 10	Cluster 26	Cluster 41	Cluster 56	Cluster 71	Cluster 86	Cluster 101										
Cluster 11	Cluster 27	Cluster 42	Cluster 57	Cluster 72	Cluster 87	Cluster 102										
Cluster 12	Cluster 28	Cluster 43	Cluster 58	Cluster 73	Cluster 88	Cluster 103										
Cluster 13	Cluster 29	Cluster 44	Cluster 59	Cluster 74	Cluster 89	Cluster 104										
Cluster 14	Cluster 30	Cluster 45	Cluster 60	Cluster 75	Cluster 90	Cluster 105										
Cluster 15	Cluster 31	Cluster 46	Cluster 61	Cluster 76	Cluster 91	Cluster 106										
Cluster 16																

Visualmente cerca de +200 agrupamentos. Com o intuito de reduzir esse número, eles foram reajustados para garantir que **106 clusters** existiram.



PCA + VISUALIZAÇÃO CLUSTERS

BUSCA POR RELAÇÕES LINEARES



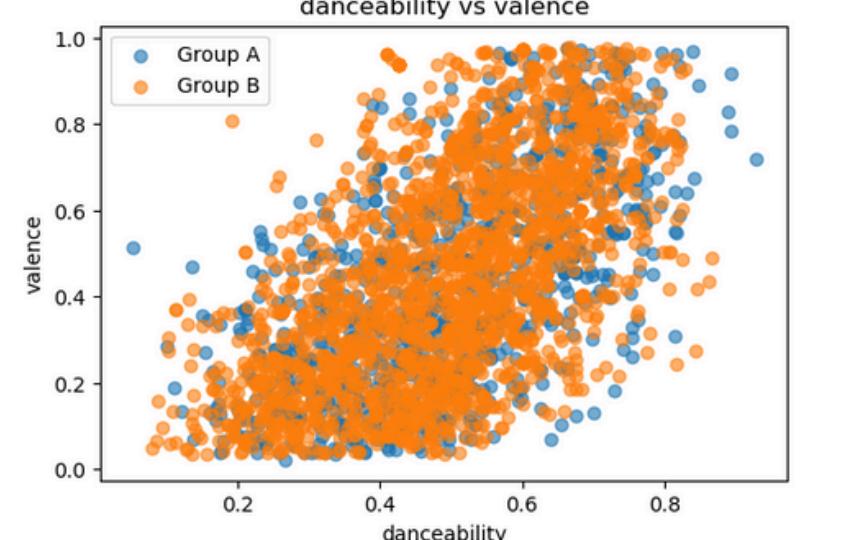
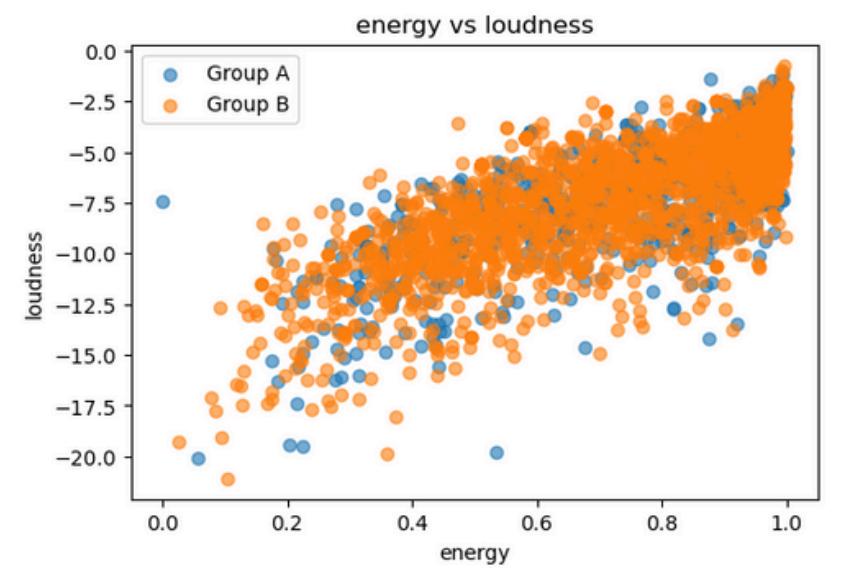
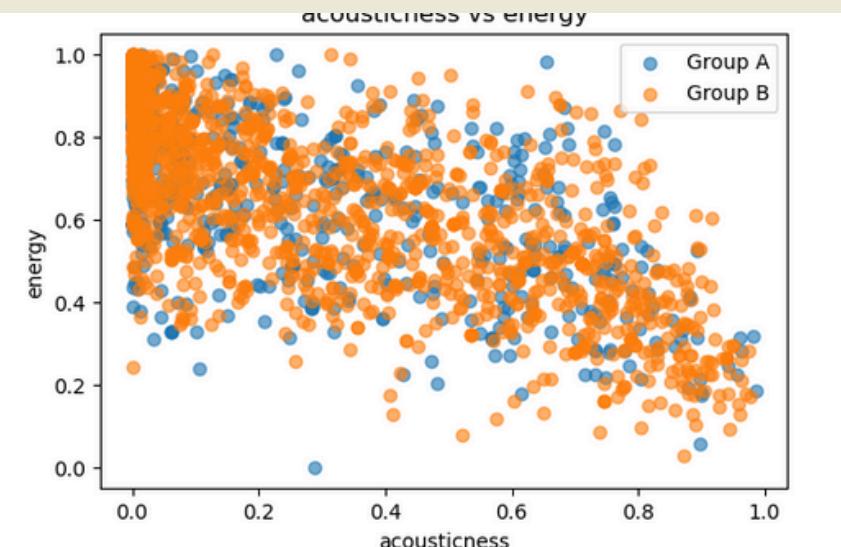
Uso da técnica PCA para visualização dos agrupamentos

PCA: Reduz os dados multidimensionais para 2D. Busca identificar e traduzir as relações lineares entre variáveis dos dados em múltiplas dimensões (matriz de covariância), encontrando novas direções (componentes) que melhor explicam a **variância** dos dados.

2 grupos foram formados?

PCA + VISUALIZAÇÃO CLUSTERS

BUSCA POR RELAÇÕES LINEARES



HIPÓTESE 1:

Grupos segregados em virtude da forte covariância negativa entre grupos entre variáveis.

FALSA: Grupos interpolados.

HIPÓTESE 2:

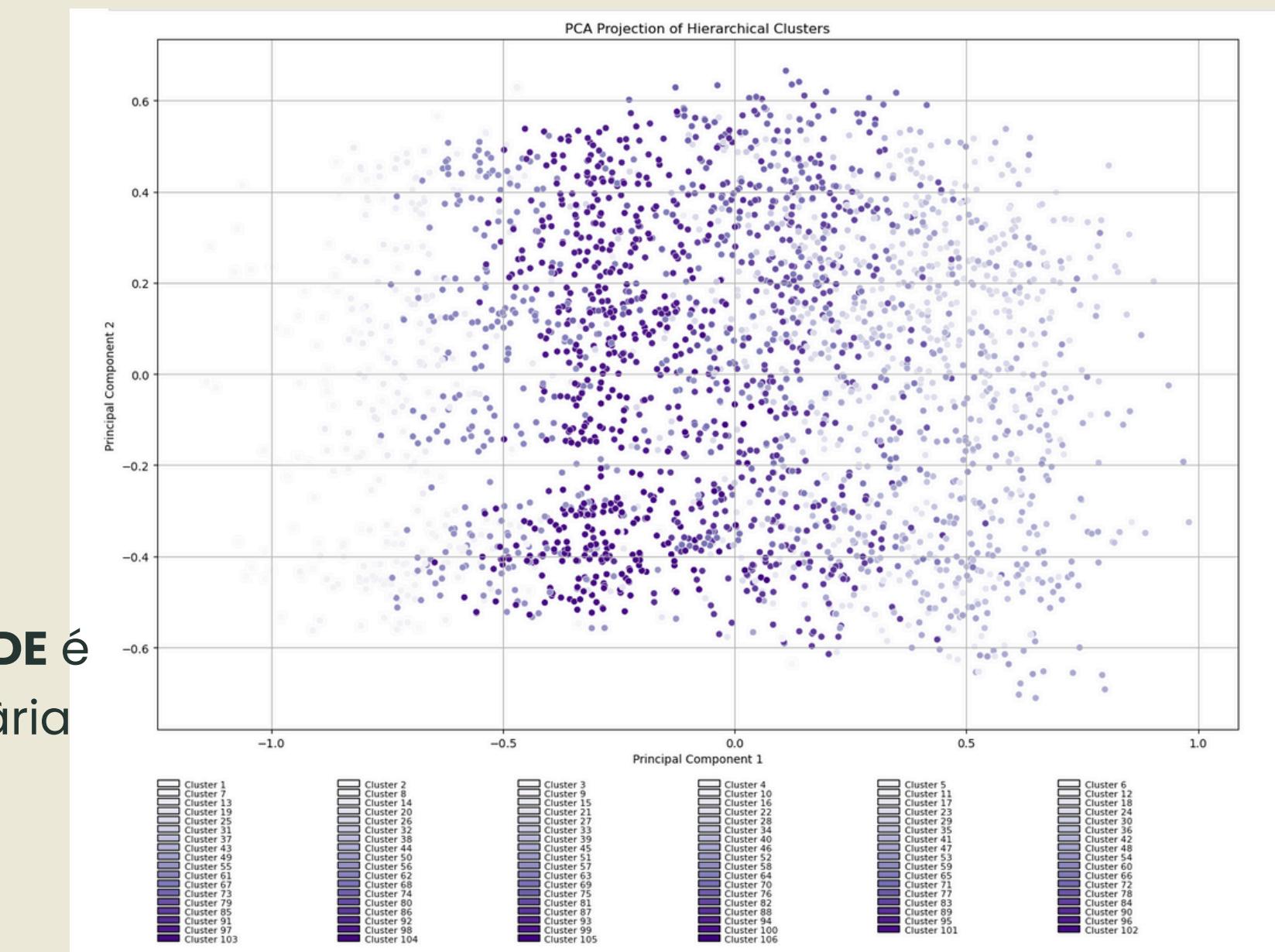
Grupos segregados em virtude de grande variância de uma variável binária.

VERDADEIRA

	PC1	PC2
mode	0.941287	-0.270838
acousticness	0.177116	0.503884
valence	0.092582	0.322209
danceability	0.046262	0.257952
liveness	0.018852	-0.006056
tempo	0.006292	-0.061811
duration_ms	-0.020770	-0.026353
popularity	-0.045738	0.048381
speechiness	-0.046865	-0.120539
loudness	-0.064208	-0.217542
explicit	-0.082406	-0.404737
instrumentalness	-0.122890	-0.285855
energy	-0.134029	-0.417181
key	-0.151006	0.118654

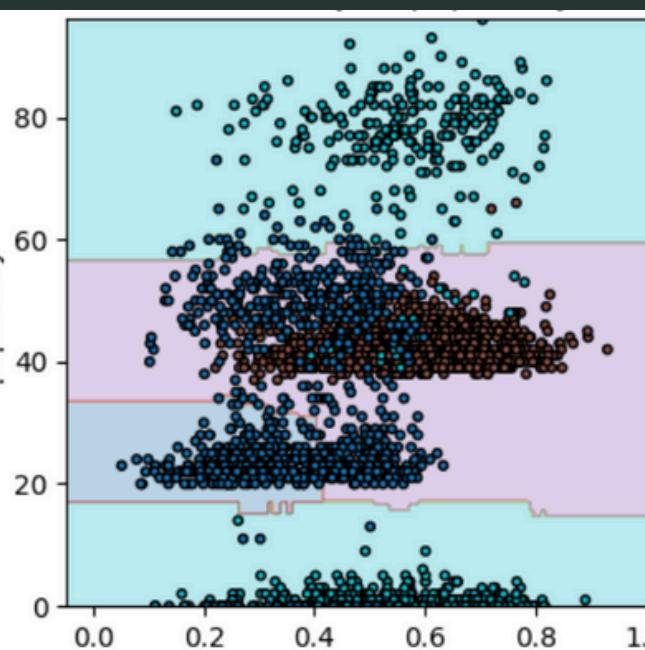
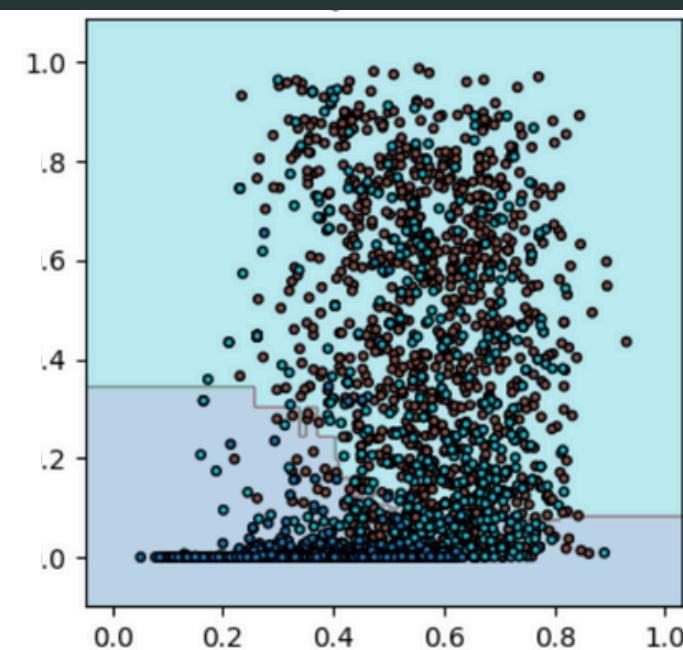
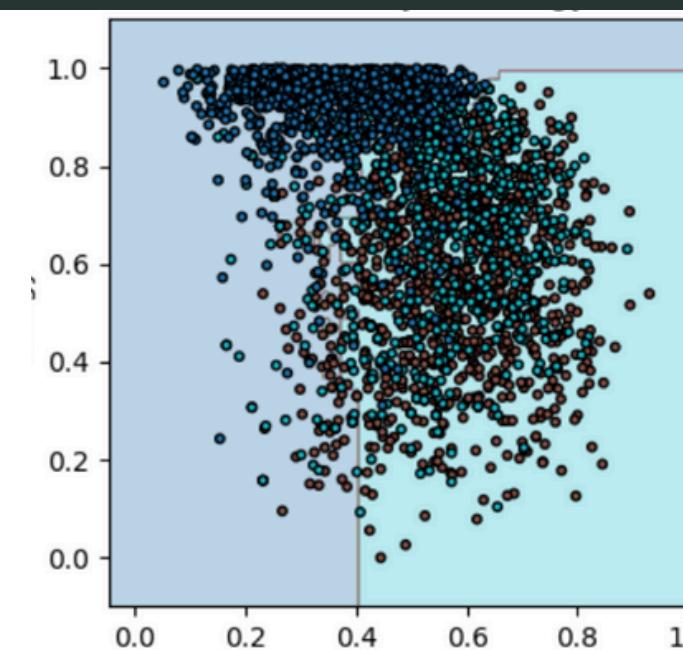
→ MODE é
Binária

AJUSTE: ELIMINAÇÃO DA FEATURE MODE:



Each cluster is represented by a unique shade of purple. The color mapping helps visually identify which samples belong to which clusters.

4. ANÁLISE DA CAPACIDADE DE CLASSIFICAÇÃO POR GÊNERO ATRAVÉS DA DEFINIÇÃO DE THRESHOLDS, AFIM DA VIABILIZAÇÃO DE AGRUPAMENTOS PRECISOS



CLASSIFICAÇÃO DE GÊNERO MUSICAL USANDO RANDOM FOREST

```
X = df.iloc[:, 6:20]
y = df["track_genre"]
feature_names = X.columns

scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X)
X = pd.DataFrame(X_scaled, columns=feature_names)

label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)
class_labels = label_encoder.classes_

X_train, X_test, y_train, y_test = train_test_split(
    X, y_encoded, test_size=0.25, random_state=42
)
```

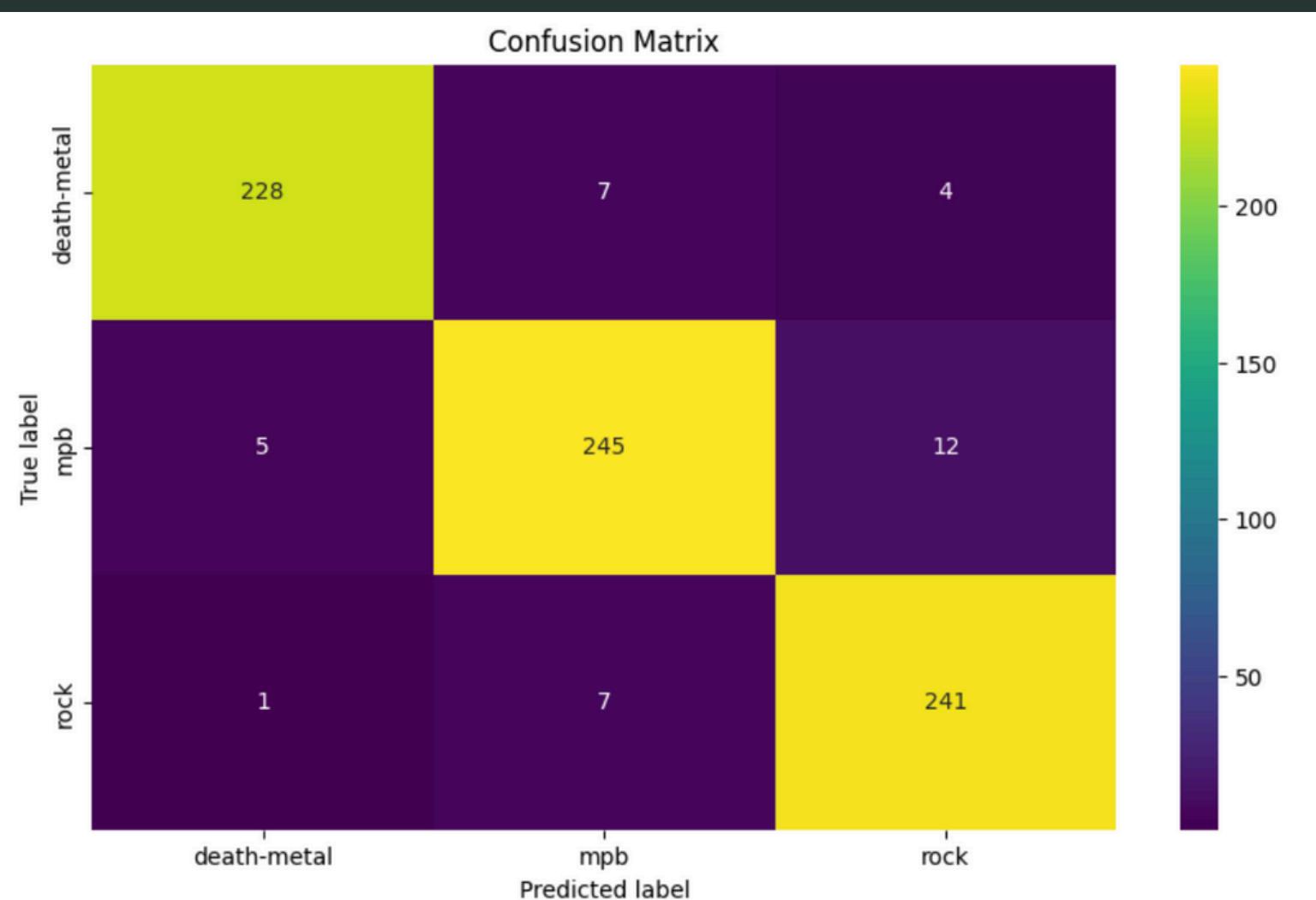
MÉTODO DE DIVISÃO DOS DADOS:

- **Técnica:** Holdout com divisão estratificada.
- **Proporção:** 75% treino (2.250 amostras) / 25% teste (750 amostras).
- **Configuração:** Random_state=42 para garantir reproduzibilidade.
- **Princípio:** Dados de teste mantidos isolados durante todo o treinamento.

ANÁLISE DOS RESULTADOS

PERFORMANCE POR GÊNERO MUSICAL

Gênero	Corretas	Total	Precisão	Erros Específicos
Death Metal	228	239	95.4%	7→MPB, 4→Rock
MPB	245	262	93.5%	12→Rock, 5→Death
Rock	241	249	96.8%	7→MPB, 1→Death



ANÁLISE DOS PADRÕES DE ERRO:

- **Death Metal – Mais Distintivo:**

- Apenas 11 erros totais de 239 instâncias
- Características musicais extremas facilitam separação

- **MPB ↔ Rock – Maior Confusão:**

- 19 casos de confusão mútua (12+7)
- Sobreposição em instrumentação e estrutura harmônica

ANÁLISE DA CAPACIDADE DE CLASSIFICAÇÃO POR GÊNERO

MÉTRICAS DE AVALIAÇÃO UTILIZADAS

ESTRATÉGIA DE VALIDAÇÃO:

- **Acurácia:**
 - Proporção de classificações corretas sobre o total
 - Métrica principal para avaliar performance geral
- **Log Loss:**
 - Avalia a qualidade das previsões probabilísticas
 - Penaliza previsões incorretas com alta confiança
- **Matriz de Confusão:**
 - Análise detalhada de erros por classe
 - Identifica padrões específicos de confusão entre gêneros

A BUSCA PELA MELHOR ÁRVORE:



Modelo	Acurácia	Log Loss
Random Forest (100 árvores)	97.33%	0.1418
Melhor Árvore Individual (#98)	95.20%	1.7301

- **Ensemble:** Random Forest supera árvore individual em +2.13% de acurácia, mas ela chega perto, apesar do nível de confiabilidade ser显著mente menor.

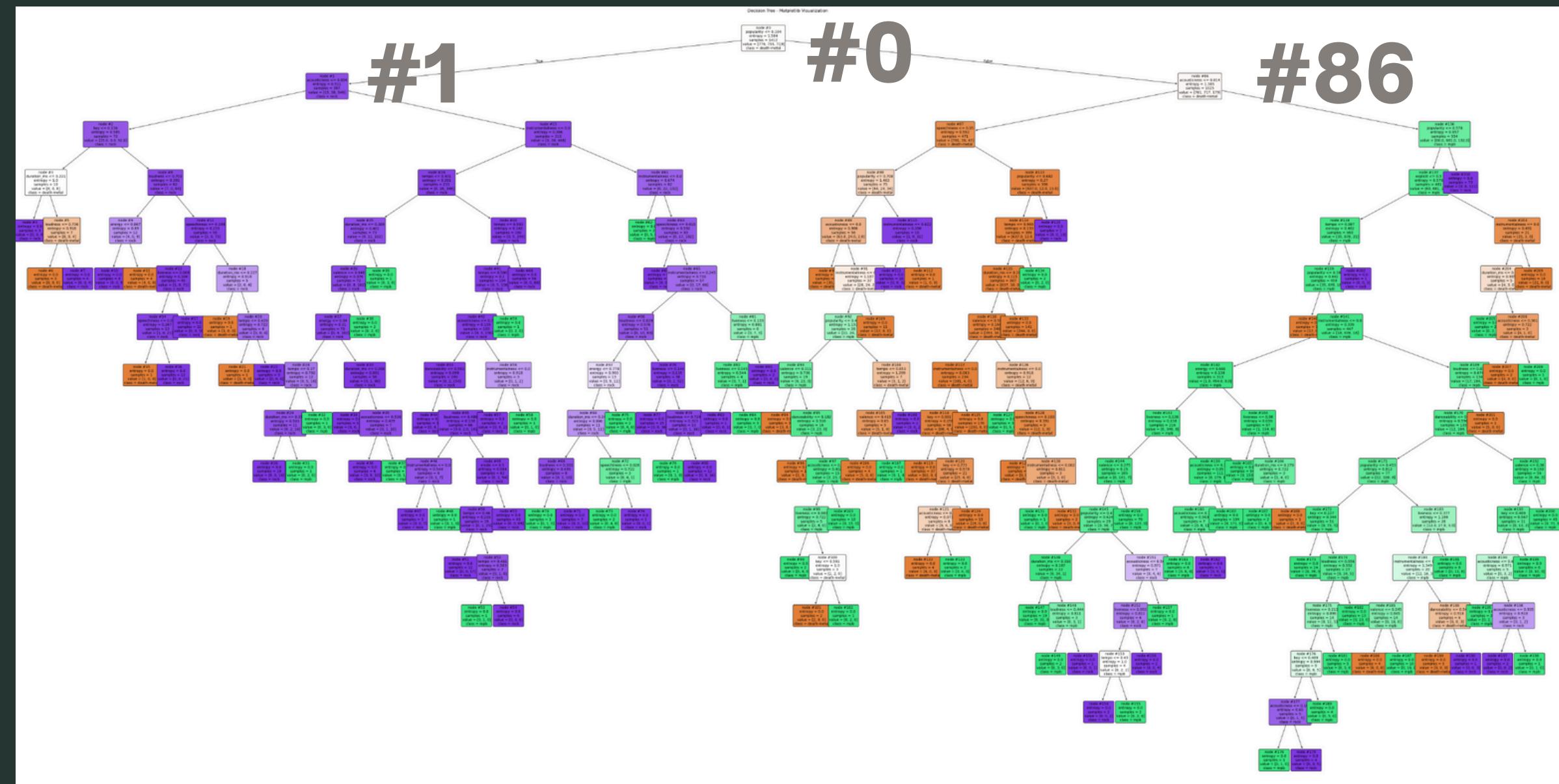
ANÁLISE DOS RESULTADOS

IMPORTÂNCIA DAS FEATURES

Rank	Feature	Importância	Interpretação Musical
1º	Popularity	46.7%	Padrões de consumo específicos por gênero
2º	Acousticness	31.2%	Separação acústico vs eletrônico
3º	Instrumentalness	4.1%	Distinção vocal vs instrumental

FEATURES MAIS DETERMINANTES

- **Popularity como fator principal:**
 - Reflete padrões de consumo diferenciados entre gêneros
 - Death Metal: nicho específico, MPB nicho brasileiro vs Rock: mainstream
- **Acousticness em segundo lugar:**
 - Alinha com expectativas teóricas musicais
 - Death Metal: baixa acousticness (eletrônico/produzido)
 - MPB: maior variabilidade acústica



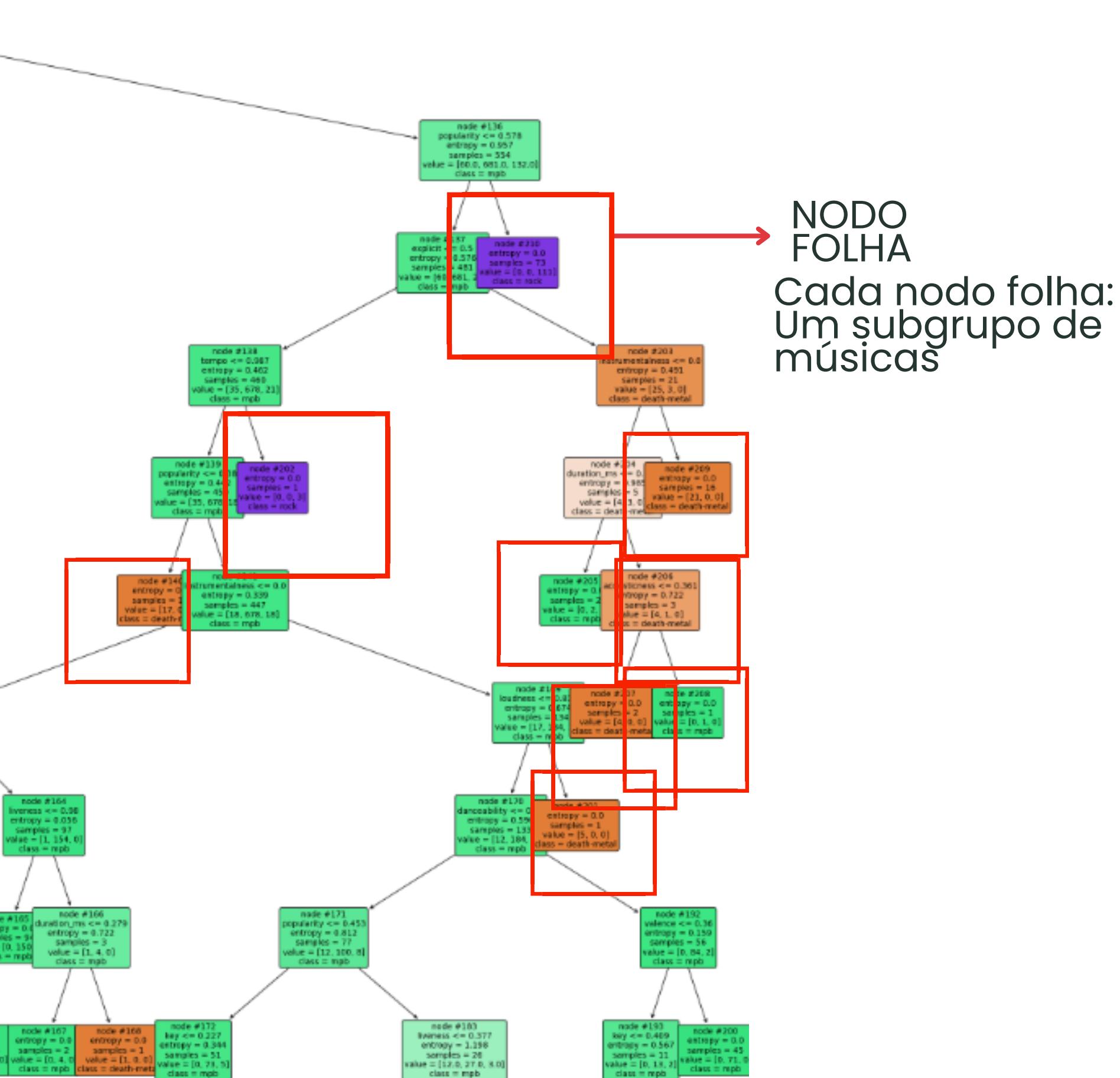
node #0
popularity <= 0.104
entropy = 1.584
samples = 1412
value = [776, 755, 719]
class = death-metal

node #1
acousticness <= 0.004
entropy = 0.511
samples = 387
value = [15, 38, 540]
class = rock

node #86
acousticness <= 0.014
entropy = 1.385
samples = 1025
value = [761, 717, 179]
class = death-metal

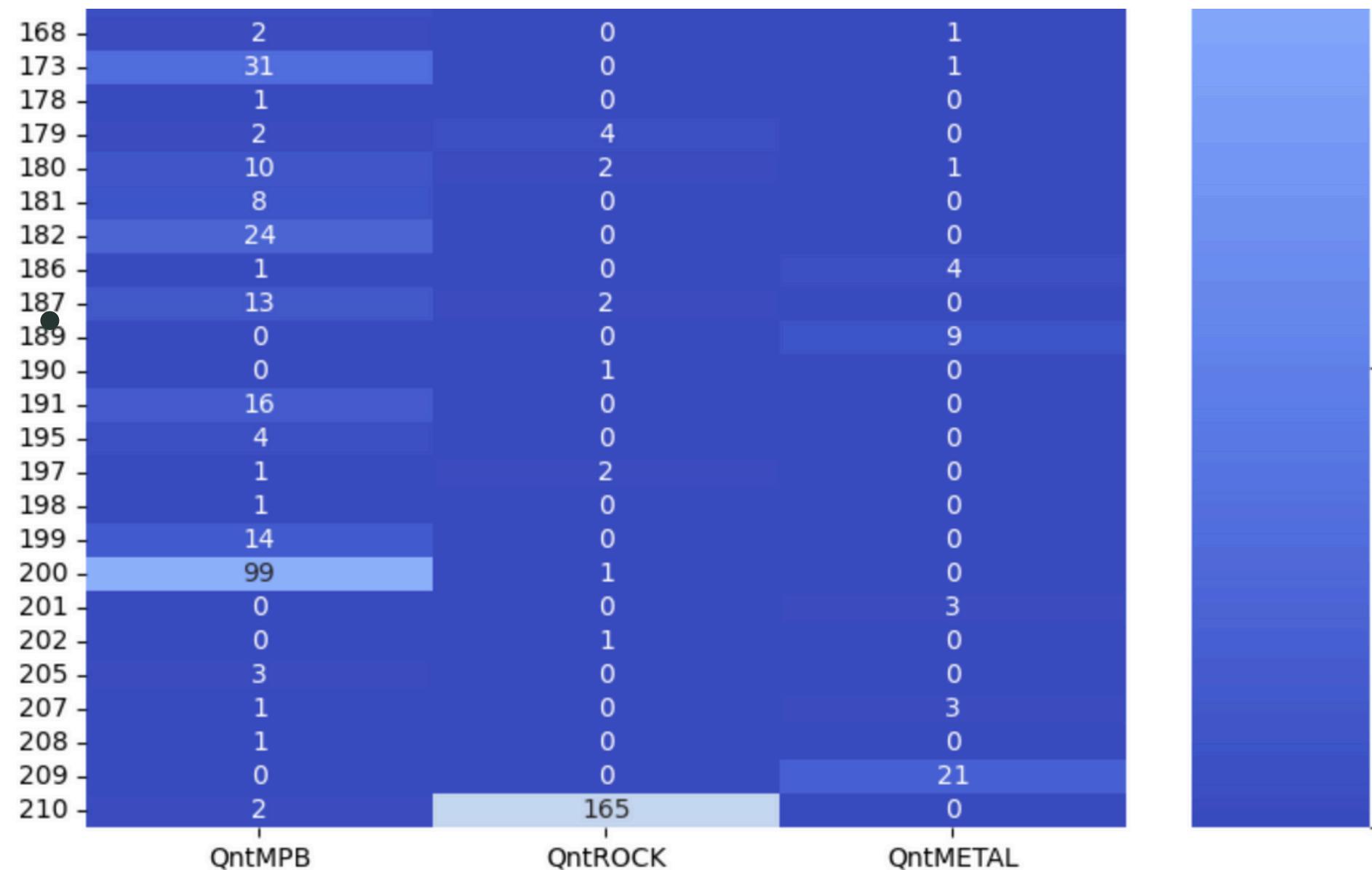
5. COMPARAÇÃO ENTRE ABORDAGENS CONSIDERANDO A PUREZA DOS AGRUPAMENTOS.

AGRUPAMENTO ATRAVÉS DO TREINAMENTO DA ÁRVORE DE DECISÃO



NODO FOLHA
Cada nodo folha:
Um subgrupo de
músicas

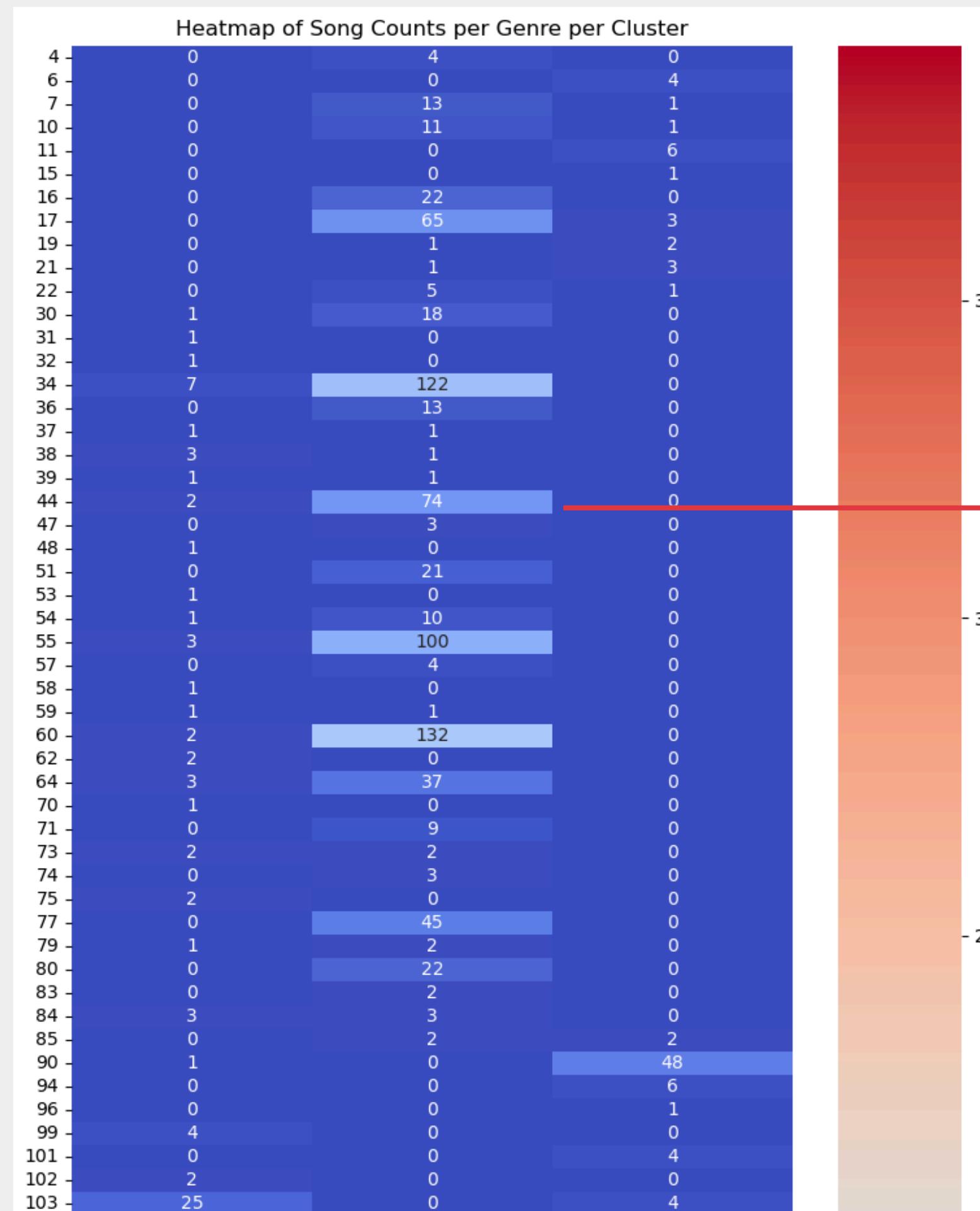
MAPA DE CALOR 106 NODOS FOLHA



Maior parte dos casos:
Cada nodo folha possui músicas
com **features similares e mesmo**
gênero agrupadas

COMPARAÇÃO ENTRE ABORDAGENS CONSIDERANDO A PUREZA DOS AGRUPAMENTOS.

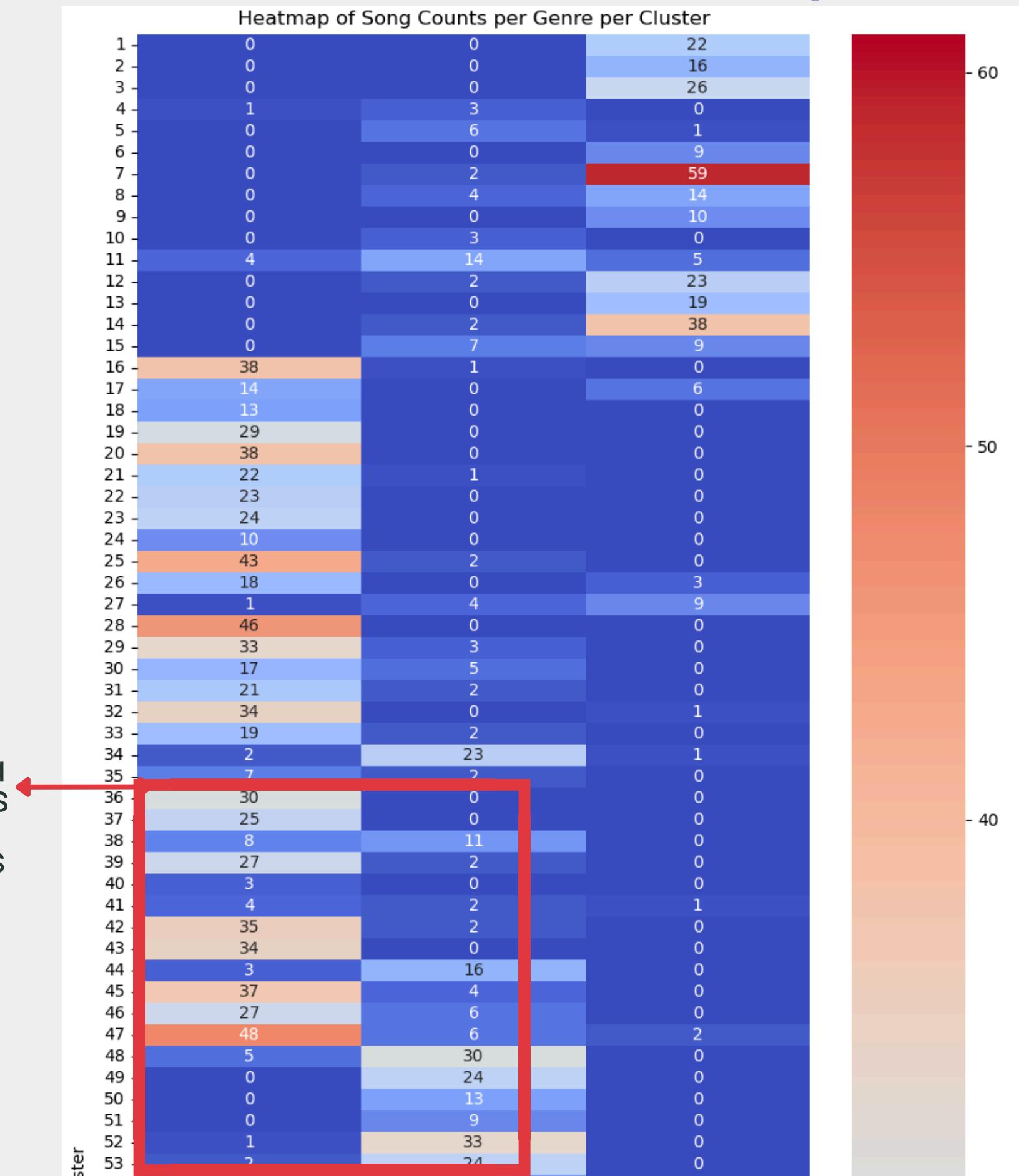
MAPA DE CALOR 106 NODOS FOLHA



Gêneros
melhor
segregados
por cluster

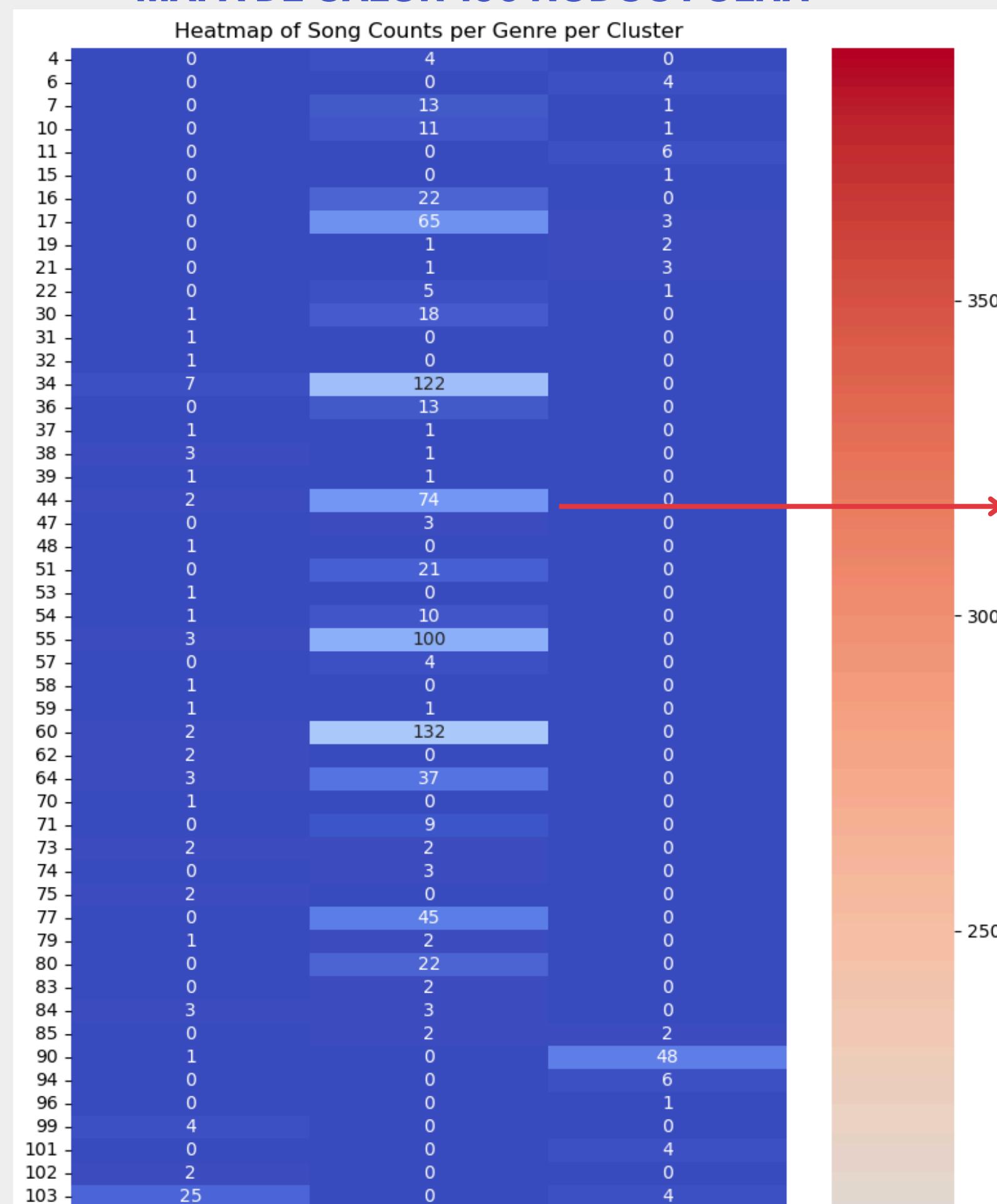
Ocorrência
de gêneros
misurados
agrupados
no mesmo
cluster

MAPA DE CALOR 106 CLUSTERS HIERARQUICOS



COMPARAÇÃO ENTRE ABORDAGENS CONSIDERANDO A PUREZA DOS AGRUPAMENTOS.

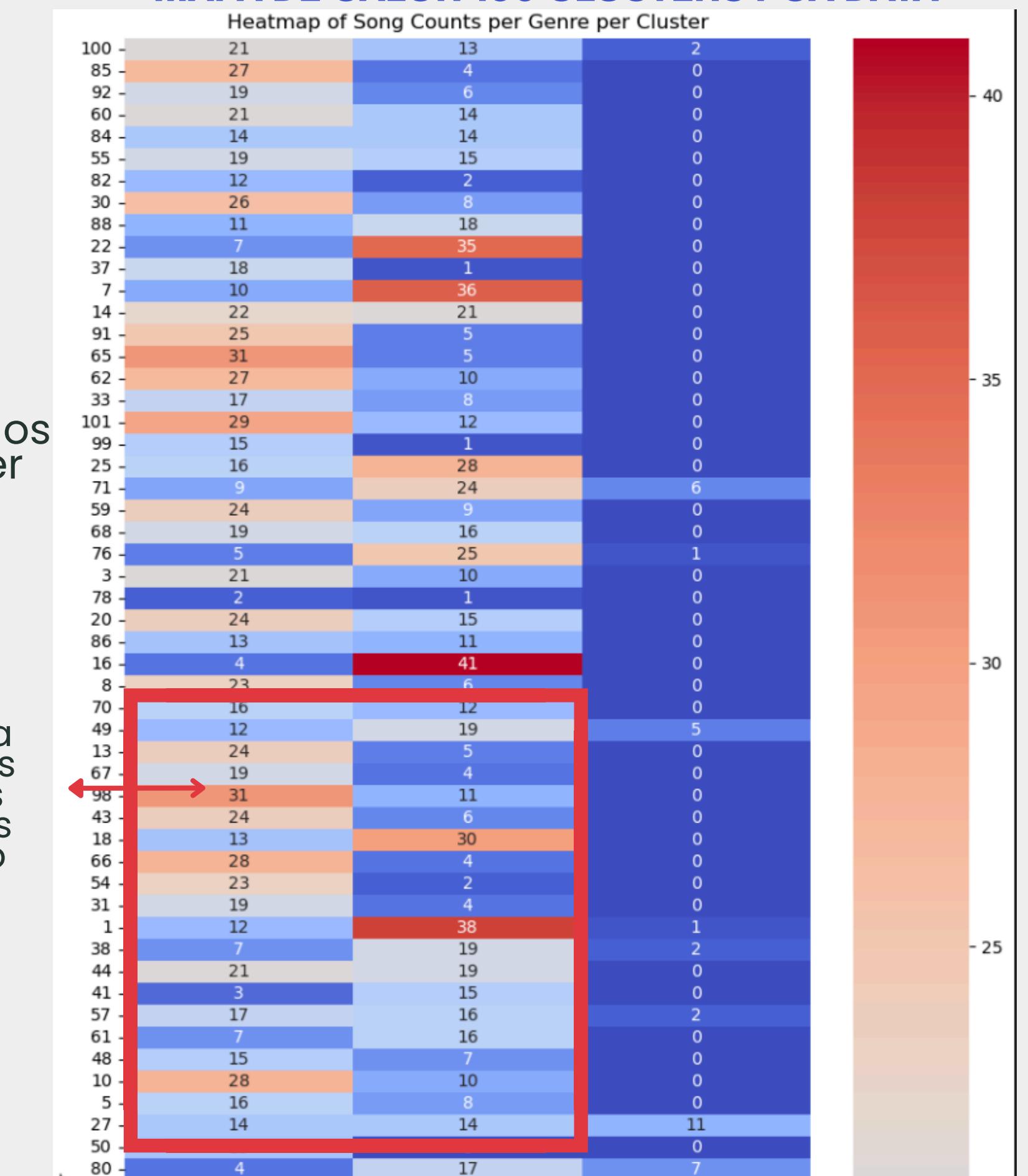
MAPA DE CALOR 106 NODOS FOLHA



Gêneros
melhor
segregados
por cluster

Ocorrência
de gêneros
misurados
agrupados
no mesmo
cluster

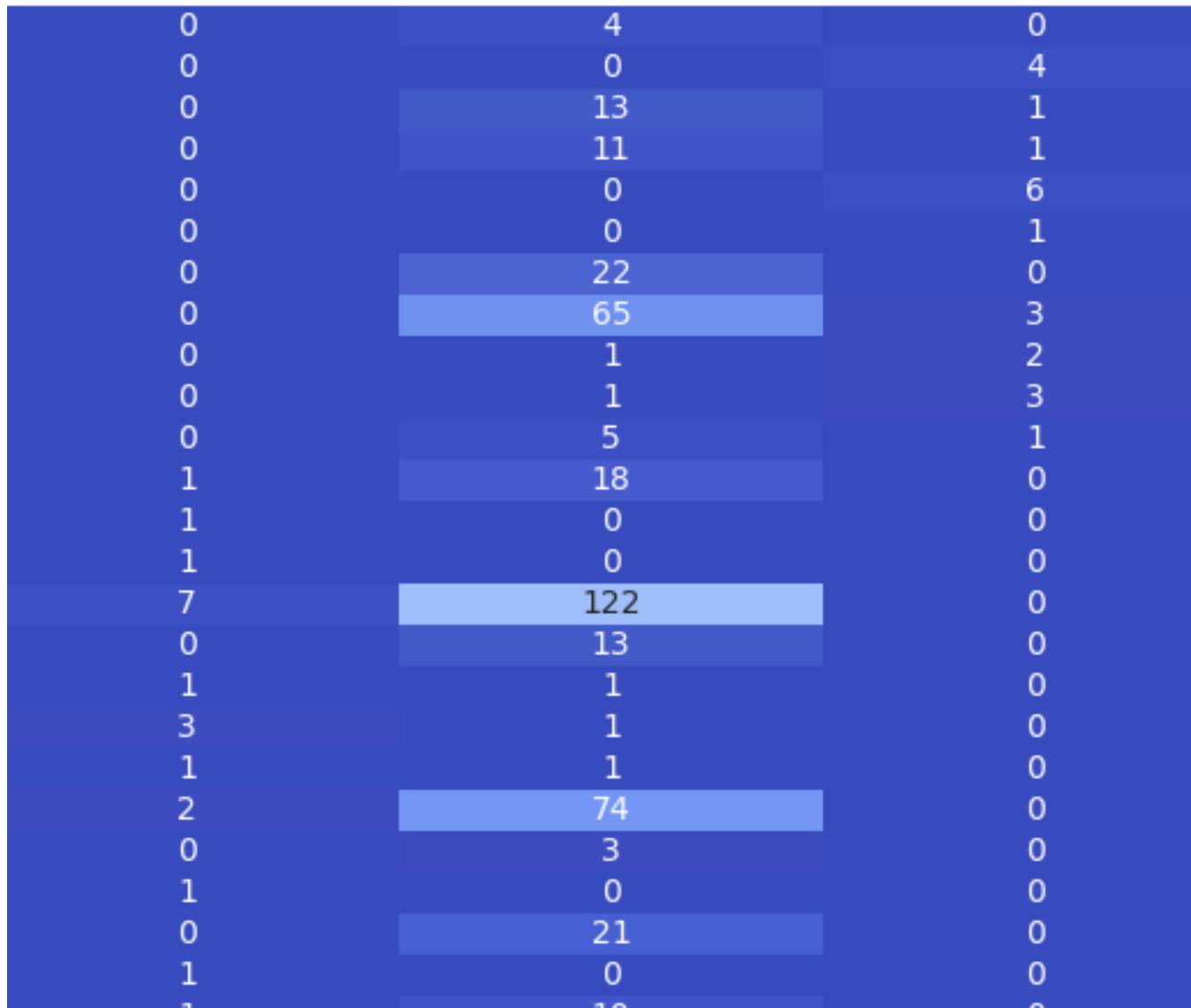
MAPA DE CALOR 106 CLUSTERS PCA DATA



COMPARAÇÃO ENTRE ABORDAGENS CONSIDERANDO A PUREZA DOS AGRUPAMENTOS.

MAPA DE CALOR 106 NODOS FOLHA

Heatmap of Song Counts per Genre per Cluster

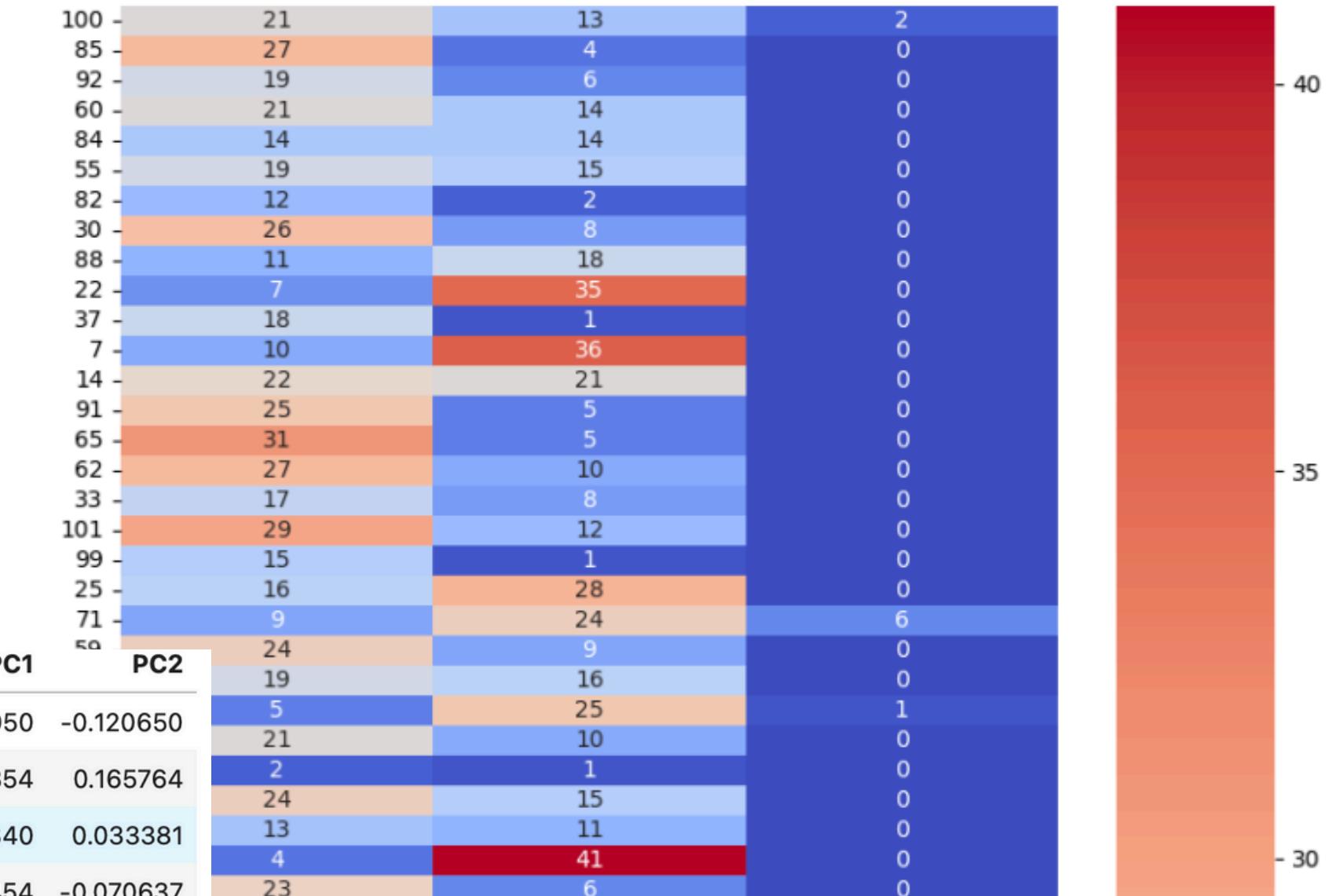


CONCLUSÃO 1:

MÉTODO DE
AGRUPAMENTO
BASE ESCOLHIDO

MAPA DE CALOR 106 CLUSTERS PCA DATA

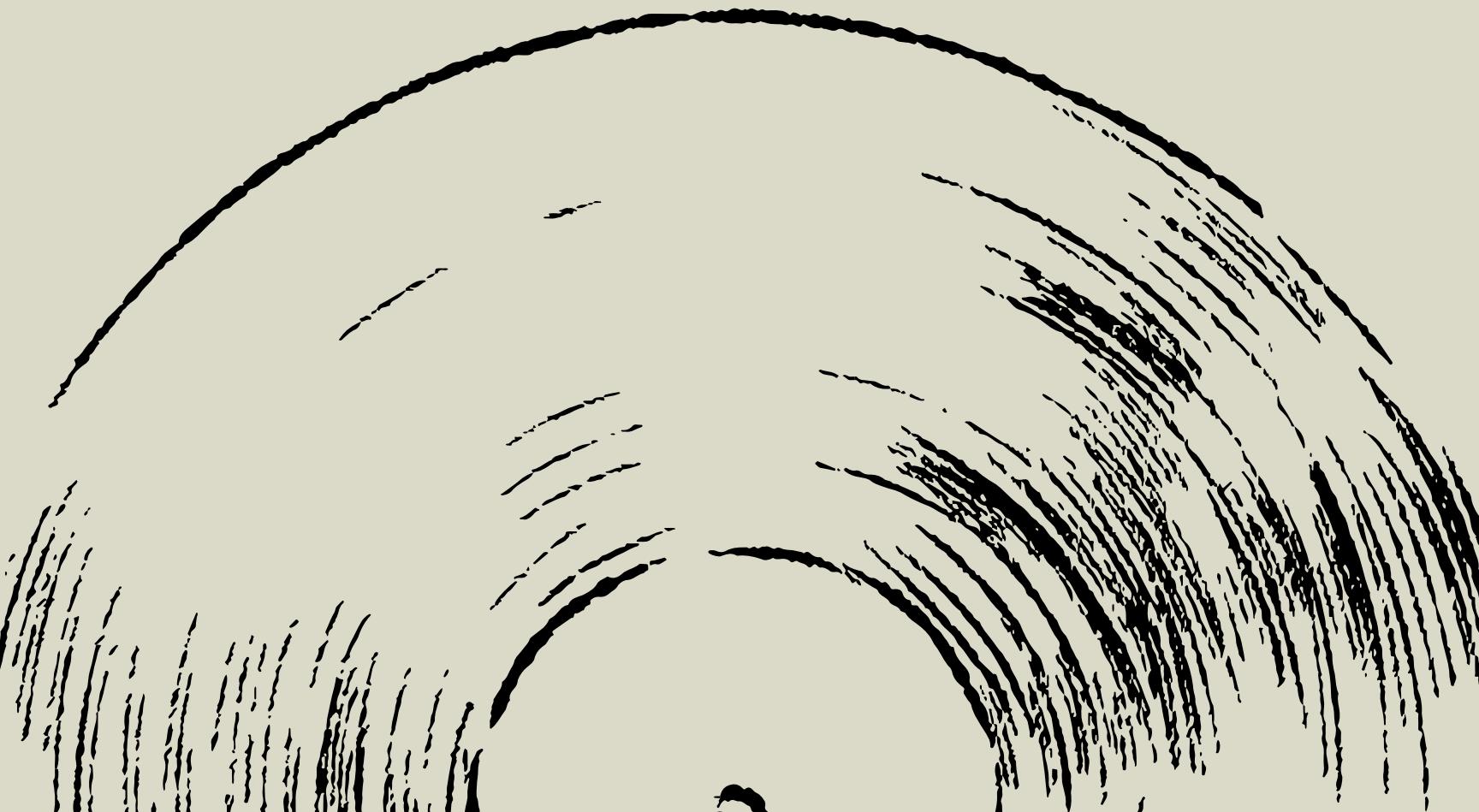
Heatmap of Song Counts per Genre per Cluster



CONCLUSÃO 2:

RELACÕES LINEARES
NÃO SUPERVISIONADAS
MOSTRAM OUTRAS RELAÇÕES
ENTRE OS DADOS

6. REALIZAÇÃO DE RECOMENDAÇÕES CONSIDERANDO GÊNEROS E PROXIMIDADES DE MÚSICAS DIMENSIONALMENTE



```

def pDecisionTree(Xdata, ydata):
    feature_names = Xdata.columns

    X_new_scaled = scaler.transform(Xdata)
    X_new = pd.DataFrame(X_new_scaled, columns=feature_names)
    y_true_encoded = label_encoder.transform(ydata)
    class_labels = label_encoder.classes_

    tree_pred = best_tree.predict(X_new)

    return X_new, tree_pred, y_true_encoded, class_labels

def addPCA(df, feature_start=5, feature_end=19, n_components=2):
    df = df.copy()
    X = df.iloc[:, feature_start:feature_end]
    X_scaled = scaler.fit_transform(X)

    pca = PCA(n_components=n_components)
    components = pca.fit_transform(X_scaled)

    df['PC1'] = components[:, 0]
    df['PC2'] = components[:, 1]

    return df

def getLeafBased2DInfo(df, pc_cols=['PC1', 'PC2'], leaf_col='leaf_node'):
    grouped = df.groupby(leaf_col)

    leaf_node_groups = {}
    for leaf, group in grouped:
        sorted_group = group.sort_values(by=pc_cols, ascending=[True, True])
        leaf_node_groups[leaf] = sorted_group

    return grouped, leaf_node_groups

```

PASSO 1: BEST_TREE FAZ AS PREDIÇÕES

PASSO 2: PCA É CALCULADO EM CIMA DE TODAS AS MÚSICAS

PASSO 3: VALORES PC1, PC2 SÃO ORDENADOS POR NODO FOLHA

```

def add_directional_recommendations(df, leaf_node_groups, id_col='index', k=5):
    df = df.copy()
    df['Recommendations'] = None

    for leaf, sorted_leaf_df in leaf_node_groups.items():
        sorted_leaf_df = sorted_leaf_df.reset_index(drop=True)
        leaf_len = len(sorted_leaf_df)

        for idx, row in sorted_leaf_df.iterrows():
            song_id = row[id_col]

            if idx < leaf_len / 2:
                recs = sorted_leaf_df.iloc[idx + 1:][id_col].tolist()
            else:
                recs = sorted_leaf_df.iloc[:idx][id_col].tolist()

            if k is not None:
                recs = recs[:k]

            df.at[df[df[id_col] == song_id].index[0], 'Recommendations'] = recs

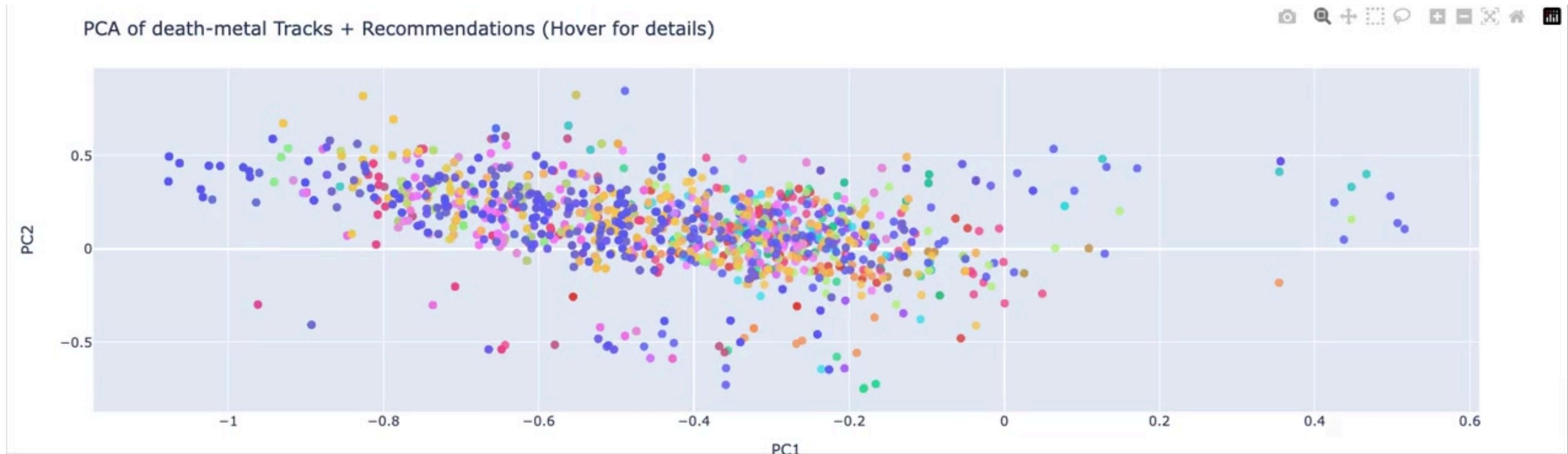
    return df

```

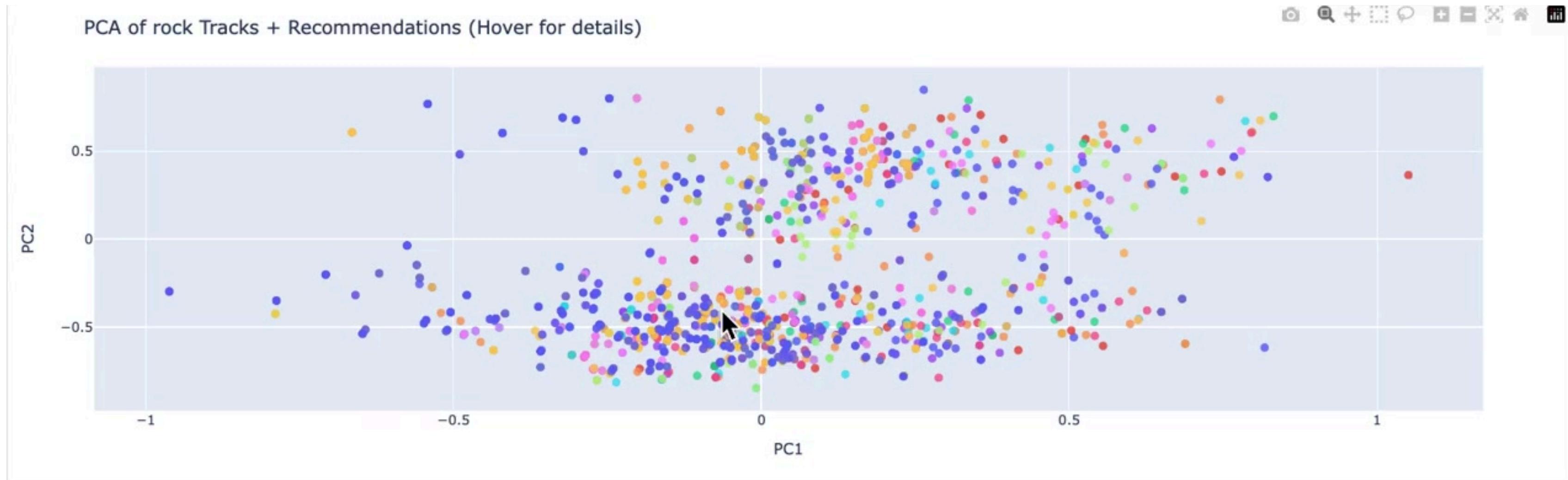
Cada track recebe entre 5 e uma 1 músicas recomendadas, que são escolhidas entre o mesmo nodo, com os valores de PC1 e PC2 mais próximos.

PASSO 4: RECOMENDAÇÕES DIRECIONAIS SÃO FORNECIDAS

DEATH METAL: ORIGINAIS E RECOMENDAÇÕES



ROCK: ORIGINAIS E RECOMENDAÇÕES



MPB: ORIGINAIS E RECOMENDAÇÕES



REPOSITÓRIO GITHUB:
**[https://github.com/carolinapedoneb/T2-
Introducao-Ciencia-de-Dados/tree/main](https://github.com/carolinapedoneb/T2-Introducao-Ciencia-de-Dados/tree/main)**