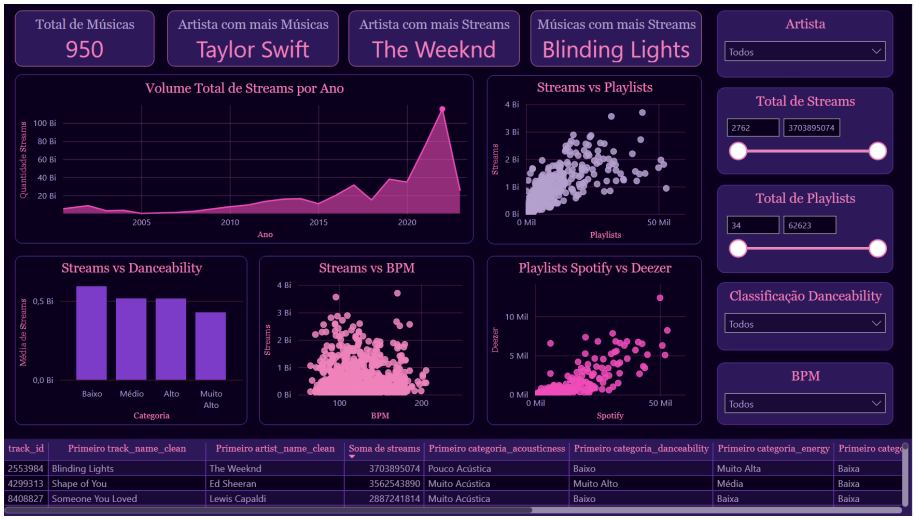
Documentação

Análise de Dados - Plataformas de Streaming - Hipóteses:

1. Equipe e Documentação:

Carla Bruckmann e Cristy Ellen Ribeiro.

- **Documentação**
- **Repositório - GitHub**
- **Repositório - BigQuery**
- **Apresentação**
- **Vídeo**
- ▼ **Dashboard**



2. Objetivo da Análise:

Avaliar os fatores que influenciam o sucesso de músicas no Spotify, com base em dados reais de músicas**,** a fim de **validar hipóteses estratégicas** e fornecer **recomendações para o lançamento bem-sucedido de um novo artista** pela gravadora. O foco está em identificar padrões, correlações e características que estejam associadas a um maior número de streams.

- ▼ ❓ **Hipóteses Norteadoras**
  - **Músicas com BPM (Batidas Por Minuto) mais altos tendem a ter mais streams no Spotify.**
  - **As músicas mais populares no ranking do Spotify também apresentam alto desempenho em outras plataformas, como Deezer.**
  - **Existe uma correlação positiva entre o número de playlists em que uma música aparece e seu número total de streams.**
  - **Artistas com um maior número de músicas disponíveis no Spotify tendem a ter um volume maior de streams totais.**

- **Características específicas das músicas (como energia, dançabilidade, valência, acústica, etc.) influenciam significativamente o número de streams no Spotify.**

3. Ferramentas e Tecnologias:

- Notion (Documentação e Gerenciamento de Tarefas).
- Google Drive.
- BigQuery.
- PowerBi.
- GitHub (Repositório).

4. Fonte dos Dados:

Origem dos dados:

Database

▼ **Definição de variáveis:**

Track*in*spotify

- **track_id**: Identificador exclusivo da música. É um número inteiro de 7 dígitos que não se repete.
- **track_name**: Nome da música.
- *artist(s)_name**: Nome do(s) artista(s) da música.
- **artist_count**: Número de artistas que contribuíram na música.
- **released_year**: Ano em que a música foi lançada.
- **released_month**: Mês em que a música foi lançada.
- **released_day**: Dia do mês em que a música foi lançada.
- **in*spotify*playlists**: Número de listas de reprodução do Spotify em que a música está incluída
- **in*spotify*charts**: Presença e posição da música nas paradas do Spotify
- **streams**: Número total de streams no Spotify. Representa o número de vezes que a música foi ouvida.

Track*in*competition

- **track_id**: Identificador exclusivo da música. É um número inteiro de 7 dígitos que não se repete.
- **in*apple*playlists**: número de listas de reprodução da Apple Music em que a música está incluída.
- **in*apple*charts**: Presença e classificação da música nas paradas da Apple Music.
- **in*deezer*playlists**: Número de playlists do Deezer em que a música está incluída.
- **in*deezer*charts**: Presença e posição da música nas paradas da Deezer.
- **in*shazam*charts**: Presença e classificação da música nas paradas da Shazam.

Track*technical*info

- **track_id**: Identificador exclusivo da música. É um número inteiro de 7 dígitos que não se repete.
- **bpm**: Batidas por minuto, uma medida do tempo da música.
- **key**: Tom musical da música.
- **mode**: Modo de música (maior ou menor).
- **danceability_%**: Porcentagem que indica o quão apropriado a canção é para dançar
- **valence_%**: Positividade do conteúdo musical da música.

- **energy_%**: Nível de energia percebido da música.
- **acusticness_%**: Quantidade de som acústico na música.
- **instrumentality_%**: Quantidade de conteúdo instrumental na música.
- **liveness_%**: Presença de elementos de performance ao vivo.
- **speechiness_%**: Quantidade de palavras faladas na música.

5. Pré-processamento:

▼ **Importação de dados:**

- Base de dados salvas em Google Drive - formato CSV.
- No BigQuery:
  - Criação de `Projeto 2 - Laboratoria` (ID: `projeto-2-laboratoria-456917`);
  - Criação de Conjunto de Dados `database_projeto_2`;
  - Criação de tabelas `track_in_spotify`, `track_in_competition`, `track_technical_info` em `database_projeto_2`.

▼ **Limpeza dos Dados:**

1. **Tabela** `track_in_spotify`**:**
   a. Contagem e identificação de Dados Ausentes:

▼ **Query identificação:**

```
SELECT
  COUNT(*)
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify`
WHERE track_id IS NULL
  OR track_name IS NULL
  OR artist_s__name IS NULL
  OR artist_count IS NULL
  OR released_year IS NULL
  OR released_day IS NULL
  OR released_month IS NULL
  OR in_spotify_playlists IS NULL
  OR in_spotify_charts IS NULL
  OR streams IS NULL;

  SELECT *
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify`
WHERE track_id IS NULL
 OR track_name IS NULL
  OR artist_s__name IS NULL
  OR artist_count IS NULL
  OR released_year IS NULL
  OR released_day IS NULL
  OR released_month IS NULL
  OR in_spotify_playlists IS NULL
  OR in_spotify_charts IS NULL
  OR streams IS NULL;
```

- Utilizando `COUNT WHERE ISNULL`, não foram encontrados dados nulos nesta tabela.

   b. Verificação e Remoção de duplicatas:

▼ **Query identificação:**

```
-- Duplicatas por track_id
SELECT track_id, COUNT(*) AS quantidade
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify`
GROUP BY track_id
HAVING COUNT(*) > 1;

-- Duplicatas por track_name + artist_s__name
SELECT track_name, artist_s__name, COUNT(*) AS quantidade
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify`
GROUP BY track_name, artist_s__name
HAVING COUNT(*) > 1;

-- Visualizar registros duplicados completos
SELECT *
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify`
WHERE CONCAT(track_name, artist_s__name) IN (
  SELECT CONCAT(track_name, artist_s__name)
  FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify`
  GROUP BY track_name, artist_s__name
  HAVING COUNT(*) > 1
)
ORDER BY track_name, artist_s__name;
```

- Durante a etapa de validação de dados, foram identificadas **4 músicas com duplicidade de nome e artista**. A seguir, foi realizada uma verificação cruzada entre as tabelas `track_in_spotify` e `track_technical_info` para entender se essas duplicações representavam registros idênticos ou versões distintas.

  ▼ **Query comparativa:**

```
WITH ids_desejados AS (
  SELECT "5080031" AS track_id, 1 AS ordem UNION ALL
  SELECT "7173596", 2 UNION ALL
  SELECT "5675634", 3 UNION ALL
  SELECT "3814670", 4 UNION ALL
  SELECT "4967469", 5 UNION ALL
  SELECT "8173823", 6 UNION ALL
  SELECT "1119309", 7 UNION ALL
  SELECT "4586215", 8
)

SELECT
  a.track_id,
  a.track_name,
  a.artist_s__name,
  a.released_day,
  a.released_month,
  a.streams,
  b.bpm,
  b.key,
  b.danceability__,
  b.valence__,
  b.energy__,
```

```
  b.acousticness__,
  b.instrumentalness__,
  b.liveness__,
  b.speechiness__
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify` a
JOIN `projeto-2-laboratoria-456917.database_projeto_2.track_technical_info` b
  ON a.track_id = b.track_id
JOIN ids_desejados i
  ON a.track_id = i.track_id
ORDER BY i.ordem;
```

Tabela track_in_spotify						tabela track_technical_info									
track_id	track_name	artist_s__name	released_month	released_day	streams	bpm	key	mode	danceability__	valence__	energy__	acousticness__	instrumentalness	liveness__	speechiness__
5080031	About Damn Time	Lizzo	7	15	726307468	109	A#	Minor	84	72	74	10	0	34	7
7173596	About Damn Time	Lizzo	4	14	711366595	109	A#	Minor	84	72	74	10	0	34	7
5675634	SNAP	Rosa Linn	3	19	723894473	170	null	Major	56	53	64	11	0	45	8
3814670	SNAP	Rosa Linn	3	19	723894473	170	null	Major	56	53	64	11	0	45	7
4967469	SPIT IN MY FACE!	ThxSoMch	10	31	130655803	94	G#	Major	73	65	79	5	2	11	6
8173823	SPIT IN MY FACE!	ThxSoMch	10	31	432702334	166	C#	Major	70	57	57	9	20	11	7
1119309	Take My Breath	The Weeknd	8	6	303216294	121	A#	Minor	70	35	77	1	0	26	4
4586215	Take My Breath	The Weeknd	8	6	301869854	121	G#	Major	75	53	74	2	0	11	5

- **1 faixa apresentou apenas diferenças na data de lançamento** (dia e mês), mantendo características técnicas idênticas (`track_id:` 5080031 e 7173596).

  Decisão: Será excluída uma duplicata da base de dados, mantendo-se a que possuí maior número de streams:

  ID mantida: 5080031 (streams: 726307468).

  ID excluída: 7173596 (streams: 711366595).

- **3 faixas possuem variações técnicas significativas** (como `bpm` , `energy__` , `valence__`), indicando que são **versões diferentes da mesma música** (ex: remix, ao vivo, acústica).

  Decisão: Serão mantidas na base por representarem versões distintas relevantes para a análise.

c. Verificação e gerenciamento de dados fora do escopo ou atípicos:

- Durante o processo de pré-análise, um dado foi identificado como **irrelevante para a validação das hipóteses definidas** e, por isso, **foi excluído da análise principal:**
  - `artist_count` : Número de artistas que contribuíram na faixa. Considerou-se que essa informação não impactaria diretamente nas análises de desempenho ou características técnicas das músicas.

  📌 Importante: Esse dado **não foi descartado permanentemente**. Permanece disponível no banco de dados e poderá ser utilizado em análises futuras, especialmente se novas hipóteses envolverem colaborações entre artistas.

d. Analise de variáveis numéricas e tipagem:

- Durante o tratamento da variável `streams` , foram identificados valores não numéricos.

  Decisão: Serão retiradas, para considerarmos apenas valores válidos contendo apenas dígitos numéricos.

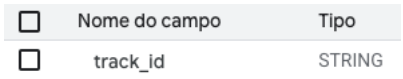
Linha	streams
1	BPM110KeyAModeMajorDanceability53Valence75Energy69Acousticness7Instrumentalness0Liveness17S

▼ **Query verificação:**

```
--identifica valor não totalmente numéricos
SELECT DISTINCT streams , track_id, track_name
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify`
WHERE NOT REGEXP_CONTAINS(streams, r'^\d+$');
```

- Durante tratamento de variável `track_id` , ao efetuar correção de tipagem, de `STRING` para `INTERGER` , conforme descrição de variável (número inteiro de 7 dígitos que não se repete), foram identificados valores **fora do padrão de id, impossibilitando a transformação de tipagem**.

  Decisão: Serão retiradas, para considerarmos apenas valores válidos contendo apenas dígitos numéricos.



Linha	track_id	track_name
1	0:00	10:35

2. **Tabela** `track_in_competition` **:**

   a. Contagem e identificação de Dados Ausentes:

   ▼ **Query identificação:**

```
SELECT
  COUNT(*)
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_competition`
WHERE track_id IS NULL
  OR in_apple_playlists IS NULL
  OR in_apple_charts IS NULL
  OR in_deezer_playlists IS NULL
  OR in_deezer_charts IS NULL
  OR in_shazam_charts IS NULL;

  SELECT *
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_competition`
WHERE track_id IS NULL
  OR in_apple_playlists IS NULL
  OR in_apple_charts IS NULL
  OR in_deezer_playlists IS NULL
  OR in_deezer_charts IS NULL
  OR in_shazam_charts IS NULL;
```

   - Utilizando `COUNT WHERE ISNULL` , o resultado indicou **50 registros com dados nulos** em coluna `in_shazam_charts` .

     Decisão: A variável `in_shazam_charts` foi **excluída da análise**, uma vez que **não apresenta dados suficientemente robustos** em comparação com outras métricas de desempenho disponibilizadas pelas demais plataformas (Spotify, Deezer e Apple Music).

   b. Verificação e Remoção de duplicatas:

   ▼ **Query identificação:**

```
SELECT
  track_id,
  in_apple_playlists,
  in_apple_charts,
  in_deezer_playlists,
  in_deezer_charts,
  in_shazam_charts,
  COUNT(*) AS total
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_competition`
```

```
GROUP BY
  track_id,
  in_apple_playlists,
  in_apple_charts,
  in_deezer_playlists,
  in_deezer_charts,
  in_shazam_charts
HAVING COUNT(*) > 1;
```

- Não foram encontrados dados duplicados nesta tabela.

c. Verificação e gerenciamento de dados fora do escopo ou atípicos:

- A variável `in_shazam_charts` foi **excluída da análise** após identificação de um número significativo de registros nulos. Além disso, observou-se que a variável **não apresenta a mesma consistência e cobertura de dados** quando comparada às métricas de desempenho de outras plataformas, o que compromete sua relevância para os objetivos da análise atual.

d. Analise de variáveis numéricas e tipagem:

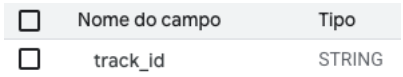
- Verificação de `MIN`, `MAX` e `AVG`, não foram encontrados dados discrepantes.

  ▼ **Query verificação:**

```
SELECT
  -- Apple Music
  MAX(CAST(in_apple_playlists AS INT64)) AS max_apple_playlists,
  MIN(CAST(in_apple_playlists AS INT64)) AS min_apple_playlists,
  AVG(CAST(in_apple_playlists AS INT64)) AS avg_apple_playlists,

  MAX(CAST(in_apple_charts AS INT64)) AS max_apple_charts,
  MIN(CAST(in_apple_charts AS INT64)) AS min_apple_charts,
  AVG(CAST(in_apple_charts AS INT64)) AS avg_apple_charts,

  -- Deezer
  MAX(CAST(in_deezer_playlists AS INT64)) AS max_deezer_playlists,
  MIN(CAST(in_deezer_playlists AS INT64)) AS min_deezer_playlists,
  AVG(CAST(in_deezer_playlists AS INT64)) AS avg_deezer_playlists,

  MAX(CAST(in_deezer_charts AS INT64)) AS max_deezer_charts,
  MIN(CAST(in_deezer_charts AS INT64)) AS min_deezer_charts,
  AVG(CAST(in_deezer_charts AS INT64)) AS avg_deezer_charts,


FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_competition`;
```

- Durante tratamento de variável `track_id`, ao efetuar correção de tipagem, de `STRING` para `INTERGER`, conforme descrição de variável (número inteiro de 7 dígitos que não se repete), foram identificados valores **fora do padrão de id, impossibilitando a transformação de tipagem**.

  Decisão: Serão retiradas, para considerarmos apenas valores válidos contendo apenas dígitos numéricos.



Linha	track_id

1	0:00

3. **Tabela** `track_technical_info`**:**
    a. Contagem e identificação de Dados Ausentes:
        ▼ **Query identificação:**

```
SELECT
  COUNT(*)
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_technical_info`
WHERE track_id IS NULL
  OR bpm IS NULL
  OR key IS NULL
  OR mode IS NULL
  OR danceability__ IS NULL
  OR valence__ IS NULL
  OR energy__ IS NULL
  OR acousticness__ IS NULL
  OR instrumentalness__ IS NULL
  OR liveness__ IS NULL
  OR speechiness__ IS NULL;

  SELECT *
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_technical_info`
WHERE track_id IS NULL
 OR bpm IS NULL
  OR key IS NULL
  OR mode IS NULL
  OR danceability__ IS NULL
  OR valence__ IS NULL
  OR energy__ IS NULL
  OR acousticness__ IS NULL
  OR instrumentalness__ IS NULL
  OR liveness__ IS NULL
  OR speechiness__ IS NULL;

  SELECT
  COUNTIF(track_id IS NULL) AS null_track_id,
  COUNTIF(bpm IS NULL) AS null_bpm,
  COUNTIF(key IS NULL) AS null_key,
  COUNTIF(mode IS NULL) AS null_mode,
  COUNTIF(danceability__ IS NULL) AS null_danceability,
  COUNTIF(valence__ IS NULL) AS null_valence,
  COUNTIF(energy__ IS NULL) AS null_energy,
  COUNTIF(acousticness__ IS NULL) AS null_acousticness,
  COUNTIF(instrumentalness__ IS NULL) AS null_instrumentalness,
  COUNTIF(liveness__ IS NULL) AS null_liveness,
  COUNTIF(speechiness__ IS NULL) AS null_speechiness
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_technical_info`;
```

- Utilizando `COUNT WHERE ISNULL COUNTIF` , foram encontradas 95 linhas com dados nulos em coluna `key` , responsável por indicar a tonalidade da faixa.

  Decisão: Os valores nulos foram **substituídos por "desconhecido"**, com o objetivo de preservar a integridade do conjunto de dados e evitar perdas durante o processo analítico.

b. Verificação e Remoção de duplicatas:

▼ **Query identificação:**

```
SELECT
  track_id,
  COUNT(*) AS total
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_technical_info`
GROUP BY track_id
HAVING COUNT(*) > 1;

SELECT
  track_id,
  bpm,
  key,
  mode,
  danceability__,
  valence__,
  energy__,
  acousticness__,
  instrumentalness__,
  liveness__,
  speechiness__,
  COUNT(*) AS total
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_technical_info`
GROUP BY
  track_id,
  bpm,
  key,
  mode,
  danceability__,
  valence__,
  energy__,
  acousticness__,
  instrumentalness__,
  liveness__,
  speechiness__
HAVING COUNT(*) > 1;
```

- Não foram encontrados dados duplicados nesta tabela.

c. Verificação e gerenciamento de dados fora do escopo ou atípicos:

- Não foram encontrados dados fora do escopo nesta tabela.

d. Analise de variáveis numéricas e tipagem:

- Verificação de `MIN,` `MAX` e `AVG`, não foram encontrados dados discrepantes.

▼ **Query verificação:**

```
SELECT
  -- bpm
  MAX(bpm) AS max_bpm,
  MIN(bpm) AS min_bpm,
  AVG(bpm) AS avg_bpm,
```

```
-- danceability
MAX(danceability__) AS max_danceability,
MIN(danceability__) AS min_danceability,
AVG(danceability__) AS avg_danceability,

-- valence
MAX(valence__) AS max_valence,
MIN(valence__) AS min_valence,
AVG(valence__) AS avg_valence,

-- energy
MAX(energy__) AS max_energy,
MIN(energy__) AS min_energy,
AVG(energy__) AS avg_energy,

-- acousticness
MAX(acousticness__) AS max_acousticness,
MIN(acousticness__) AS min_acousticness,
AVG(acousticness__) AS avg_acousticness,

-- instrumentalness
MAX(instrumentalness__) AS max_instrumentalness,
MIN(instrumentalness__) AS min_instrumentalness,
AVG(instrumentalness__) AS avg_instrumentalness,

-- liveness
MAX(liveness__) AS max_liveness,
MIN(liveness__) AS min_liveness,
AVG(liveness__) AS avg_liveness,

-- speechiness
MAX(speechiness__) AS max_speechiness,
MIN(speechiness__) AS min_speechiness,
AVG(speechiness__) AS avg_speechiness

FROM `projeto-2-laboratoria-456917.database_projeto_2.track_technical_info`;
```

- Durante tratamento de variável `track_id` , ao efetuar correção de tipagem, de `STRING` para `INTERGER`, conforme descrição de variável (número inteiro de 7 dígitos que não se repete), foram identificados valores **fora do padrão de id, impossibilitando a transformação de tipagem**.

  Decisão: Serão retiradas, para considerarmos apenas valores válidos contendo apenas dígitos numéricos.

	Nome do campo	Tipo
☐	track_id	STRING

Linha	track_id
1	0:00

▼ **Transformações:**

1. . **Tabela** `track_in_spotify`**:**
   - Criação de View `vw1_track_in_spotify`:

- Retirando variáveis fora de escopo de análise e dados duplicados, utilizando `NOT IN`.
- Limpando variáveis com caracteres e símbolos estranhos, utilizando `REGEXP_REPLACE` , criando novas colunas `track_name_clean` e `artist_name_clean`.
- Limpando variável track_id, alterada para tipagem `INTERGER` com `CAST` e removendo item não totalmente numérico com `REGEX_CONTAINS`.
- Limpando variável `streams` , alterada para tipagem `INTERGER` com `CAST` e removendo item não totalmente numérico com `REGEX_CONTAINS`.
- Criando nova variável `release_date`, utilizando `CONCAT`, `CAST` e `LPAD` em variáveis de `released_year`, `released_month` e `released_day`, retornando formato aaaa-mm-dd.
- Criando nova variável `total_playlist_presence`, comatória das variáveis `in_spotify_playlists` e `in_spotify_charts` .

▼ **Query criação:**

```
CCREATE OR REPLACE VIEW `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_in_spotify` AS
SELECT
  -- Colunas originais
  CAST(track_id AS INT64) AS track_id,--converte para número
  track_name,
  artist_s__name,
  released_year,
  released_month,
  released_day,
  in_spotify_playlists,
  in_spotify_charts,
  CAST(streams AS INT64) AS streams,--converte para número

  -- Colunas limpas
  REGEXP_REPLACE(track_name, r'[^a-zA-Z0-9áéíóúãõâêîôûçÁÉÍÓÚÃÕÂÊÎÔÛÇ"()\-.,!?\n\r ]', '') AS track_name
  REGEXP_REPLACE(artist_s__name, r'[^a-zA-Z0-9áéíóúãõâêîôûçÁÉÍÓÚÃÕÂÊÎÔÛÇ"()\-.,!?\n\r ]', '') AS artist_na

  -- Nova variável: data no formato 'aaaa-mm-dd'
  PARSE_DATE('%Y-%m-%d', CONCAT(
  CAST(released_year AS STRING), '-',
  LPAD(CAST(released_month AS STRING), 2, '0'), '-',
  LPAD(CAST(released_day AS STRING), 2, '0')
)) AS release_date,

  -- Nova variável: presença total em playlists
  in_spotify_playlists + in_spotify_charts AS total_playlist_presence

FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_spotify`
WHERE
  REGEXP_CONTAINS(track_id, r'^\d+$') --garante que track_id seja numérico
  AND REGEXP_CONTAINS(streams, r'^\d+$') --retira valores não numéricos
  AND track_id NOT IN ("5080031"); --retira musica duplicada
```

2. **Tabela** `track_in_competition` **:**

- Criação de View `vw1_track_in_competition:`
  - Retirando variáveis fora de escopo de análise.
  - Limpando variável `track_id`, alterada para tipagem `INTERGER` com `CAST` e removendo item não totalmente numérico com `REGEX_CONTAINS`.

▼ **Query criação:**

```
CREATE OR REPLACE VIEW `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_in_competition` A
SELECT
  CAST(track_id AS INT64) AS track_id, -- convertido para número
  in_apple_playlists,
  in_apple_charts,
  in_deezer_playlists,
  in_deezer_charts
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_in_competition`
WHERE REGEXP_CONTAINS(track_id, r'^\d+$'); -- garante que track_id seja numérico
```

3. **Tabela** `track_technical_info`**:**
   - Criação de View `vw1_track_technical_info`:
     - Substituindo dados nulos (utilizando `COALESCE`).
     - Limpando variável `track_id`, alterada para tipagem `INTERGER` com `CAST` e removendo item não totalmente numérico com `REGEX_CONTAINS`.

▼ **Query criação:**

```
CREATE OR REPLACE VIEW `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_technical_info` A
SELECT
  CAST(track_id AS INT64) AS track_id,  -- convertido para número
  bpm,
  mode,
  danceability__,
  valence__,
  energy__,
  acousticness__,
  instrumentalness__,
  liveness__,
  speechiness__,
  COALESCE(key, 'desconhecido') AS key  -- Substituindo valores nulos por "desconhecido"
FROM `projeto-2-laboratoria-456917.database_projeto_2.track_technical_info`
WHERE REGEXP_CONTAINS(track_id, r'^\d+$');  -- garante que track_id seja numérico
```

4. **Junção de Tabelas:**
   - Criando View unificada `vw_final_tracks`, Utilizando `LEFT JOIN` :

▼ **Query junção:**

```
CREATE OR REPLACE VIEW `projeto-2-laboratoria-456917.dados_consolidados.vw_final_tracks` AS

-- Seleciona as colunas que queremos trazer das três tabelas
SELECT
  sp.track_id,                  -- ID único da música
  sp.track_name_clean,            -- Nome da música (tratado/limpo)
  sp.artist_name_clean,           -- Nome do(s) artista(s) (tratado/limpo)
  sp.release_date,              -- Data de lançamento no formato YYYY-MM-DD
  sp.in_spotify_playlists,        -- Número de playlists do Spotify que a música participa
  sp.in_spotify_charts,           -- Posição nas paradas do Spotify
  sp.streams,                   -- Número de streams no Spotify
```

```
  comp.in_apple_playlists,          -- Número de playlists da Apple Music
  comp.in_apple_charts,             -- Posição nas paradas da Apple Music
  comp.in_deezer_playlists,         -- Número de playlists do Deezer
  comp.in_deezer_charts,            -- Posição nas paradas do Deezer

  tech.bpm,                     -- Batidas por minuto da música
  tech.key,                    -- Tom musical
  tech.mode,                    -- Modo musical (Maior ou Menor)
  tech.danceability__,          -- Índice de dançabilidade
  tech.valence__,               -- Índice de positividade
  tech.energy__,                -- Nível de energia
  tech.acousticness__,          -- Quantidade de sons acústicos
  tech.instrumentalness__,      -- Quantidade de sons instrumentais
  tech.liveness__,              -- Presença de público ou som ao vivo
  tech.speechiness__            -- Presença de fala na música

-- Define a primeira tabela base: Spotify (sp)
FROM `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_in_spotify` AS sp

-- Faz um LEFT JOIN com a view de competição (comp)
LEFT JOIN `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_in_competition` AS comp
  ON sp.track_id = comp.track_id        -- Conecta pelo track_id

  -- Faz um LEFT JOIN com a view de informações técnicas (tech)
LEFT JOIN `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_technical_info` AS tech
  ON sp.track_id = tech.track_id;       -- Conecta pelo track_id
```

5. **Criação de Tabelas Auxiliares:**

- Criada tabela `solo_tracks` utilizando `WITH`, contendo o total de músicas por artista solo e total de streams:

▼ **Query criação:**

```
WITH solo_tracks AS ( --cria tabela temporária
  SELECT
    artist_name_clean, --seleciona o nome do artista
    COUNT(track_id) AS total_solo_tracks, --conta quantas musicas (track_id) esse artista tem
    SUM(streams) AS total_streams --soma total de streams
  FROM `projeto-2-laboratoria-456917.dados_consolidados.vw_final_tracks`
  WHERE NOT REGEXP_CONTAINS(artist_name_clean, r',') --filtrar para pegar somente artistas solo, sem vi
  GROUP BY artist_name_clean -- agrupa os resultados por artista para fazer a contagem -- total 301 result
)
SELECT *
FROM solo_tracks;

--teste
SELECT
  artist_name_clean,
  track_name_clean
FROM `projeto-2-laboratoria-456917.dados_consolidados.vw_final_tracks`
WHERE artist_name_clean LIKE 'Taylor Swift'; --teste com o nome Taylor Swift, para saber se a quantidade
```

- Criada tabela `partnered_tracks` utilizando `WITH`, contendo total de musicas com participação de mais de um artista:

▼ **Query criação:**

```
WITH total_partnered_tracks AS (
  SELECT
    artist_name_clean,
    track_name_clean,
    track_id
  FROM `projeto-2-laboratoria-456917.dados_consolidados.vw_final_tracks`
  WHERE REGEXP_CONTAINS(artist_name_clean, r',') -- identifica colaborações
)

-- Consulta final, usando a tabela temporária
SELECT *
FROM total_partnered_tracks;
```

- Criada tabela `tbl_final_tracks`, contendo a junção de tabelas, para importação no PowerBI.

  ▼ **Query criação:**

```
CREATE OR REPLACE TABLE `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks` AS
SELECT
  sp.track_id,
  sp.track_name_clean,
  sp.artist_name_clean,
  sp.release_date,
  sp.in_spotify_playlists,
  sp.in_spotify_charts,
  sp.streams,

  comp.in_apple_playlists,
  comp.in_apple_charts,
  comp.in_deezer_playlists,
  comp.in_deezer_charts,

  tech.bpm,
  tech.key,
  tech.mode,
  tech.danceability__,
  tech.valence__,
  tech.energy__,
  tech.acousticness__,
  tech.instrumentalness__,
  tech.liveness__,
  tech.speechiness__

-- Define a primeira tabela base: Spotify (sp)
FROM `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_in_spotify` AS sp

-- Faz um LEFT JOIN com a view de competição (comp)
LEFT JOIN `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_in_competition` AS comp
  ON sp.track_id = comp.track_id          -- Conecta pelo track_id

  -- Faz um LEFT JOIN com a view de informações técnicas (tech)
LEFT JOIN `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_technical_info` AS tech
  ON sp.track_id = tech.track_id;          -- Conecta pelo track_id
```

- Criada tabela ttl_playlists, contendo somatória de playlists de streams.

  ▼ **Query criação:**

```
CREATE OR REPLACE TABLE `projeto-2-laboratoria-456917.dados_consolidados.ttl_playlists` AS
SELECT
  sp.track_id,
  sp.track_name_clean,
  sp.artist_name_clean,
  sp.in_spotify_playlists,
  cp.in_deezer_playlists,
  cp.in_apple_playlists,
  -- nova variável somando tudo
  (IFNULL(sp.in_spotify_playlists, 0) +
   IFNULL(cp.in_deezer_playlists, 0) +
   IFNULL(cp.in_apple_playlists, 0)) AS total_playlists
FROM `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_in_spotify` sp
LEFT JOIN `projeto-2-laboratoria-456917.dados_consolidados.vw1_track_in_competition` cp
  ON sp.track_id = cp.track_id;
```

6. Análise Exploratória:

▼ **Agrupamento de variáveis categóricas em tabelas no Power BI:**

a. Número de músicas por artista, total de playlists por streams e total playlists:

- Campos:
  - `artist_name_clean` (categórica) → Linha
  - `track_id` (numérica) → Contagem distinta (número de músicas)
  - `in_apple_playlists`, `in_deezer_playlists`, `in_spotify_playlist` → Soma (total de playlists por streams)
  - `total_playlists` → Soma (totalizador de playlists por artista)
- Analise:
  - **Número de Músicas por Artista:** Identificar quais artistas possuem o maior número de músicas. Esse número pode ajudar a entender a presença de um artista ao longo do tempo.
  - **Total de Playlists por Streams:** Comparar a quantidade total de playlists em que as músicas de cada artista aparecem, levando em consideração também o impacto de cada plataforma de streaming (Apple, Deezer, Spotify) nos streams.
  - **Total de Playlists:** Verificar quais artistas têm mais presença em playlists no geral, considerando todas as plataformas de streaming.

▼ **Visualização de variáveis categóricas:**

a. Quantidade de músicas por artista:

- Gráfico: de colunas.
- Campos:
  - `artist_name_clean` (categórica) - Eixo X
  - `track_id` (numérica) → Contagem distinta (número de músicas) - Eixo Y
- Principais insights encontrados:
  - **Taylor Swift** lidera com o maior número de músicas cadastradas.
  - Seguida por **The Weeknd** e **Bad Bunny**, também com grande volume de lançamentos.

b. Quantidade de playlists por artista:

- Gráfico: de colunas.
- Campos:
  - `artist_name_clean` (categórica) - Eixo X
  - `total_playlists` (numérica) → Soma de total de playlists por artista - Eixo Y
- Principais insights encontrados:
  - **The Weeknd** é o artista com maior presença em playlists, indicando forte apelo nas plataformas de streaming.
  - Em seguida, **Ed Sheeran** e **Taylor Swift** também se destacam em visibilidade.

▼ **Aplicação de Medidas de Tendência Central:**

As medidas de tendência central são utilizadas para **resumir um conjunto de dados** em torno de um valor representativo. As principais são:

- **Média**: soma dos valores dividida pela quantidade de elementos.
- **Mediana**: valor central quando os dados estão em ordem crescente.
- **Moda**: valor que mais se repete.

a. Média e Mediana de `streams` :

Cálculada através de tabela dinâmica.

Média dos streams: `514.115.096,46`

Mediana dos streams: `289.165.138,5`

Análise:

- A **média é significativamente maior que a mediana**, o que indica:
  - Presença de músicas com **streams extremamente altos**, que elevam a média.
  - **Distribuição assimétrica à direita**, com a maioria das músicas abaixo da média e poucos hits extremamente populares.

Interpretação:

- A mediana indica que **metade das músicas têm menos de 289 milhões de streams**.
- Já a média alta mostra que **um pequeno grupo de músicas domina a audiência**, influenciando fortemente os números globais.

b. Média e Mediana de `total_playlists` :

Cálculada através de tabela dinâmica.

Média: `5.659,84`

Mediana: `2.302`

Análise:

- Assim como nos streams, há **diferença significativa entre média e mediana**, sugerindo:
  - Algumas músicas com **altíssima presença em playlists**.
  - **Maioria das faixas com inserções mais modestas**, o que puxa a mediana para baixo.

Interpretação:

- A mediana indica que **mais da metade das músicas aparecem em menos de 2.302 playlists**.
- A média, bem mais alta, reforça a concentração: **um número pequeno de músicas domina as playlists**.

c. Gráfico de disperção:

- **Eixo X** → `total_playlists`
- **Eixo Y** → `streams`
- **Legenda**→ `track_name_clean` *(Para identificação dos pontos).*

O gráfico mostra uma **relação positiva** entre o número de playlists e a quantidade de streams:

→ **Quanto mais playlists uma música aparece, mais streams ela tende a ter.**

Também observamos **outliers**:

- Músicas com muitos streams e poucas playlists (possivelmente virais).
- Músicas em muitas playlists, mas com menos streams (menor engajamento).

Conclusão:

Estar em playlists ajuda a aumentar os streams, mas **não é o único fator**. Popularidade orgânica e engajamento também influenciam.

▼ **Histograma com Python:**

a. Instalação do Python:

- Download: https://www.python.org/downloads/
- Ao executar o instalador, **importante** marcar a opção **“Add Python to PATH”** antes de clicar em **“Install Now”**.
- Verificação de instalação: no **terminal** `python --version` .

b. Integração com PowerBi:

- PowerBi → **Arquivo > Opções e configurações > Opções**.
- Aba **Script do Python** (em “Global”), campo para indicar o caminho de instalação do Python.

  `C:\Users\SeuNome\AppData\Local\Programs\Python\Python3x` **OK**.
- Em **terminal**: `pip install matplotlib pandas` (garante que os pacotes usados para o histograma funcionem corretamente no Power BI.)

c. Histograma - `streams`:

▼ **Criação:**

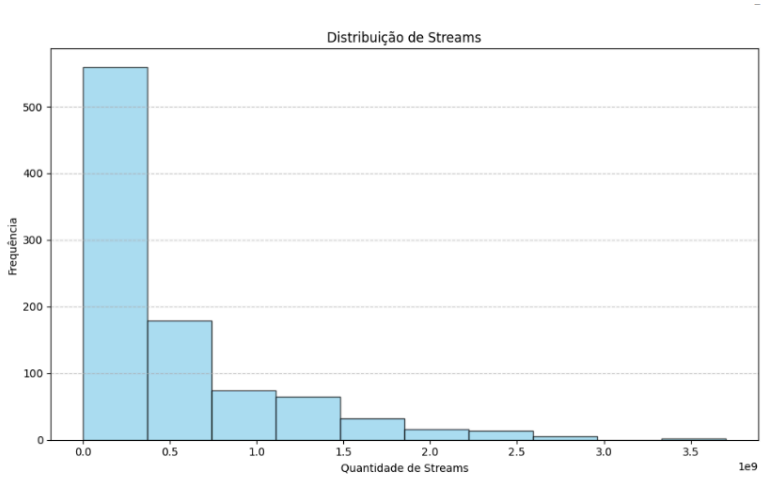
- **PowerBi** → Relatório > Visualizações > PY (Visual Python).
  - Em **valores**, coluna `streams` → Não Resumir.

- Em **Editor de script do Python**

▼ **Código:**

```
import matplotlib.pyplot as plt
import pandas as pd

# Power BI passa os dados selecionados nesta variável chamada 'dataset'
data = dataset[['streams']].dropna()  # remove valores ausentes se houver

# Criação do histograma
plt.figure(figsize=(10,6))
plt.hist(data['streams'], bins=10, color='skyblue', edgecolor='black', alpha=0.7)

# Rótulos e título
plt.xlabel('Quantidade de Streams')
plt.ylabel('Frequência')
plt.title('Distribuição de Streams')

# Exibe o gráfico
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```

▼ **Resultado:**



▼ **Analise:**

- **Alta concentração de streams baixos**: Muitos itens têm um número relativamente baixo de streams, com apenas algumas músicas ou artistas alcançando números muito altos.
- **Poucas músicas ou artistas muito populares**: A maioria tem números baixos, mas um pequeno número se destaca com uma quantidade muito maior de streams. Isso é comum em plataformas de streaming, onde a maioria dos itens tem uma audiência pequena, mas os "sucessos" têm números muito maiores.

d. Histograma - `total_playlists:`

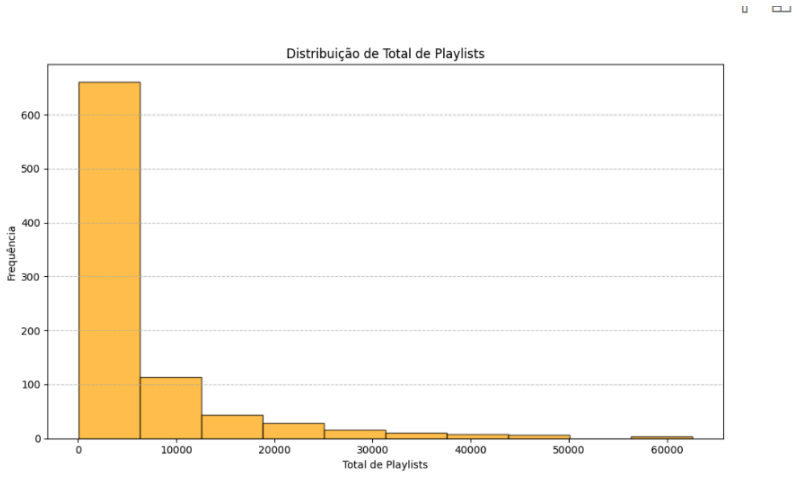
▼ **Criação:**

- **PowerBi** → Relatório > Visualizações > PY (Visual Python).
  - Em **valores**, coluna `total_playlists` → Não Resumir.
  - Em **Editor de script do Python**

▼ **Código:**

```
import matplotlib.pyplot as plt
import pandas as pd

# Power BI passa os dados nesta variável chamada 'dataset'
data = dataset[['total_playlists']].dropna()

# Criação do histograma
plt.figure(figsize=(10,6))
plt.hist(data['total_playlists'], bins=10, color='orange', edgecolor='black', alpha=0.7)

# Rótulos e título
plt.xlabel('Total de Playlists')
plt.ylabel('Frequência')
plt.title('Distribuição de Total de Playlists')

# Grade e layout
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```

▼ **Resultado:**



▼ **Analise:**

- **Alta concentração de playlists baixas**: A maioria dos artistas ou músicas está presente em poucas playlists, indicando uma grande concentração de valores baixos. Isso sugere que muitos conteúdos ainda não têm

ampla visibilidade nas plataformas.

- **Poucos artistas ou músicas altamente populares**: Um número pequeno de itens aparece em muitas playlists, o que puxa a média para cima e gera a cauda longa à direita. Esses casos representam os conteúdos de maior sucesso e visibilidade nas plataformas, uma dinâmica comum no mercado musical, onde poucos atingem grande alcance.
- **Distribuição assimétrica positiva**: A média significativamente maior que a mediana indica que a distribuição é enviesada à direita, reforçando a presença de valores extremos altos (outliers).
- **Oportunidade de segmentação**: Essa variável pode ser usada para identificar artistas em diferentes estágios de popularidade, desde emergentes (com poucas playlists) até consolidados (com ampla presença).

▼ **Medidas de Dispersão - Desvio Padrão:**

> 🔎 **Desvio padrão** é uma medida de dispersão que mostra **o quanto os valores de uma variável se afastam da média**. (Maior → mais variáveis dispersas. Menor → mais concentradas ao redor da média).

a. Desvio Padrão - `streams` :

▼ **Criação**:

- Em PowerBi → Dados → tabela `tbl_final_tracks` → selecionar **Nova Medida**

```
DesvioP_Streams = STDEV.P('solo_tracks'[streams])
```

- Em **Visualizações** → selecionar **Cartão** → Campo **Valores**: `DesvioP_Streams` .

▼ **Resultado:**

- Desvio Padrão: 567,08 Mi.

▼ **Analise:**

- **Desvio padrão elevado** (`567,08 milhões`) indica que **os valores de streams variam muito em relação à média**.
- A **média de streams** é de aproximadamente `514 milhões` , mas a **mediana é bem menor** (`289 milhões`), o que mostra que **a maior parte das músicas tem menos streams que a média**.
- Em termos de distribuição, isso confirma que **o mercado é altamente concentrado em poucos grandes sucessos**, enquanto a maioria dos artistas ou faixas tem performance mais modesta.
- O **alto desvio padrão**, combinado com a **assimetria da distribuição**, mostra que a **popularidade nas plataformas é desigual**, com destaque para um pequeno grupo de músicas que dominam a audiência.

b. Desvio Padrão - `total_playlists` :

▼ **Criação**:

- Em PowerBi → Dados → tabela `ttl_playlists` → selecionar **Nova Medida**

```
DesvioP_Streams = STDEV.P('ttl_playlists'[total_playlists])
```

- Em **Visualizações** → selecionar **Cartão** → Campo **Valores**: `DesvioP_Playlists` .

▼ **Resultado:**

- Desvio Padrão: 8,92 Mil.

▼ **Analise:**

- **Desvio padrão de 8,92 mil**: Esse valor indica que há uma **variação considerável** no número de playlists em que as músicas estão presentes, embora seja uma variação mais moderada em comparação com outras

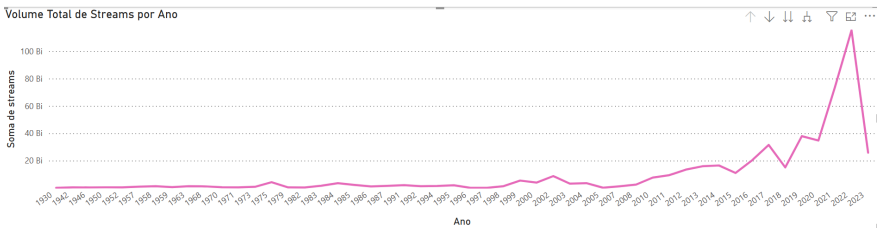
métricas do mercado. A presença em playlists não segue uma distribuição uniforme, com algumas músicas muito mais expostas que outras.

- O **alto desvio padrão**, combinado com a grande diferença entre **média e mediana**, reflete uma **concentração em poucos sucessos** ou músicas mais promovidas, que têm uma presença massiva nas playlists, enquanto a maioria das músicas é distribuída em um número reduzido de playlists.

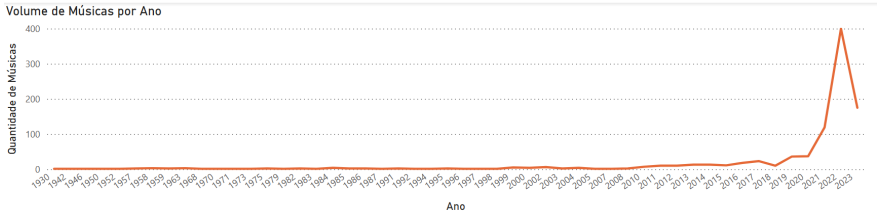
▼ **Comportamento de Dados ao Longo do Tempo:**

Criação de Gráficos de Linha para as variáveis:

▼ `Streams` **por ano:**



▼ **Músicas** `Track_id` **por ano:**



▼ **Quartis e Categorização de Variáveis no BigQuery:**

▼ **Query de cálculo de quartis** de variáveis**:**

```
SELECT
  APPROX_QUANTILES(danceability__, 4) AS quartis
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
---
SELECT
  APPROX_QUANTILES(valence__, 4) AS quartis
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
---
SELECT
  APPROX_QUANTILES(energy__, 4) AS quartis
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
---
SELECT
  APPROX_QUANTILES(acousticness__, 4) AS quartis
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
---
SELECT
  APPROX_QUANTILES(instrumentalness__, 4) AS quartis
```

```
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`
WHERE instrumentalness__ IS NOT NULL;
---
SELECT
  APPROX_QUANTILES(liveness__, 4) AS quartis
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
---
SELECT
  APPROX_QUANTILES(speechiness__, 4) AS quartis
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
```

- **Quartis:**
  - ▼ `danceability__` :

    Q1: 57

    Q2: 69

    Q3: 78
  - ▼ `valence__` :

    Q1: 32

    Q2: 51

    Q3: 70
  - ▼ `energy__` :

    Q1: 53

    Q2: 66

    Q3: 77
  - ▼ `acousticness__`:

    Q1: 6

    Q2: 18

    Q3: 43
  - ▼ `instrumentalness__`:

    Q1: 0

    Q2: 0

    Q3: 0

    Indica que mais de 75% das músicas têm valor zero nessa métrica.

    - Essa variável está extremamente concentrada no valor zero, indicando que a maioria das faixas analisadas não são instrumentais.
    - Apenas uma minoria (os 25% superiores) possui algum grau de instrumentalidade significativa.
  - ▼ `liveness__` :

    Q1: 9

    Q2: 12

    Q3: 23
  - ▼ `speechiness__` :

Q1: 4

Q2: 6

Q3: 11

▼ **Query de criação de tabela** `tbl_tracks_technical_category`, contendo a **categorização de variáveis**:

```
CREATE OR REPLACE TABLE `projeto-2-laboratoria-456917.dados_consolidados.tbl_tracks_technical_category` AS
SELECT
 *,

 -- Danceability
 CASE
  WHEN danceability__ <= 57 THEN 'Baixo'
  WHEN danceability__ <= 69 THEN 'Médio'
  WHEN danceability__ <= 78 THEN 'Alto'
  ELSE 'Muito Alto'
 END AS categoria_danceability,

 -- Valence
 CASE
  WHEN valence__ <= 32 THEN 'Muito Triste'
  WHEN valence__ <= 51 THEN 'Triste'
  WHEN valence__ <= 70 THEN 'Feliz'
  ELSE 'Muito Feliz'
 END AS categoria_valence,

 -- Energy
 CASE
  WHEN energy__ <= 53 THEN 'Baixa'
  WHEN energy__ <= 66 THEN 'Média'
  WHEN energy__ <= 77 THEN 'Alta'
  ELSE 'Muito Alta'
 END AS categoria_energy,

 -- Acousticness
 CASE
  WHEN acousticness__ <= 6 THEN 'Pouco Acústica'
  WHEN acousticness__ <= 18 THEN 'Moderadamente Acústica'
  WHEN acousticness__ <= 43 THEN 'Acústica'
  ELSE 'Muito Acústica'
 END AS categoria_acousticness,

 -- Instrumentalness - não foi criada por quartis - utilizado (0, 50%)
 CASE
  WHEN instrumentalness__ = 0 THEN 'Baixa'
  WHEN instrumentalness__ <= 50 THEN 'Média'
  ELSE 'Alta'
 END AS categoria_instrumentalness,

 -- Liveness
 CASE
  WHEN liveness__ <= 9 THEN 'Pouca Presença ao Vivo'
  WHEN liveness__ <= 12 THEN 'Média Presença ao Vivo'
```

```
  WHEN liveness__ <= 23 THEN 'Alta Presença ao Vivo'
  ELSE 'Muito Alta Presença ao Vivo'
 END AS categoria_liveness,

 -- Speechiness
 CASE
  WHEN speechiness__ <= 4 THEN 'Pouco Falada'
  WHEN speechiness__ <= 6 THEN 'Moderadamente Falada'
  WHEN speechiness__ <= 11 THEN 'Bastante Falada'
  ELSE 'Muito Falada'
 END AS categoria_speechiness

FROM
 `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
```

▼ **Correlação entre variáveis:**

🔎 **Coeficiente de correlação de Pearson**

1: Correlação positiva forte

0: Sem correlaçao relevante

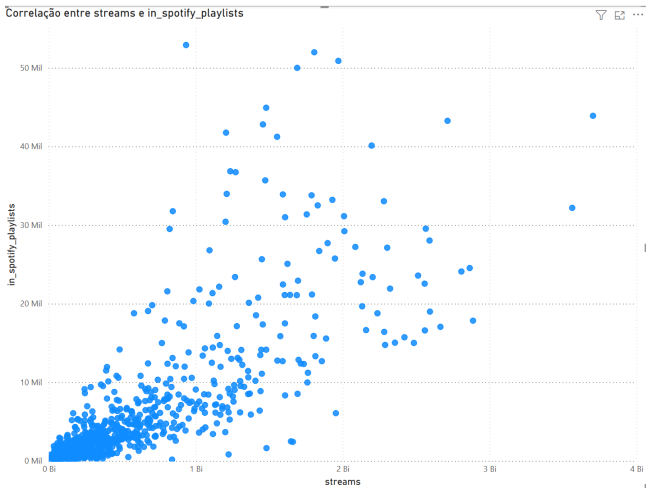
-1: Correlação negativa

▼ **Correlação (** `CORR` **) entre** `streams` **e** `in_spotify_playlists` **:** `0,79`

▼ **Query:**

```
SELECT
  CORR(streams, in_spotify_playlists) AS correlacao_streams_playlists
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
```

- **Correlação forte e positiva**, o que significa que:
  - Quanto **mais playlists** uma música aparece, **maior tende a ser seu número de streams**.
  - Playlists têm um papel essencial na promoção de músicas. Estar presente em várias playlists contribui significativamente para o alcance e o número de streams, reforçando sua importância como canal de descoberta nas plataformas.

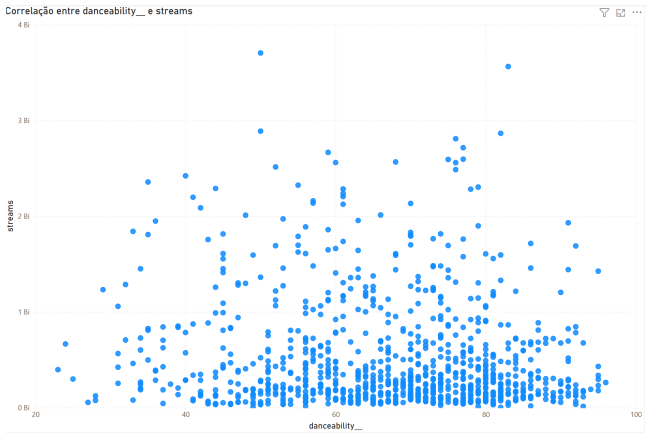


▼ **Correlação (** `CORR` **) entre** `streams` **e** `danceability__` **:** `-0,11`

▼ **Query:**

```
SELECT
  CORR(streams, danceability__) AS correlacao_streams_danceability
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
```

- **Correlação fraca e negativa**, o que significa que:
  - Isso indica que **a característica "danceability" não é um fator decisivo** para o sucesso de uma música em termos de streams. Outros elementos, como visibilidade em playlists ou popularidade do artista, provavelmente têm mais influência no alcance da faixa.



7. Técnica de Analise:

▼ **Segmentação:**

🔎 Análise da variação do número médio de **streams** entre categorias definidas pelos **quartis** das características musicais, calculadas durante a análise exploratória. Essa segmentação busca identificar possíveis relações entre atributos técnicos das músicas e seu desempenho em termos de popularidade na plataforma.

a. **Segmentação** `danceability__` :

Categoria	Média de streams
Baixo	**592442106.33**
Médio	515911337.85
Alto	515000534.96
Muito Alto	427583874.89

Observa-se **tendência inversa** entre nível de “dançabilidade” e o número médio de streams.

- Músicas com categoria **Baixo** apresentam, em média, **mais streams**.
- Músicas com categoria **Muito Alto** registram média **menor**.

Isso sugere que, para esta base de dados, **músicas mais dançantes não necessariamente têm melhor desempenho em streams**, o que pode refletir preferências do público ou estratégias de promoção distintas.

b. **Segmentação** `valence_` :

Categoria	Média de streams
Muito Triste	520173193.76
Triste	**560640466.80**
Feliz	503539643.53
Muito Feliz	471270398.19

Observa-se **tendência inversa** entre o nível de “valência” (positividade emocional) e o número médio de streams.

- Músicas com categoria **Triste** apresentam, em média, **mais streams**.
- Músicas com categoria **Muito Feliz** registram média **menor**.

Isso sugere que, para esta base de dados, **músicas excessivamente alegres não necessariamente têm melhor desempenho em streams**, indicando que tonalidades emocionais mais neutras ou melancólicas podem ressoar mais com o público ou receber maior destaque nas plataformas.

c. **Segmentação** `energy_` :

Categoria	Média de streams
Baixa	**548708501.10**
Média	518188269.58
Alta	490695325.19
Muito Alta	496378508.52

Observa-se **leve tendência decrescente** entre os níveis de “energia” e o número médio de streams.

- Músicas com categoria **Baixa energia** apresentam, em média, **mais streams**.
- Músicas com categoria **Alta** e **Muito Alta energia** registram médias **menores**.

Isso sugere que, para esta base de dados, **músicas com energia mais moderada ou baixa tendem a ter melhor desempenho em streams**, o que pode indicar uma preferência do público por sons menos intensos ou estratégias de curadoria que favorecem esse tipo de música.

d. **Segmentação** `acousticness_` :

Categoria	Média de streams
Pouco Acústica	**584466274.61**
Modeiradamente Acústica	459884317.49
Acústica	461770563.42
Muito Acústica	538446485.19

Observa-se **uma tendência não linear** entre os níveis de “acústica” e o número médio de streams.

- Músicas **Pouco Acústicas** apresentam a **maior média de streams**.
- Músicas **Moderadamente Acústicas** e **Acústicas** têm médias **menores**.
- Músicas **Muito Acústicas** voltam a apresentar uma média mais alta.

Isso sugere que, nesta base de dados, **músicas com extremos de "acústica" (muito baixa ou muito alta)** podem ter melhor desempenho em streams, enquanto aquelas com níveis médios tendem a performar menos, o que pode refletir nichos de público bem definidos ou variações de uso em playlists.

e. **Segmentação** `instrumentalness_` :

Categoria	Média de `streams`
Baixa	**521339236.68**
Média	462269816.87
Alta	289880448.30

Observa-se **uma tendência clara** de que músicas **menos instrumentais** tendem a ter **mais streams**.

- Músicas com *baixa* instrumental apresentam a **maior média de streams**.
- Músicas com *média* instrumental têm uma média **intermediária**.
- Músicas com *alta* instrumental registram a **menor média de streams**.

Esse padrão sugere que, para esta base de dados, **músicas com vocais são mais populares**, possivelmente por serem mais atrativas para um público amplo ou mais presentes em playlists populares.

f. **Segmentação** `liveness_` :

Categoria	Média de `streams`
Pouca presença ao vivo	**573136871.43**
Média presença ao vivo	484018326.98
Alta presença ao vivo	510693582.21
Muito Alta presença ao vivo	490676719.168

Observa-se que **músicas com menor presença ao vivo** tendem a ter **mais streams em média**.

- Músicas com **Pouca presença ao vivo** registram a **maior média de streams**.
- As demais categorias (**Média**, **Alta** e **Muito Alta presença ao vivo**) apresentam **médias inferiores**, com variação moderada entre elas.

Esse resultado sugere que, nesta base, **músicas com características mais "de estúdio" são mais populares**, podendo refletir a preferência dos ouvintes por produções com menos elementos ao vivo ou menos improviso.

g. **Segmentação** `speechiness_` :

Categoria	Média de `streams`
Pouco falada	**559296860.76**
Moderadamene falada	545200653.63
Bastante falada	525776009.25
Muito Alta	413725777.29

Observa-se uma **tendência decrescente** entre o nível de "fala" na música e a média de streams.

- Músicas **Pouco faladas** apresentam a **maior média de streams**.
- À medida que o conteúdo falado aumenta, a **média de streams diminui**, sendo a menor registrada na categoria **Muito Alta**.

Isso indica que, para esta base, **músicas com menos elementos falados tendem a ter melhor desempenho**, possivelmente por serem mais alinhadas ao estilo musical predominante nas plataformas ou por facilitarem a

experiência de escuta contínua.

▼ **Validação de Hipóteses:**

1. **Músicas com BPM (Batidas Por Minuto) mais altos tendem a ter mais streams no Spotify.**

   Validação realizada através de cálculo de correlação entre `bpm` e `streams` usando o comando `CORR(bpm, streams)` no BigQuery.

   ▼ **Query**:

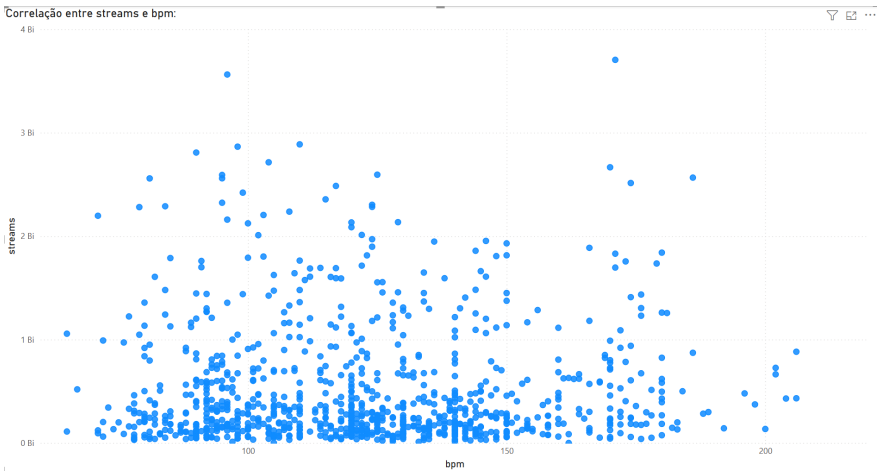
   ```
   SELECT
     CORR(bpm, streams) AS correlacao_streams_bpm
   FROM
     `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
   ```

   Resultado:

   A correlação entre BPM (Batidas Por Minuto) e streams foi `-0.0023`, o que indica uma correlação muito fraca e negativa entre essas duas variáveis.

   Interpretação:

   - BPM e Streams têm uma relação muito fraca e negativa. Isso sugere que, para essa base de dados, não há uma correlação significativa entre o número de batidas por minuto de uma música e o número de streams que ela recebe.
   - **Conclusão**: **Músicas com BPM mais altos não tendem a ter mais streams no Spotify**, **contrariando a hipótese inicialmente levantada**.

2. **As músicas mais populares no ranking do Spotify também apresentam alto desempenho em outras plataformas, como Deezer.**

   Validação realizada através de cálculo de correlação entre:

   - `in_spotify_charts` e `in_deezer_charts`

     ▼ **Query:**

     ```
     SELECT
       CORR(in_spotify_charts, in_deezer_charts) AS correlacao_charts
     FROM
       `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
     ```

   - `in_spotify_playlists` e `in_deezer_playlists`

▼ **Query:**

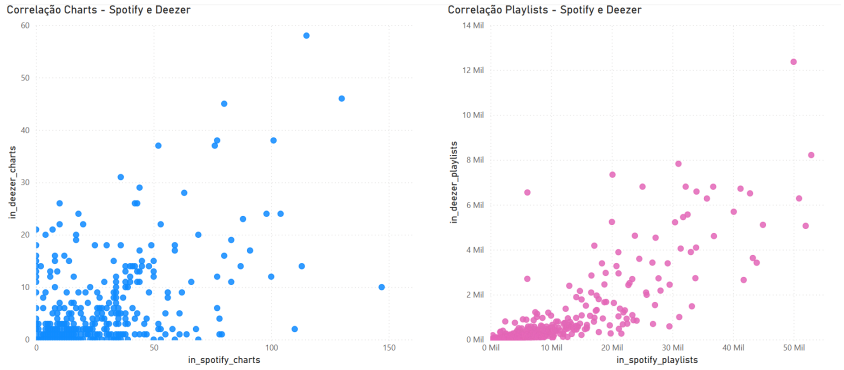
```
SELECT
  CORR(in_spotify_playlists, in_deezer_playlists) AS correlacao_playlists
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
```

Resultado:

- A correlação entre presença nos **charts** do Spotify e do Deezer foi de `0.6003`, indicando uma correlação moderada e positiva entre as duas variáveis.
- A correlação entre **playlists** do Spotify e do Deezer foi de `0.8264`, indicando uma correlação forte e positiva entre as duas variáveis.

Interpretação:

- Existe uma relação significativa entre o desempenho das músicas nas duas plataformas, tanto em rankings quanto em presença em playlists.
- Isso sugere que **músicas populares no Spotify também tendem a ser populares no Deezer, apoiando a hipótese**.
- A alta correlação de presença em playlists mostra uma provável semelhança entre as curadorias editoriais ou uma forte tendência do mercado em promover os mesmos artistas em múltiplos serviços.



3. **Existe uma correlação positiva entre o número de playlists em que uma música aparece e seu número total de streams.**

   Validação realizada através de cálculo de correlação entre `in_spotify_playlists` e `streams`

   ▼ **Query:**

   ```
   SELECT
     CORR(streams, in_spotify_playlists) AS correlacao_streams_playlists
   FROM
     `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
   ```

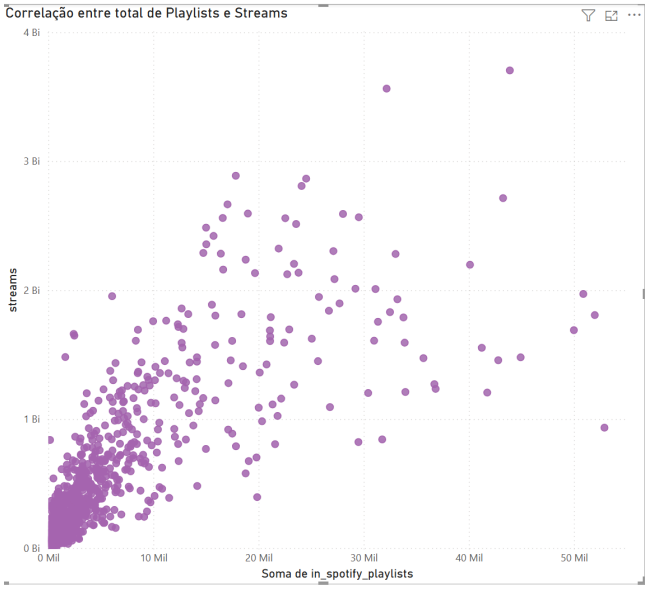
   Resultado:

   - A correlação entre o número de playlists e o número total de streams foi de `0.7901`, indicando uma correlação forte e positiva.

   Interpretação:

   Isso sugere que quanto mais playlists uma música aparece, maior tende a ser o número de streams. Playlists funcionam como um canal de descoberta eficiente para músicas, impactando diretamente no seu sucesso.

Portanto, a hipótese de que **existe uma correlação positiva entre o número de playlists em que uma música aparece e seu número total de streams** é **confirmada** com base nos dados analisados.



4. **Artistas com um maior número de músicas disponíveis no Spotify tendem a ter um volume maior de streams totais.**

   Validação realizada através de cálculo de correlação entre `total_solo_tracks` e `total_streams`

   ▼ **Query:**

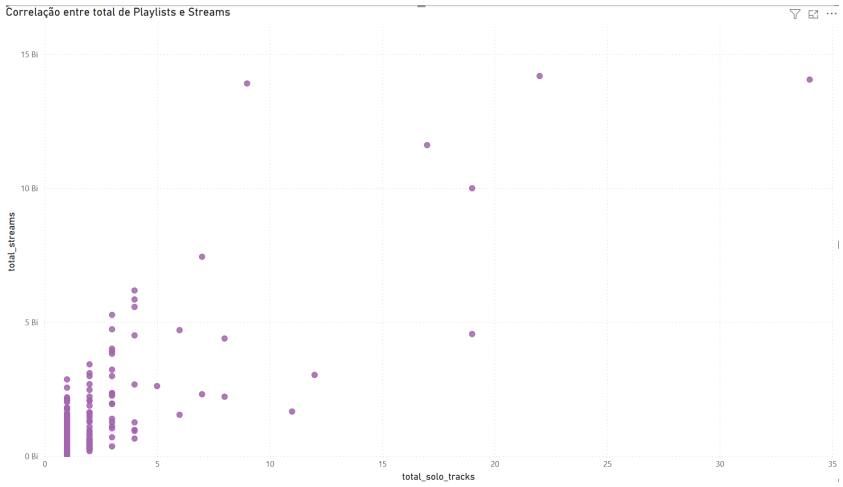
```
SELECT
  CORR(total_solo_tracks, total_streams) AS correlacao_musicas_streams
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.solo_tracks`;
```

   Resultado:
   A correlação entre o número de músicas de artistas solo e o número total de streams é `0.8045`, indicando uma correlação positiva forte entre essas duas variáveis.

   Interpretação:

   Isso sugere que artistas com mais músicas disponíveis tendem a ter mais streams no total, o que é consistente com a ideia de que mais músicas no catálogo aumentam as chances de atrair mais ouvintes e, consequentemente, gerar mais streams.

   Portanto, a hipótese de que **artistas com um maior número de músicas disponíveis no Spotify tendem a ter um volume maior de streams totais é confirmada com base nos dados analisados**.



5. **Características específicas das músicas (como energia, dançabilidade, valência, acústica, etc.) influenciam significativamente o número de streams no Spotify.**

   Validação realizada através do cálculo de correlação entre `streams` e variáveis de características musicais, utilizando a função `CORR()` no BigQuery.

   ▼ **Querys:**

```
SELECT
  CORR(streams, danceability__) AS correlacao_streams_danceability
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;

SELECT
  CORR(streams, valence__) AS correlacao_streams_valence
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;

SELECT
  CORR(streams, energy__) AS correlacao_streams_energy
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;

SELECT
  CORR(streams, acousticness__) AS correlacao_streams_acousticness
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;

SELECT
  CORR(streams, instrumentalness__) AS correlacao_streams_instrumentalness
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;

SELECT
  CORR(streams, liveness__) AS correlacao_streams_liveness
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;

SELECT
  CORR(streams, speechiness__) AS correlacao_streams_speechiness
```

```
FROM
  `projeto-2-laboratoria-456917.dados_consolidados.tbl_final_tracks`;
```

Resultados:

Característica	Correlação com `streams`	Interpretação
`danceability__`	-0,1059	Muito fraca e negativa
`valence__`	-0,0409	Muito fraca e negativa
`energy__`	-0,0259	Muito fraca e negativa
`acousticness__`	-0,0045	Muito fraca e negativa
`instrumentalness__`	-0,0449	Muito fraca e negativa
`liveness__`	-0,0488	Muito fraca e negativa
`speechiness__`	-0,1122	Muito fraca e negativa

Interpretação:

As características específicas analisadas apresentam **correlação muito fraca** com o número de streams no Spotify. Isso indica que, isoladamente, nenhuma dessas variáveis tem influência significativa sobre o sucesso de uma música em termos de reproduções.

Conclusão: A hipótese de que essas características influenciam significativamente o número de streams **não é confirmada** com base nos dados analisados.

8. Conclusões:

▼ **Principais descobertas:**

1. **Produtividade vs Popularidade:**
   - Embora **Taylor Swift** tenha o maior número de músicas cadastradas, **The Weeknd** lidera em presença em playlists, o que sugere que **quantidade de músicas não significa necessariamente mais visibilidade** nas plataformas.
   - Isso indica que **a curadoria das plataformas favorece certas músicas ou artistas**, mesmo que eles tenham um catálogo menor.
2. **Influência das Playlists no Alcance:**
   - Artistas com maior presença em playlists — como **The Weeknd** e **Ed Sheeran** — tendem a alcançar um público mais amplo, pois **playlists são uma via direta de descoberta para novos ouvintes**.
   - Isso reforça **a importância de estar em playlists populares para aumentar os streams.**
3. **Diferenciação entre plataformas:**
   - A presença variada de artistas em playlists do Spotify, Apple e Deezer pode indicar **diferenças na curadoria e público-alvo entre as plataformas**.
   - Um artista pode performar melhor em uma plataforma do que em outra, o que pode orientar estratégias de marketing musical.
4. **Correlação Positiva entre Playlists e Streams:**
   - Existe uma **relação direta**: músicas presentes em mais playlists **tendem a ter mais streams**.
   - Porém, há exceções: algumas músicas viralizam com poucas playlists, e outras aparecem muito sem alcançar grande volume de streams.
5. **Distribuição Assimétrica dos Dados:**
   - Tanto os **streams** quanto o **total de playlists** apresentam distribuições assimétricas, com alta concentração de valores baixos e **caudas longas à direita**.

- A **média é muito maior que a mediana** em ambas as variáveis, indicando a presença de **outliers** — poucos artistas ou músicas com números extremamente altos.
- Isso reflete a lógica das plataformas de streaming, onde **o mercado é dominado por poucos grandes sucessos**, enquanto a maioria dos conteúdos tem alcance mais limitado.

6. **Segmentação por características musicais (via quartis):**
   - **Danceability**: Músicas menos dançantes têm, em média, **mais streams**. As mais dançantes performam pior, sugerindo que **dançabilidade alta não garante popularidade**.
   - **Valence**: Músicas *tristes* têm média de streams maior do que *felizes*, sugerindo que **músicas emocionalmente neutras ou melancólicas atraem mais ouvintes**.
   - **Energy**: Faixas com *baixa energia* lideram em média de streams, sugerindo **preferência por músicas menos intensas**.
   - **Acousticness**: Músicas *pouco acústicas* tiveram as maiores médias de streams. Faixas mais digitais parecem ser **mais bem recebidas**.
   - **Instrumentalness**: Músicas com *baixa* instrumental obtiveram significativamente mais streams, o que reforça a **importância da presença vocal** no desempenho.
   - **Liveness**: Músicas com *pouca presença ao vivo* tiveram maiores médias, indicando **preferência por gravações de estúdio**.
   - **Speechiness**: Quanto *menos falada* a música, maior a média de streams, indicando que **excesso de conteúdo falado reduz a popularidade**.
7. **Hipoteses:**
   - **Músicas com BPM mais altos não tendem a ter mais streams no Spotify.**
     - Correlação praticamente nula (-0,0023) entre BPM e número de streams, **refutando a hipótese**.
   - **Músicas populares no Spotify também apresentam bom desempenho no Deezer.**
     - Correlação moderada (0,60) entre presença nos charts das duas plataformas.
     - Correlação forte (0,82) entre presença em playlists de ambas.
     - **Hipótese confirmada**: há consistência no sucesso entre as plataformas.
   - **Existe correlação positiva entre número de playlists e streams.**
     - Correlação forte (0,79) entre total de playlists e streams.
     - **Hipótese confirmada**: estar em mais playlists está associado a mais streams.
   - **Artistas com mais músicas tendem a ter mais streams totais.**
     - Correlação forte (0,80) entre quantidade de músicas por artista e total de streams.
     - **Hipótese confirmada**: ter um portfólio maior está ligado a maior volume de reprodução.
   - **Características específicas das músicas não influenciam significativamente o número de streams.**
     - Todas as correlações foram muito fracas e negativas (ex: `speechiness_`: -0,11).
     - **Hipótese refutada**: individualmente, essas variáveis não explicam o sucesso em streams.

▼ **Recomendações:**

1. **Priorizar a inclusão em playlists relevantes**

Justificativa:

Foi observada uma **correlação positiva e significativa entre o número de playlists e o volume de streams**, especialmente para artistas como The Weeknd e Ed Sheeran, que se destacam mesmo com catálogos menores que o de outros nomes como Taylor Swift.

Recomendações práticas:

- Estabelecer relacionamento com curadores editoriais e independentes.
- Acompanhar playlists em crescimento e buscar inserção estratégica.
- Criar versões de faixas adaptáveis a diferentes estilos de playlist (ex: acústica, remix, versão curta).
- Utilizar dados analíticos para identificar playlists com maior retorno de audiência.

2. **Focar em qualidade e posicionamento estratégico, não apenas em volume**

Justificativa:

Apesar de artistas com muitos lançamentos (ex: Taylor Swift) dominarem em quantidade, os dados mostram que **a presença em playlists e a curadoria editorial são mais determinantes** para o sucesso que o número absoluto de faixas.

Recomendações práticas:

- Planejar lançamentos com base em dados de mercado e janelas estratégicas.
- Trabalhar campanhas de pré-lançamento com teasers, colaborações e conteúdos paralelos.
- Medir desempenho individual de cada faixa, ao invés de lançar álbuns completos sem acompanhamento tático.

3. **Customizar estratégias para cada plataforma de streaming**

Justificativa:

A presença dos artistas varia significativamente entre Spotify, Apple Music e Deezer, revelando **diferentes perfis de público e curadoria** em cada plataforma.

Recomendações práticas:

- Desenvolver campanhas de marketing específicas por plataforma.
- Analisar onde o artista performa melhor e ampliar presença nesse ambiente.
- Adaptar o conteúdo às diretrizes e tendências de cada serviço (ex: exclusividades, formatos curtos, lyrics, visualizações).

4. **Aproveitar a lógica de assimetria no consumo musical**

Justificativa:

Os dados revelam uma **distribuição assimétrica** de streams: poucas músicas concentram a maioria do tráfego, enquanto a maioria tem baixo alcance — reflexo da dinâmica do mercado digital, dominado por "blockbusters".

Recomendações práticas:

- Identificar as “faixas líderes” e derivar delas novos conteúdos (remixes, versões acústicas, colaborações).
- Reforçar campanhas para faixas de catálogo antigo com potencial ainda inexplorado.
- Realocar investimento de marketing conforme desempenho real por faixa.

5. **Evitar basear decisões exclusivamente em características musicais**

Justificativa:

Análises de correlação entre `streams` e variáveis como `danceability`, `energy`, `valence`, `acousticness`, `instrumentalness`, `liveness` e `speechiness` mostraram **correlações fracas ou quase nulas** com o volume de execuções.

Recomendações práticas:

- Usar características musicais como complemento, e não base para criação.
- Priorizar fatores como identidade artística, contexto cultural, narrativa e autenticidade.
- Combinar dados técnicos com feedback qualitativo de público-alvo.

6. **Investir fortemente em branding e identidade artística**

Justificativa:

O sucesso de artistas em playlists e charts está associado à **força de marca, reconhecimento e consistência de imagem**, além da qualidade musical em si.

Recomendações práticas:

- Desenvolver uma identidade visual e sonora coesa.
- Trabalhar a presença digital com conteúdos autênticos e engajamento em redes sociais.
- Colaborar com outros artistas e influenciadores para ampliar alcance.
- Usar estratégias de storytelling para fortalecer o vínculo emocional com os fãs.

7. **Atuar de forma ágil para aproveitar tendências e virais**

Justificativa:

Algumas faixas viralizam independentemente do apoio editorial inicial — a **viralização orgânica ainda é possível**, embora rara.

Recomendações práticas:

- Monitorar dados em tempo real para identificar movimentos de crescimento inesperado.
- Criar estratégias de amplificação rápida quando uma faixa começa a ganhar tração.
- Estimular conteúdos gerados por fãs (fan content) e desafios em redes sociais como TikTok e Instagram.

8. **Otimizar a composição com base em padrões de consumo**

Justificativa:

Faixas com menos `speechiness`, `instrumentalness`, `acousticness` e `liveness` obtiveram desempenho superior, sugerindo **preferência por músicas vocais, limpas e produzidas em estúdio**.

Recomendações práticas:

- Evitar excesso de falas, experimentalismos acústicos ou versões ao vivo quando o foco for o desempenho comercial.
- Priorizar produção moderna com elementos pop e apelo digital.

9. Limitações:

▼ **Limitações de dados e projeto:**

- **Dados Limitados:** A análise foi realizada com base em um recorte específico de músicas e anos, que não representa todo o universo disponível nas plataformas de streaming. Isso pode influenciar a generalização dos resultados.
- **Ausência de Dados Temporais:** As análises não consideraram variações ao longo do tempo (como lançamentos, sazonalidade ou tendências), o que limita a compreensão de mudanças no comportamento dos ouvintes ou nas estratégias de curadoria.
- **Classificação de Artistas:** A definição de artistas solo foi feita com base em um filtro textual (ausência de vírgula no nome). Isso pode ter excluído artistas solo com nomes compostos ou incluído colaborações sem curadoria manual.
- **Engajamento do Usuário Não Medido:** Métricas como número de ouvintes únicos, curtidas, tempo médio de escuta ou skip rate não foram incluídas, o que restringe a análise de engajamento real com as músicas.
- **Plataformas com Critérios Diferentes:** As diferenças de algoritmo, curadoria e público-alvo entre Spotify, Deezer e Apple Music não são completamente transparentes, o que pode afetar a comparação entre elas.
- **Causalidade Não Estabelecida:** As correlações identificadas não indicam causa e efeito. Por exemplo, estar em playlists pode aumentar os streams, mas músicas populares também têm mais chance de entrar em playlists.
- **Variável Excluída:** O Shazam foi retirado da análise por conter poucos dados relevantes, comprometendo sua contribuição estatística e interpretativa.

10. Referências:

▼ **Referências utilizadas:**

- Documentação BigQuery - Google
- Vinculação de BigQuery e Google Sheets - Canal Eliabe Silva - Youtube
- Como Criar Vizualizações no SQL - Canal Hashtag Programação - Youtube
- Concatenate - Geek for Geeks
- Matrizes no PowerBi - Canal Hashtag Treinamentos (Youtube)
- Correlação de Pearson - Psicometria OnLine
- Repositório Projeto de Análise de Dados com BigQuery - GitHub - Usuário Daniel010203
- Gemini AI.
- Chatgpt AI.
- Gamma AI (Template Apresentação).