**FICHA TÉCNICA- PROJETO 2 (HIPÓTESES)**

**Arquivo do Power BI:**

https://drive.google.com/file/d/1duRSKR8LnRIZXbvGCteUyoTEpdYfMlSO/view?usp=sharing

**Arquivo dos histogramas:**

https://docs.google.com/spreadsheets/d/1bAZmMA9pLgE1\_dVwjycnVilF7BuGzTChqC-YnSri2fY/edit?usp=sharing

**Estrutura das tabelas**

**Trackinspotify**

* 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.
* inspotifyplaylists: Número de listas de reprodução do Spotify em que a música está incluída
* inspotifycharts: 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.

#### **Trackincompetition**

* track\_id: Identificador exclusivo da música. É um número inteiro de 7 dígitos que não se repete.
* inappleplaylists: número de listas de reprodução da Apple Music em que a música está incluída.
* inapplecharts: Presença e classificação da música nas paradas da Apple Music.
* indeezerplaylists: Número de playlists do Deezer em que a música está incluída.
* indeezercharts: Presença e posição da música nas paradas da Deezer.
* inshazamcharts: Presença e classificação da música nas paradas da Shazam.

#### **Tracktechnicalinfo**

* 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.

**Caso**

Num mundo onde a indústria musical é extremamente competitiva e em constante evolução, a capacidade de tomar decisões baseadas em dados tornou-se um ativo inestimável.

Neste contexto, uma gravadora enfrenta o emocionante desafio de lançar um novo artista no cenário musical global. Felizmente, ela tem uma ferramenta poderosa em seu arsenal: um extenso conjunto de dados do Spotify com informações sobre as músicas mais ouvidas em 2023.

* A gravadora levantou uma série de hipóteses sobre o que faz uma música seja mais ouvida. Essas hipóteses incluem:
* Músicas com BPM (Batidas Por Minuto) mais altas fazem mais sucesso em termos de número de streams no Spotify.
* As músicas mais populares no ranking do Spotify também possuem um comportamento semelhante em outras plataformas, como a Deezer.
* A presença de uma música em um maior número de playlists está correlacionada com um maior número de streams.
* Artistas com um maior número de músicas no Spotify têm mais streams.
* As características da música influenciam o sucesso em termos de número de streams no Spotify.

Você deve validar (refutar ou confirmar) essas hipóteses através da análise de dados e fornecer recomendações estratégicas com base em suas descobertas. O objetivo principal desta análise é que a gravadora e o novo artista possam tomar decisões informadas que aumentem suas chances de alcançar o “sucesso”.

**1- PROCESSAR A BASE DE DADOS**

**-Conectar/importar dados para outras ferramentas**

As tabelas foram copiadas para o Google Drive, baixadas e depois carregadas no BigQuery.

**-Identificar e tratar valores nulos**

Os valores nulos foram identificados com a query:

**SELECT**

**COUNT (\*)**

**FROM**

**`projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`**

**WHERE track\_id IS NULL**

Foi feito o mesmo para todas as variáveis nas tabelas.

Tabela track\_in\_spotify:

track\_id- SEM NULOS

track\_name- SEM NULOS

artist\_s\_\_name- SEM NULOS

artist\_count- SEM NULOS

released\_year- SEM NULOS

released\_month- SEM NULOS

released\_day- SEM NULOS

in\_spotify\_playlists- SEM NULOS

in\_spotify\_charts- SEM NULOS

streams- SEM NULOS

Tabela track\_in\_competition

track\_id- SEM NULOS

in\_apple\_playlists- SEM NULOS

in\_apple\_charts- SEM NULOS

in\_deezer\_playlists- SEM NULOS

in\_deezer\_charts- SEM NULOS

in shazam charts- 50 REGISTROS NULOS

Tabela track\_technical\_info

track\_id- SEM NULOS

bpm- SEM NULOS

key- 95 REGISTROS NULOS

mode- SEM NULOS

danceability- SEM NULOS

valence- SEM NULOS

energy- SEM NULOS

acousticness- SEM NULOS

instrumentalness- SEM NULOS

liveness- SEM NULOS

speechiness- SEM NULOS

Foi verificado o número total de registros, para decidir se o número de nulos é relevante

Total de registros em track\_in\_competition: 953

Total de nulos na variável in\_shazam\_charts: 50

SELECT

COUNT (\*)

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_competition`

Como uma das hipóteses da gravadora leva em consideração a popularidade da música em outras plataformas, os nulos foram excluídos.

Total de registros em track\_technical\_info: 953

Total de nulos na variavel key: 95

SELECT

COUNT (\*)

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_technical\_info`

Como uma das hipóteses é sobre as características de cada música, não ter o tom afeta a análise. Os nulos foram excluídos.

Foram selecionados todos os registros da tabela track\_in\_competition, menos os valores nulos:

SELECT

\*

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_competition`

WHERE in\_shazam\_charts IS NOT NULL

Foram selecionados todos os registros da tabela track\_technical\_info, menos os valores nulos:

track

SELECT

\*

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_technical\_info`

WHERE key IS NOT NULL

**-Identificar e tratar valores duplicados**

Os valores duplicados foram verificados com a query:

**SELECT**

**track\_id,**

**track\_name,**

**artist\_s\_\_name,**

**COUNT(\*) AS quantidade**

**FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`**

**GROUP BY track\_id, track\_name, artist\_s\_\_name**

**HAVING COUNT (\*) > 1**

Duplicados: 0

Foram verificados os duplicados na tabela track\_in\_competition:

SELECT

track\_id,

COUNT (\*) AS quantidade

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_competition`

GROUP BY track\_id

HAVING COUNT (\*) > 1

Duplicados: 0

Foram verificados os duplicados na tabela track\_technical\_info:

SELECT

track\_id,

bpm,

key,

COUNT (\*) AS quantidade

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_technical\_info`

GROUP BY track\_id, bpm, key

HAVING COUNT (\*) > 1

Duplicados: 0

**-Identificar e tratar dados fora do escopo de análise**

Foram excluídas as variáveis key e mode, da tabela track\_technical\_info:

SELECT

\*

EXCEPT (key, mode)

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_technical\_info`

**-Identificar e tratar dados discrepantes em variáveis ​​categóricas**

Variáveis que são do tipo string

Tabela SPOTIFY

track\_id

track\_name

artist\_s\_\_name

streams

Tabela COMPETITION

track\_id

Tabela TECHNICAL

track\_id

key

mode

As variáveis do tipo string foram limpas com o comando: (ajustado para as variáveis string de cada tabela)

SELECT

track\_id,

track\_name,

artist\_s\_\_name,

streams,

REGEXP\_REPLACE(track\_id, r'[^a-zA-Z0-9]', '') AS track\_id\_clean,

REGEXP\_REPLACE(track\_name, r'[^a-zA-Z0-9]', '') AS track\_name\_clean,

REGEXP\_REPLACE(artist\_s\_\_name, r'[^a-zA-Z0-9]', '') AS artisti\_s\_\_name\_clean,

REGEXP\_REPLACE(streams, r'[^a-zA-Z0-9]', '') AS streams\_clean,

FROM

`projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`

**-Identificar e tratar dados discrepantes em variáveis ​​numéricas**

**Variáveis numéricas na tabela SPOTIFY**

artist\_count

ano/mês/dia

in\_spotify\_playlists

in\_spotify\_charts

streams

Foram identificados os valores máximo, mínimo e a média de cada variável numérica com a consulta:

SELECT

MAX (artist\_count) AS max\_artistas,

MIN (artist\_count) AS min\_artistas,

AVG (artist\_count) AS media\_artistas,

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`

Para ano/mês/dia foi usado outro comando, para que transformasse as variáveis numa data única:

SELECT

MIN(CAST(FORMAT('%i-%02i-%02i', released\_year, released\_month, released\_day) AS DATE)) AS data\_minima,

MAX(CAST(FORMAT('%i-%02i-%02i', released\_year, released\_month, released\_day) AS DATE)) AS data\_maxima

FROM

`projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`

A variável streams da tabela spotify estava como string, então foi usado outro comando, para mudar para integer e ver mínimo, máximo e média:

SELECT

MIN(SAFE\_CAST(streams AS INT64)) AS min\_streams,

MAX(SAFE\_CAST(streams AS INT64)) AS max\_streams,

AVG(SAFE\_CAST(streams AS INT64)) AS media\_streams

FROM

`projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`

**Variáveis numéricas na tabela COMPETITION**

in\_apple\_playlists

in\_apple\_charts

in\_deezer\_playlists

in\_deezer\_charts

in shazam charts

Foram identificados os valores máximo, mínimo e a média de cada variável numérica com a consulta:

SELECT

MAX (in\_apple\_playlists) AS max\_apple\_playlists,

MIN (in\_apple\_playlists) AS min\_apple\_playlists,

AVG (in\_apple\_playlists) AS media\_apple\_playlists,

MAX (in\_apple\_charts) AS max\_apple\_charts,

MIN (in\_apple\_charts) AS min\_apple\_charts,

AVG (in\_apple\_charts) AS media\_apple\_charts,

MAX (in\_deezer\_playlists) AS max\_deezer\_playlists,

MIN (in\_deezer\_playlists) AS min\_deezer\_playlists,

AVG (in\_deezer\_playlists) AS media\_deezer\_playlists,

MAX (in\_deezer\_charts) AS max\_deezer\_charts,

MIN (in\_deezer\_charts) AS min\_deezer\_charts,

AVG (in\_deezer\_charts) AS media\_deezer\_charts,

MAX (in\_shazam\_charts) AS max\_shazam\_charts,

MIN (in\_shazam\_charts) AS min\_shazam\_charts,

AVG (in\_shazam\_charts) AS media\_shazam\_charts,

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_competition`

**Variáveis numéricas na tabela TECHNICAL INFO**

BPM

danceability

valence

energy

acousticness

instrumentalness

liveness

speechiness

**-Verificar e alterar o tipo de dados**

Essa etapa já foi feita junto com a anterior, modificando a variável streams para integer.

Foi usado o comando SAFE\_CAST

SELECT

SAFE\_CAST (streams AS INT64)

AS streams\_limpo

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`

**-Criar novas variáveis ​​**

Foi criada a variável de data de lançamento usando os comandos CONCAT e CAST

SELECT

DATE (CONCAT(CAST(released\_year AS STRING),'-',CAST(released\_month AS STRING),'-', CAST(released\_day AS STRING))) AS data\_lancamento,

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`

**-Unir tabelas**

Foi criada uma consulta para cada tabela, para unir os passos de limpeza numa query só e criar uma tabela limpa

Tabela spotify:

CREATE OR REPLACE TABLE `projeto-2-hipoteses-423413.projeto2hipoteses.spotify\_limpo\_1` AS (

SELECT

REGEXP\_REPLACE(track\_id, r'[^a-zA-Z0-9]', '') AS track\_id\_limpo,

REGEXP\_REPLACE(track\_name, r'[^a-zA-Z0-9]', '') AS track\_name\_limpo,

REGEXP\_REPLACE(artist\_s\_\_name, r'[^a-zA-Z0-9]', '') AS artisti\_s\_\_name\_limpo,

artist\_count,

DATE (CONCAT(CAST(released\_year AS STRING),'-',CAST(released\_month AS STRING),'-', CAST(released\_day AS STRING))) AS data\_lancamento,

in\_spotify\_playlists, in\_spotify\_charts,

IF(REGEXP\_REPLACE(streams, r'[^a-zA-Z0-9]', '') = 'BPM110KeyAModeMajorDanceability53Valence75Energy69Acousticness7Instrumentalness0Liveness17Speechiness3', 0, CAST(REGEXP\_REPLACE(streams, r'[^a-zA-Z0-9]', '') AS INT64)) AS streams\_limpo,

FROM

`projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_spotify`

WHERE REGEXP\_REPLACE(track\_id, r'[^a-zA-Z0-9]', '') <> '000'

ORDER BY track\_id ASC)

Tabela Competition:

CREATE OR REPLACE TABLE `projeto-2-hipoteses-423413.projeto2hipoteses.competition\_limpo\_1` AS (

SELECT

REGEXP\_REPLACE(track\_id, r'[^a-zA-Z0-9]', '') AS track\_id\_clean,

in\_apple\_playlists, in\_apple\_charts, in\_deezer\_playlists, in\_deezer\_charts,

IFNULL(CAST(in\_shazam\_charts AS STRING), 'não identificado') AS in\_shazam\_charts\_sem\_nulos

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_in\_competition`

WHERE track\_id <> '0:00'

ORDER BY REGEXP\_REPLACE(track\_id, r'[^a-zA-Z0-9]', '') ASC

)

Tabela Technical Info:

CREATE OR REPLACE TABLE `projeto-2-hipoteses-423413.projeto2hipoteses.technical\_info\_limpo\_1` AS (

SELECT

REGEXP\_REPLACE(track\_id, r'[^a-zA-Z0-9]', '') AS track\_id\_clean,

bpm, danceability\_\_, valence\_\_, energy\_\_, acousticness\_\_, instrumentalness\_\_, liveness\_\_, speechiness\_\_

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.track\_technical\_info`

WHERE REGEXP\_REPLACE(track\_id, r'[^a-zA-Z0-9]', '') <> '000'

ORDER BY REGEXP\_REPLACE(track\_id, r'[^a-zA-Z0-9]', '') ASC

)

Posteriormente a tabela Spotify foi unida à tabela Competition:

CREATE OR REPLACE TABLE `projeto-2-hipoteses-423413.projeto2hipoteses.spotifyecompetition` AS (

SELECT

track\_id\_limpo, track\_name\_limpo, artisti\_s\_\_name\_limpo, artist\_count, data\_lancamento, in\_spotify\_playlists, in\_spotify\_charts, streams\_limpo,

in\_apple\_playlists, in\_apple\_charts, in\_deezer\_playlists, in\_deezer\_charts, in\_shazam\_charts\_sem\_nulos,

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.spotify\_limpo\_1` LEFT

OUTER JOIN `projeto-2-hipoteses-423413.projeto2hipoteses.competition\_limpo\_1`

ON track\_id\_limpo = track\_id\_clean)

Após a união das duas primeiras tabelas, foi adicionada a tabela Technical Info:

SELECT

track\_id\_limpo, track\_name\_limpo, artisti\_s\_\_name\_limpo, artist\_count, data\_lancamento, in\_spotify\_playlists, in\_spotify\_charts, streams\_limpo,

in\_apple\_playlists, in\_apple\_charts, in\_deezer\_playlists, in\_deezer\_charts, in\_shazam\_charts\_sem\_nulos, bpm, danceability\_\_, valence\_\_, energy\_\_, acousticness\_\_, instrumentalness\_\_, liveness\_\_, speechiness\_\_

FROM `projeto-2-hipoteses-423413.projeto2hipoteses.spotifyecompetition` LEFT

OUTER JOIN `projeto-2-hipoteses-423413.projeto2hipoteses.technical\_info\_limpo\_1`

ON track\_id\_limpo = track\_id\_clean

**-Construir tabelas auxiliares**

O total de músicas de artistas solo foi calculado com a query:

**WITH tabela\_temporaria AS (**

**SELECT**

**artisti\_s\_\_name\_limpo,**

**COUNT (\*)**

**AS numero\_musicas\_solo**

**FROM `projeto-2-hipoteses-423413.projeto2hipoteses.tabelas\_unidas`**

**GROUP BY artisti\_s\_\_name\_limpo )**

**SELECT**

**SUM(numero\_musicas\_solo) AS total\_musicas\_solo**

**FROM tabela\_temporaria**

Posteriormente foi verificado que um valor da variável streams (na linha do id 4061483) ficou com caracteres errados que foram corrigidos usando o código:

**UPDATE projeto-2-hipoteses-423413.projeto2hipoteses.tabelas\_unidas**

**SET streams\_limpo = 'não identificado'**

**WHERE streams\_limpo = 'BPM110KeyAModeMajorDanceability53Valence75Energy69Acousticness7Instrumentalness0Liveness17Speechiness3'**

**2- FAZER UMA ANÁLISE EXPLORATÓRIA**

**-Agrupar dados de acordo com variáveis categóricas**

Etapa realizada no Power BI

**-Ver variáveis ​​categóricas**

Etapa realizada no Power BI

**-Aplicar medidas de tendência central**

Etapa realizada no Power BI

**-Ver distribuição**

Os histogramas foram feitos no google sheets.

**-Aplicar medidas de dispersão**

Etapa realizada no Power BI

**-Visualizar o comportamento dos dados ao longo do tempo**

Etapa realizada no Power BI

**-Calcular quartis, decis ou percentis**

Os quartis foram calculados e foram criadas categorias para cada variável de característica das músicas com o comando (ajustado para cada variável):

**WITH Quartil AS (**

**SELECT**

**track\_id\_limpo,**

**bpm,**

**NTILE(4) OVER (ORDER BY track\_id\_limpo ASC) AS quartil\_bpm**

**FROM `projeto-2-hipoteses-423413.projeto2hipoteses.tabelas\_unidas`**

**)**

**SELECT**

**a.\*,**

**b.quartil\_bpm,**

**IF(b.quartil\_bpm = 1, 'baixo',**

**IF(b.quartil\_bpm = 2, 'medio\_baixo',**

**IF(b.quartil\_bpm = 3, 'medio\_alto', 'alto'))) AS categoria\_bpm**

**FROM `projeto-2-hipoteses-423413.projeto2hipoteses.tabelas\_unidas` a**

**LEFT JOIN Quartil b**

**ON a.track\_id\_limpo = b.track\_id\_limpo;**

Foram unidas uma coluna para o quartil e uma coluna para a categoria de cada variável à tabela completa com o comando CREATE OR REPLACE TABLE.

**-Calcular correlação entre variáveis ​​**

Comando (ajustado para cada par de variáveis):

SELECT CORR (streams\_limpo, total\_playlists) AS correlation\_value

FROM projeto-2-hipoteses-423413.projeto2hipoteses.tabelas\_unidas

Resultados:

Spotify playlists e Apple playlists- 0.7090714372622

Spotify playlists e Deezer playlists- 0.709071437262

Streams e Spotify playlists- 0.78976021815974

Streams e Apple Playlists- 0.77350566354905

Streams e Deezer Playlists- 0.5980999712634

Streams e Total de playlists- 0.782983734858

Streams e Total de músicas- 0.780930042883

Streams e BPM- 0.5980999712634

Streams e danceability- -0.10438719371078

Streams e valence- -0.0414887245770

Streams e energy- -0.0260130078016

Streams e acousticness- -0.00401675105

Streams e instrumentalness- -0.044772094681

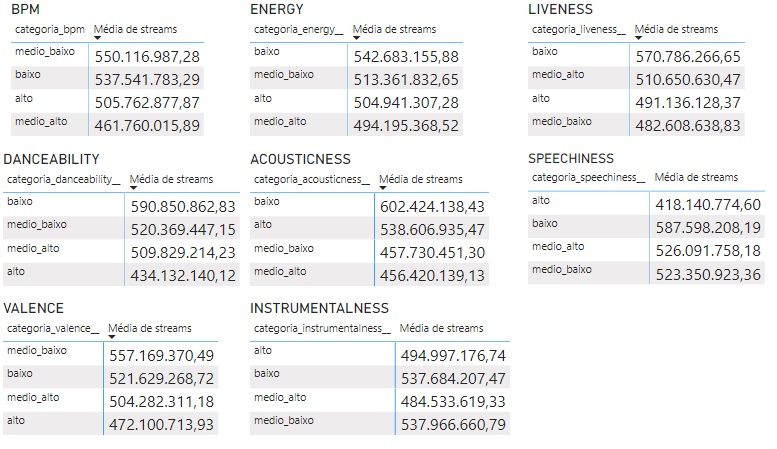
Streams e liveness- -0.04824014673

Streams e speechness- -0.11157957701

**3-APLICAR TÉCNICA DE ANÁLISE**

**-Aplicar segmentação**

Etapa realizada no Power BI

****

**-Validar hipótese**

Primeiro foi criada uma tabela para agrupar o total de músicas por artista:

**CREATE OR REPLACE TABLE projeto-2-hipoteses-423413.projeto2hipoteses.musicas\_artistas AS**

**WITH musicas\_artistas AS (**

**SELECT**

**artisti\_s\_\_name\_limpo,**

**COUNT(artisti\_s\_\_name\_limpo) AS total\_musicas,**

**SUM(streams\_limpo) AS total\_streams**

**FROM**

**projeto-2-hipoteses-423413.projeto2hipoteses.tabelas\_unidas**

**GROUP BY**

**artisti\_s\_\_name\_limpo**

**)**

**SELECT**

**artisti\_s\_\_name\_limpo,**

**total\_musicas,**

**total\_streams**

**FROM**

**musicas\_artistas**

**ORDER BY total\_streams DESC**

Foi feito um gráfico de dispersão para cada hipótese e também comparado ao valor de correlação.

Etapa realizada no Power BI.

### **4- RESUMIR INFORMAÇÕES EM UM DASHBOARD OU RELATÓRIO**

Etapa realizada no Power BI

**5- APRESENTAR RESULTADOS**

Link da apresentação:

<https://docs.google.com/presentation/d/1Viy8BMpuErmIY4jvmM1Hs-YrgVxwkBfXuqVoc1gvI0w/edit?usp=sharing>