

Workshop 02 - ETL

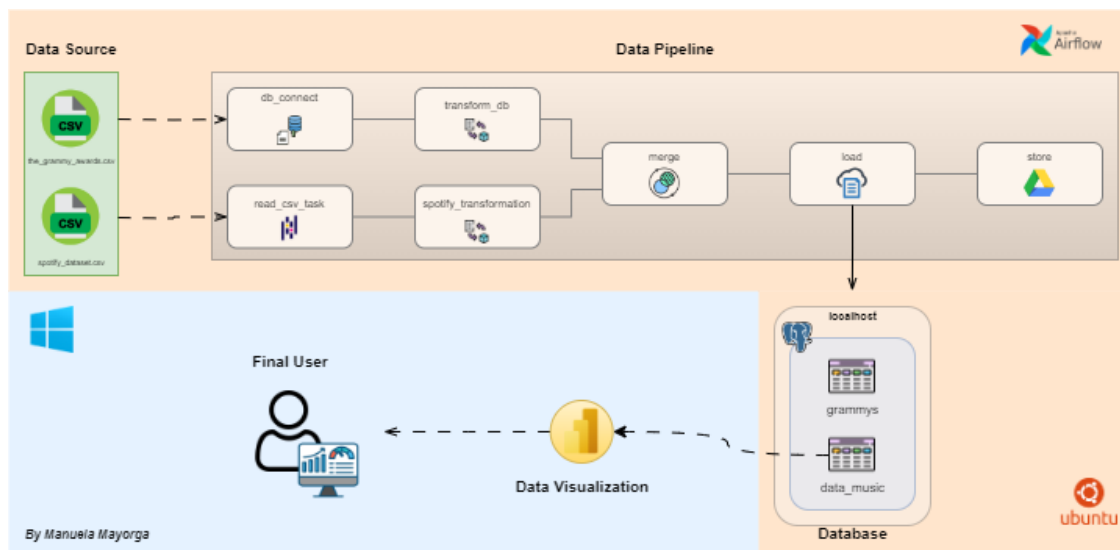
Manuela Mayorga Rojas
Universidad Autónoma de Occidente
Abril, 2024

Introduction

This workshop is about how to build an ETL pipeline using Apache Airflow, this will be done using 2 data sources (csv file, database), the first data set is from Spotify (csv file) and the second set is from Grammys (database), using apache airflow we will read the data, one as a csv and another using SQLAlchemy database engine, also using Airflow we will perform transformations, then, merge both data sets and load them in google drive, finally visualize this information in PowerBI.

Objectives

- Use Apache Airflow to read data from multiple sources, such as CSV files and databases.
- Apply transformations to the data read using Apache Airflow.
- Upload the transformed data to an external storage platform (Google Drive).
- Merge data sets.
- Use PowerBI to visualize information.



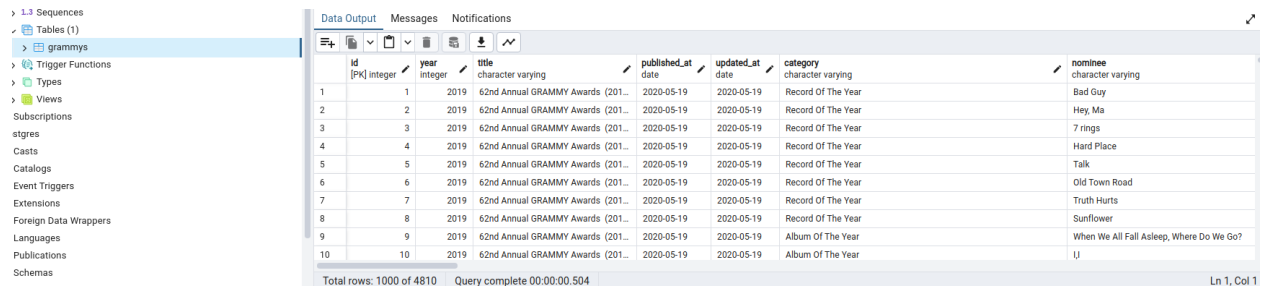
Let's start with the workshop

The first thing that was done was an exploratory analysis of the data for each set, in order to identify the type of data and all the characteristics they have. In this way it will be possible to identify which transformations are necessary to perform in a next process.

Exploratory Data Analysis

The Grammy Awards Dataset

Before starting with this EDA the first thing that was done was the loading and insertion of the data to Postgres by means of SQLAlchemy, in the code is the way in which the function for this process was created, but the most important thing for this is that the database has to be previously created, to only make the connection to Postgres, create the table called 'grammys' and upload the data. Here is the evidence of this process:



| | id | year | title | published_at | updated_at | category | nominee |
|----|----|------|-----------------------------------|--------------|------------|--------------------|--|
| 1 | 1 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Bad Guy |
| 2 | 2 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Hey, Ma |
| 3 | 3 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | 7 rings |
| 4 | 4 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Hard Place |
| 5 | 5 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Talk |
| 6 | 6 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Old Town Road |
| 7 | 7 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Truth Hurts |
| 8 | 8 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Sunflower |
| 9 | 9 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Album Of The Year | When We All Fall Asleep, Where Do We Go? |
| 10 | 10 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Album Of The Year | J |

Total rows: 1000 of 4810 Query complete 00:00:00.504 Ln 1, Col 1

Understanding the data

We start by making an observation of the data we have and we have a description of the columns:

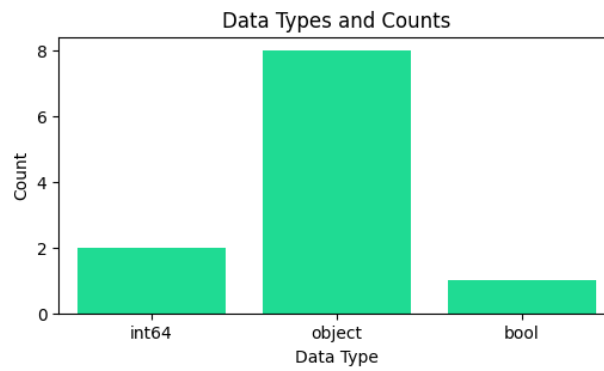
Description of columns

- **id**: Unique identifier for each entry in the dataset. This is not included in the original dataset, but was added as an ID at the time of loading into the database.
- **year**: The year in which the Grammy Awards ceremony took place.
- **title**: Title of the Grammy Awards ceremony, including the year.
- **published_at**: Date and time the record was published.
- **updated_at**: Date and year the record was updated.
- **category**: The category in which the nomination falls.
- **nominee**: Song, Album or Artist nominated.
- **artist**: The name of the artist associated with the nomination.
- **workers**: People involved in the production of the nominated song or album, such as producers, sound engineers, etc.
- **img**: URL of the image associated with the nomination.
- **winner**: Boolean indicator that shows whether the nomination was a winner (*True*) or not (*False*).

Then we did a count to see how many rows and columns we have, we realize that we have 4810 rows and 11 columns. Next we do a data dictionary to see what type of data and additional information:

| Column Name | Data Type | Null Values | Unique Values |
|--------------|-----------|-------------|---------------|
| id | int64 | 0 | 4810 |
| year | int64 | 0 | 62 |
| title | object | 0 | 62 |
| published_at | object | 0 | 4 |
| updated_at | object | 0 | 10 |
| category | object | 0 | 638 |
| nominee | object | 6 | 4131 |
| artist | object | 1840 | 1658 |
| workers | object | 2190 | 2366 |
| img | object | 1367 | 1463 |
| winner | bool | 0 | 1 |

Bar chart data types



According to this diagram the types of each column with raw data are as follows:

- **integers:** 2
- **object:** 8.
- **boolean:** 1

The process of eliminating the columns 'published_at', 'updated_at' and 'img' columns that we are not going to use was carried out because by eliminating these columns, the idea is to simplify the dataset, eliminating redundant or non-relevant information for the analysis, which can speed up the analysis process and reduce the complexity of your models.

Descriptive analysis

In this case we performed descriptive analysis for both quantitative and qualitative columns.

Quantitative analysis

| | Count | Mean | Std | Min | 25% | 50% | 75% | Max |
|------|--------|-------------|-------------|--------|---------|--------|---------|--------|
| id | 4810.0 | 2405.500000 | 1388.671727 | 1.0 | 1203.25 | 2405.5 | 3607.75 | 4810.0 |
| year | 4810.0 | 1995.566944 | 17.149720 | 1958.0 | 1983.00 | 1998.0 | 2010.00 | 2019.0 |

The results are interpreted for the year column since the id cannot give us much information.

- The year data range from 1958 to 2019, with a mean of approximately 1995.80. This suggests that the data are centered around the 1990s and early 2000s.

- The standard deviation is relatively low (17.05), indicating that the values tend to be closer to the mean.
- The median is close to the mean, suggesting a symmetrical distribution of the data.

Qualitative analysis

| | Count | Unique | Top | Freq |
|----------|-------|--------|---|------|
| title | 4810 | 62 | 62nd Annual GRAMMY Awards (2019) | 433 |
| category | 4810 | 638 | Song Of The Year | 70 |
| nominee | 4804 | 4131 | Berlioz: Requiem | 7 |
| artist | 2970 | 1658 | (Various Artists) | 66 |
| workers | 2620 | 2366 | John Williams, composer (John Williams) | 20 |

These results indicate the distribution and frequency of the data:

- title: There are 4810 titles in total. The most common title is "62nd Annual GRAMMY Awards (2019)" and appears 433 times.
- category: There are 4810 unique entries in the category. "Song Of The Year" is the most common category, with 70 occurrences.
- nominee: There are 4804 unique nominees in total. "Berlioz: Requiem" is the most common nominee, with 7 occurrences.
- artist: 2970 unique artists are found in total. "(Various Artists)" is the most common artist, with 66 occurrences.
- workers: There are 2620 unique workers in total. "John Williams, composer (John Williams)" is the most common worker, with 20 occurrences.

Then we check for duplicate values and in this case there are none.

Null values

Previously we noticed that we have null values in 3 columns, so we decided to take a closer look at what is in each column. For the 'nominee' column we decided to eliminate these null values because there was really no way to do anything with them and make their information valuable. Continuing with the artist and workers columns after observing in detail their information we realized that in the data there is a pattern, where some columns have information that we assume and assume that it is about the artist located in parentheses, so this could be a way to complete the null values in the artist column, and for the case of the nulls in the workers column, the information that serves to fill in the nulls is the artist, since we assume that these columns are related and thus find a way for one column to complement the other and not simply eliminate all the null values.

Then to carry out this process we are going to use regular expressions to analyze and manipulate the text patterns, in this case for the process that the artist column is completed with the values of workers and then that the workers column is completed with the artist column.

And finally these were the remaining nulls after the process performed:

| Variable | Missing Values |
|----------|----------------|
| id | 0 |
| year | 0 |
| title | 0 |
| category | 0 |
| nominee | 0 |
| artist | 468 |
| workers | 180 |
| winner | 0 |

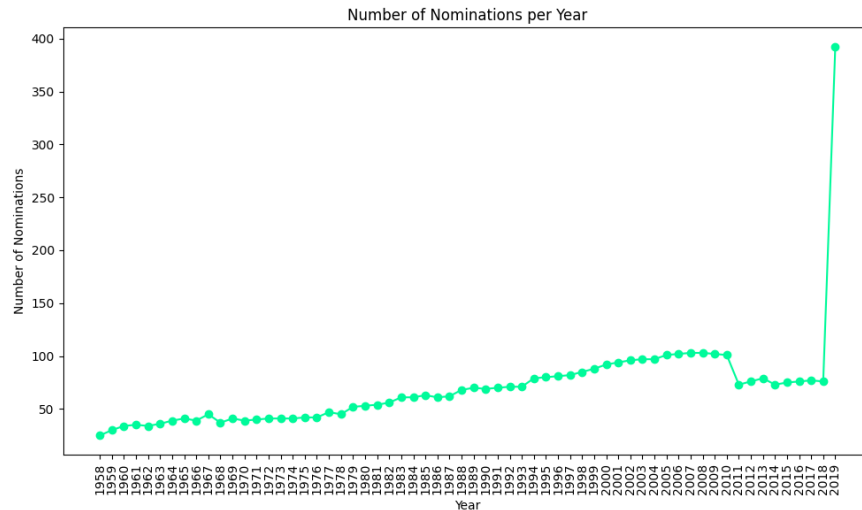
This remaining null data ends up being eliminated as it does not contribute any information of value to the process and there is no longer anything to do with it. And then we perform a data dictionary again to verify the new data information.

| Column Name | Null Values | Unique Values |
|-------------|-------------|---------------|
| id | 0 | 4336 |
| year | 0 | 62 |
| title | 0 | 62 |
| category | 0 | 569 |
| nominee | 0 | 3773 |
| artist | 0 | 2510 |
| workers | 0 | 3371 |
| winner | 0 | 1 |

Column Year

To begin the analysis of this column, what was done was to identify from which year there is information in the dataset, and we realized that it goes from 1958 to 2019.

Analysis of nominations by year:



- The number of nominations increased between 1958 and 2019.
- The general trend is an increase in the number of nominations.

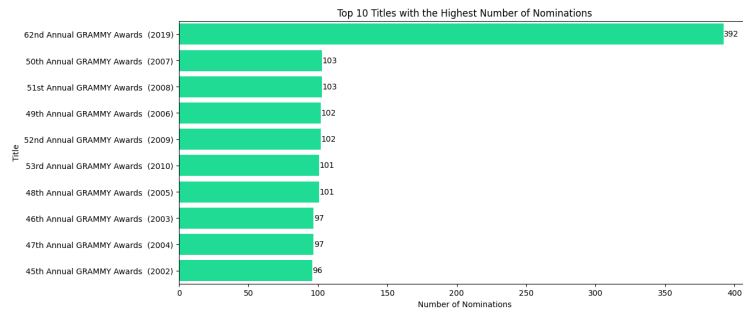
It is clear that the graph shows an increasing trend but has a decrease between 2010 and 2018, but what is most significant is the large increase in 2019, which could be due to the fact that music has become more diverse in recent decades and has also experienced significant growth in recent decades. This has led to a

greater number of musical genres and styles represented at the Grammys, which in turn has increased the number of nominations.

Column title

For this column we identified the unique values which are 62, where we realized that the title consists of the name of the event plus the year in parentheses.

The top 10 titles with the most nominations were observed:

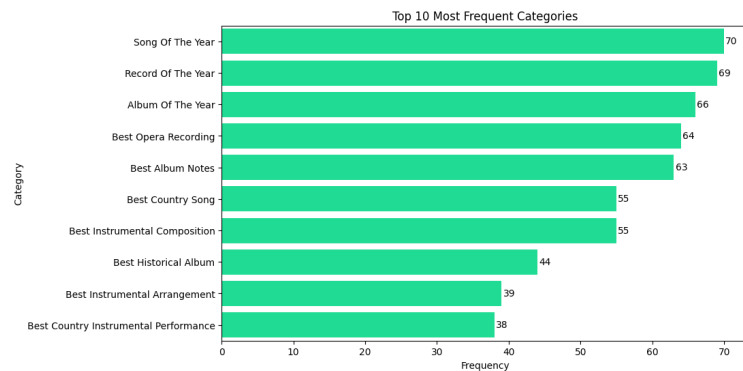


It is evident that the title 62nd which corresponds to the year 2019 has a fairly large increase in nominations compared to other years, then I searched the internet the 2019 grammys and see if it is possible to find a reason why it is the title with more nominations, and I found that the number of the title is wrong and the 2019 grammys are the number 61st and the title 62nd corresponds to 2020 according to what I found. Then I looked for different titles and I realized that the year is wrong, but I looked in the official page of the grammys and there the information was correct, that is to say, it coincides exactly as in the dataset. Then I realized some very important information the 1st and 2nd Grammy Awards ceremony were held in the same year in 1959 "The first thing you should know about the 2nd Annual GRAMMY Awards is that they weren't actually "annual" at all. In fact, this awards presentation marked the only time in GRAMMY history that two awards presentations were ever made in one year, with both the 1st and 2nd GRAMMYS falling in 1959. Call it a slightly embarrassing case of premature validation" (Grammy Awards, n.d.).

But finally I did not find the reason for this increase, so we assume that it is due to the popularity that has been reaching the different musical styles, involving different cultures, rhythms and lyrics, or that there are many artists who have reached a high level of popularity and this makes more nominations are generated.

Column category

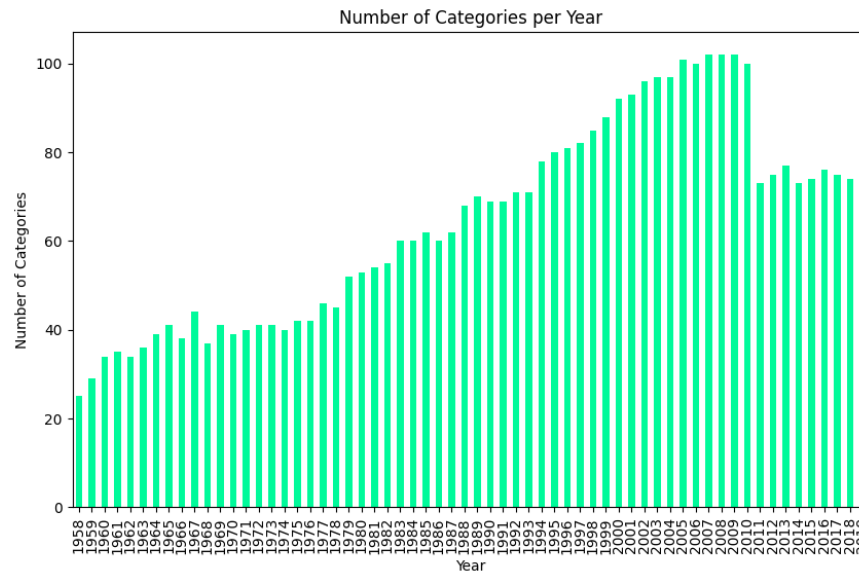
Initially we see that we have 569 different categories, so let's see which are the most frequent:



The most frequent categories at the Grammy Awards are those that encompass popular genres, are considered the most important and have had a significant cultural impact. On the other hand, other categories with

high frequency focus on specific genres or aspects of music, rewarding quality in particular areas of music production.

We then looked at the number of categories per year:



The increase in the number of categories at the Grammy Awards is due to several key factors. It reflects the increasing diversification of popular music, adapting to new genres and emerging subgenres, and is related to the expansion of the music market, which allows more artists to be honored and attract a wider audience. And about the minimization of categories between 2010 and 2019, maybe it is due to internal processes of the organization.

Column nominee

For this column we identified what information is inside to realize that it contains the names of artists, songs, among others that have been nominated for the grammys. Then we see what is the top 10 most nominated artists to the grammys:

| Song | Number of Nominations |
|-----------------------------|-----------------------|
| Bridge Over Troubled Water | 7 |
| Berlioz: Requiem | 6 |
| Up, Up And Away | 6 |
| Gentle On My Mind | 5 |
| Need You Now | 5 |
| A Taste Of Honey | 5 |
| King Of The Road | 5 |
| West Side Story | 5 |
| Blackstar | 5 |
| Mahler: Symphony No. 9 In D | 4 |

These songs could have had a great impact or are very good and popular songs, due to their musical composition, lyrics, vocal performance and production.

Column artist

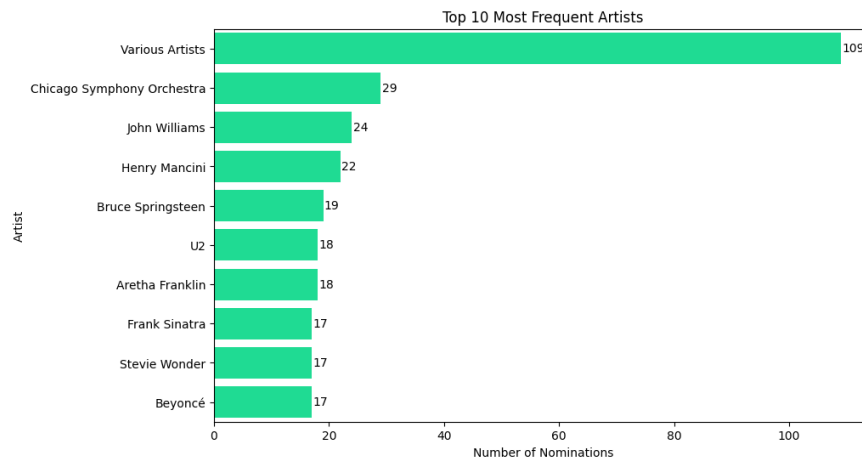
We explore what information makes up this column and identify that there are artist names in parentheses, this is due to the process that was done with the null values. Then what we did was to test with any artist

what information is in the rows that have the name in parentheses and those that do not.

And then we saw that they are stored as if they were different artists and this could harm the analysis. So we will do the procedure to remove this parenthesis using regular expressions.

Then we verified that there are no artists in parentheses and this was done by checking with the previous artist and we see that the process worked. (the visualization of this process is in the code in the notebook 002)

So here we verify that we went from having 2510 unique values to having 2242. In addition we observe the top 10 most frequent artists:

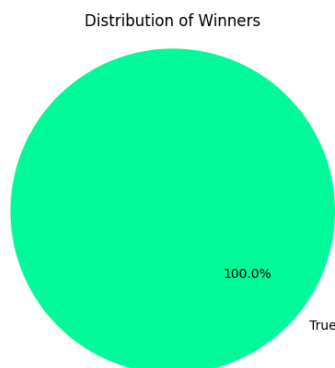


The chart provides valuable information about the artists who have had the most success at these awards or those who have had a significant impact on the music industry throughout their careers. It also reflects the diversity of musical genres that are recognized at the Grammys.

We check which rows correspond to 'Various Artists' and we realize that these correspond to the data that were completed in the null values process, as we can see that they correspond to the information in the winners column

Column winner

We made a pie chart to analyze the distribution of the data in this column.



This graph shows that all nominees were winners, I think it would make much more sense to rename this column nominee, this with the objective of the work to be done in the future and rename this column would allow a better result.

Spotify Dataset

Understanding the data

We read this dataset as csv, and we start by making an observation of the data we have and we have a description of the columns:

Description of columns

- **Unnamed: 0:** This column appears to have been created automatically when downloading the data, as the original set lacked this column. It functions as a default sequential ID, providing a unique identifier for each entry in the data set.
- **track_id:** Spotify ID of the track in the dataset.
- **artists:** The names of the artists who performed the track. If there is more than one artist, they are separated by a ; (semicolons).
- **album_name:** Name of the album the track is included in.
- **track_name:** Name of the track.
- **popularity:** Numerical value between 0 and 100 indicating the popularity of a track, with 100 being the highest value.
- **duration_ms:** Track duration in milliseconds.
- **explicit:** If the hint has explicit letters or not (true = yes, false = no, no OR unknown)
- **danceability:** describes how suitable a track is for dancing, a value of 0.0 is the least danceable and 1.0 is the most danceable.
- **energy:** Energy is a measure of 0.0 to 1.0 and represents a perceptual measure of intensity and activity.
- **key:** The key in which the track is located. Integers are assigned to tones using standard tone class notation. For example 0 = C, 1 = C \sharp /D \flat , 2 = D, etc. If no key was detected, the value is -1.
- **loudness:** The overall loudness of a track in decibels (dB).
- **mode:** Mode indicates the mode (major or minor) of a track, the scale from which its melodic content is derived. Major is represented by 1 and minor is 0.
- **speechiness:** detects the presence of spoken words in a track. Values below 0.33 are likely to be non-speech tracks. Values between 0.33 and 0.66 describe tracks that may contain both music and speech. Values above 0.66 describe tracks that are likely to be composed of spoken words only.
- **acousticness:** A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence that the track is acoustic.
- **instrumentalness:** The closer the instrumentalness value is to 1.0, the higher the probability that the track contains no vocal content.
- **liveness:** Detects the presence of an audience in the recording. A high value means a higher probability that the track was played live. Values range from 0.0 - 1.0
- **valence:** A measure from 0.0 to 1.0 that describes the musical positivity transmitted by a track.
- **tempo:** The estimated overall tempo of a track in beats per minute (BPM).
- **time_signature:** Is an estimated time signature. The time signature varies from 3 to 7, which indicates time signatures from 3/4 to 7/4.
- **track_genre:** The genre to which the track belongs.

Then we did a count to see how many rows and columns we have, we realize that we have 114000 rows and 21 columns. Next we do a data dictionary to see what type of data and additional information:

| Column Name | Data Type | Null Values | Unique Values |
|------------------|-----------|-------------|---------------|
| Unnamed: 0 | int64 | 0 | 114000 |
| track_id | object | 0 | 89741 |
| artists | object | 1 | 31437 |
| album_name | object | 1 | 46589 |
| track_name | object | 1 | 73608 |
| popularity | int64 | 0 | 101 |
| duration_ms | int64 | 0 | 50697 |
| explicit | bool | 0 | 2 |
| danceability | float64 | 0 | 1174 |
| energy | float64 | 0 | 2083 |
| key | int64 | 0 | 12 |
| loudness | float64 | 0 | 19480 |
| mode | int64 | 0 | 2 |
| speechiness | float64 | 0 | 1489 |
| acousticness | float64 | 0 | 5061 |
| instrumentalness | float64 | 0 | 5346 |
| liveness | float64 | 0 | 1722 |
| valence | float64 | 0 | 1790 |
| tempo | float64 | 0 | 45653 |
| time_signature | int64 | 0 | 5 |
| track_genre | object | 0 | 114 |

Here we can see that we have several types of data and also 3 columns with a null value



According to this diagram the types of each column with raw data are as follows:

- integers: 6
- object: 5
- boolean: 1
- float: 9

Delete column that will not be used

In this case the Unnamed : 0 column was removed, which according to the description is the id that was created when loading the data, but this ID column is not needed. In addition, when checking for duplicate values and this column was present, no duplicate values appeared, but when it was eliminated, they were identifiable.

Descriptive analysis

Quantitative analysis

| Column | Count | Mean | Std | Min | 25% | 50% | 75% | Max |
|------------------|----------|---------------|---------------|---------|------------|------------|------------|-------------|
| popularity | 114000.0 | 33.238535 | 22.305078 | 0.000 | 17.000 | 35.000 | 50.000 | 100.000 |
| duration_ms | 114000.0 | 228029.153114 | 107297.712645 | 0.000 | 174066.000 | 212906.000 | 261506.000 | 5237295.000 |
| danceability | 114000.0 | 0.566800 | 0.173542 | 0.000 | 0.456 | 0.580 | 0.695 | 0.985 |
| energy | 114000.0 | 0.641383 | 0.251529 | 0.000 | 0.472 | 0.685 | 0.854 | 1.000 |
| key | 114000.0 | 5.309140 | 3.559987 | 0.000 | 2.000 | 5.000 | 8.000 | 11.000 |
| loudness | 114000.0 | -8.258960 | 5.029337 | -49.531 | -10.013 | -7.004 | -5.003 | 4.532 |
| mode | 114000.0 | 0.637553 | 0.480709 | 0.000 | 0.000 | 1.000 | 1.000 | 1.000 |
| speechiness | 114000.0 | 0.084652 | 0.105732 | 0.000 | 0.03590 | 0.048900 | 0.0845 | 0.965 |
| acousticness | 114000.0 | 0.314910 | 0.332523 | 0.000 | 0.01690 | 0.169000 | 0.5980 | 0.996 |
| instrumentalness | 114000.0 | 0.156050 | 0.309555 | 0.000 | 0.00000 | 0.000042 | 0.0490 | 1.000 |
| liveness | 114000.0 | 0.213553 | 0.190378 | 0.000 | 0.09800 | 0.132000 | 0.2730 | 1.000 |
| valence | 114000.0 | 0.474068 | 0.259261 | 0.000 | 0.26000 | 0.464000 | 0.6830 | 0.995 |
| tempo | 114000.0 | 122.147837 | 29.978197 | 0.000 | 99.21875 | 122.017000 | 140.0710 | 243.372 |
| time_signature | 114000.0 | 3.904035 | 0.432621 | 0.000 | 4.000 | 4.000 | 4.000 | 5.000 |

We can tell a few things from this:

- Initially we observe that all columns have a count of 114000 since that is the amount of data.
- Popularity has a range from 0 to 100. The mean popularity is about 33.24, suggesting that the songs have moderate popularity overall.
- The average song duration is about 228,030 milliseconds (approximately 3 minutes and 48 seconds), and has a fairly high standard deviation, indicating a wide variability in song duration.
- Musical characteristics such as danceability, energy, and valence have moderate means. This suggests that the songs in general have a medium level of these characteristics, making them suitable for dancing and potentially enjoyable in terms of energy and emotion.
- The mean values and standard deviations for Loudness, Speechiness, Acousticness vary, indicating a diversity in musical content.

Qualitative analysis

| Variable | Count | Unique | Top | Freq |
|-------------|--------|--------|----------------------------|------|
| track_id | 114000 | 89741 | 6S3JIDAGk3uu3NtZbPnuhS | 9 |
| artists | 113999 | 31437 | The Beatles | 279 |
| album_name | 113999 | 46589 | Alternative Christmas 2022 | 195 |
| track_name | 113999 | 73608 | Run Rudolph Run | 151 |
| track_genre | 114000 | 114 | acoustic | 1000 |

We note that:

- track_id: There are a total of 89,740 unique values. The identifier "6S3JIDAGk3uu3NtZbPnuhS" appears most frequently, 9 times.
- artists: There are a total of 31,437 unique values. "The Beatles" is the most common artist, appearing 279 times.
- album_name: There are a total of 46,589 unique values. "Alternative Christmas 2022" is the most common album name, appearing 195 times.
- track_name: There are a total of 73,608 unique values. "Run Rudolph Run" is the most common track name, appearing 151 times.

- track_genre: There are 114 unique genres. "Acoustic" is the most common genre, appearing 1000 times.

Null values

As we saw in the data dictionary we have a null value corresponding to the track_id = "1kR4gIb7nGxHPI3D2ifs59", this row corresponding to this track id will be deleted because it has null values in the artists, album_name and track_name, so it has no information of value to contribute to the analysis.

Check data

Now we redo a data dictionary to verify that there are no null values:

| Column Name | Data Type | Null Values | Unique Values |
|------------------|-----------|-------------|---------------|
| track_id | object | 0 | 89740 |
| artists | object | 0 | 31437 |
| album_name | object | 0 | 46589 |
| track_name | object | 0 | 73608 |
| popularity | int64 | 0 | 101 |
| duration_ms | int64 | 0 | 50696 |
| explicit | bool | 0 | 2 |
| danceability | float64 | 0 | 1174 |
| energy | float64 | 0 | 2083 |
| key | int64 | 0 | 12 |
| loudness | float64 | 0 | 19480 |
| mode | int64 | 0 | 2 |
| speechiness | float64 | 0 | 1489 |
| acousticness | float64 | 0 | 5061 |
| instrumentalness | float64 | 0 | 5346 |
| liveness | float64 | 0 | 1722 |
| valence | float64 | 0 | 1790 |
| tempo | float64 | 0 | 45652 |
| time_signature | int64 | 0 | 5 |
| track_genre | object | 0 | 114 |

Check duplicate values

As mentioned above, it was important to remove the Unnamed column in order to see the duplicates. When verifying the duplicates we see that there are 894 but reviewing them we see that there are track_id that have the same information but the only thing that changes is the track_genre, then verifying again we see that in total there are 450 duplicates totally the same even in the track_genre, then the decision was taken to eliminate these duplicates since they do not contribute anything.

After eliminating duplicate values we have 113549 rows and 20 columns (due to the elimination of Unnamed).

I wanted to verify how many value counts there are in the track_id and there is one that has 9, I checked if maybe there was an error when eliminating the nulls but what happens is that as mentioned above the only thing that changes is the track_genre, that is, there are songs that have several genres, later there will be a due procedure with the genres.

Column artists

Now let's explore the artists column, where we see the number of artists and the 10 most frequent artists, along with the number of occurrences of

| Artists | Count |
|-----------------|-------|
| The Beatles | 279 |
| George Jones | 260 |
| Stevie Wonder | 235 |
| Linkin Park | 224 |
| Ella Fitzgerald | 221 |
| Prateek Kuhad | 217 |
| Feid | 201 |
| Chuck Berry | 190 |
| Håkan Hellström | 183 |
| OneRepublic | 181 |

One reason why artists are so frequent is because these artists have many songs, many albums or their songs cover many genres and this causes them to generate a very high level of frequencies.

Column album_name

For this column we are going to identify the 10 albums with the highest number of occurrences in the data set

| Album Name | Count |
|-----------------------------|-------|
| Alternative Christmas 2022 | 195 |
| Feliz Cumpleaños con Perreo | 180 |
| Metal | 143 |
| Halloween con perreito | 122 |
| Halloween Party 2022 | 114 |
| The Complete Hank Williams | 110 |
| Fiesta portatil | 108 |
| Frescura y Perreo | 106 |
| Esto me suena a Farra | 105 |
| On air 70's Hits | 101 |

We can see that the dataset seems to have a variety of albums, including theme music, popular genres and compilations of specific artists, also we could say that they have so many repetitions because the albums can have many songs and then this would make this record repeat and that is why it has so many numbers of occurrences.

And additionally I wanted to check what records are inside 'Alternative Christmas 2022' and we can see that the dataset seems to have a variety of albums, including theme music, popular genres and compilations of specific artists, also we could say that they have so many repetitions because the albums can have many songs and then this would make this record repeat and that is why it has so many numbers of occurrences.

Column track_name

We continue to observe which is the top 10 of the most repeated song accompanied by the number of occurrences

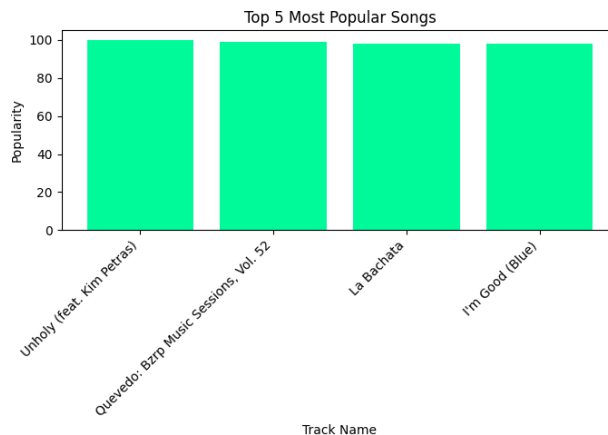
| Track Name | Count |
|--------------------------------|-------|
| Run Rudolph Run | 151 |
| Halloween | 88 |
| Frosty The Snowman | 80 |
| Little Saint Nick - 1991 Remix | 74 |
| Christmas Time | 72 |
| Last Last | 70 |
| CÓMO SE SIENTE - Remix | 64 |
| Sleigh Ride | 61 |
| RUMBATÓN | 57 |
| X ÚLTIMA VEZ | 57 |

As it is evident there is a song that is repeated a total of 151 times, then we will observe why, and we wanted to evaluate if maybe that song had that much frequency because it may have the same name but have a different artist, so we checked that but apparently the song is by the artist 'Chuck Berry'.

The song belongs to only one artist but belongs to different albums. So it is possible that several songs belong to the same artist but different albums or simply have the same song name and different artist. In this case the song is from the same artist

Column popularity

We start by evaluating the range in this column from 0 to 100. We look at the top 5 most popular songs:



We note that in the graph there are only 4 songs and they are very even, so it is better to look at this information in another way. The reason why there are only 4 songs is because the first two songs are the same, but they have different genres and that's why it comes out 2 times.

Column duration_ms

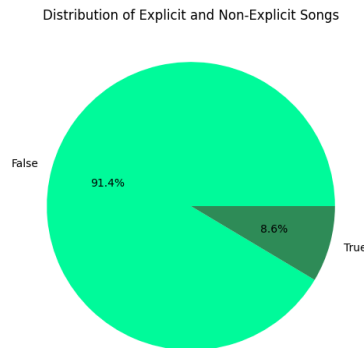
This column is the duration of the songs per milliseconds, so we start by observing the maximum and minimum of this column (8586, 5237295). In this column a process was made to better understand the data but this process will be seen in the transformations section.

Column explicit

For this column I verified the number of songs that are explicit and those that are not.

| Explicit | Count |
|----------|--------|
| False | 103831 |
| True | 9718 |

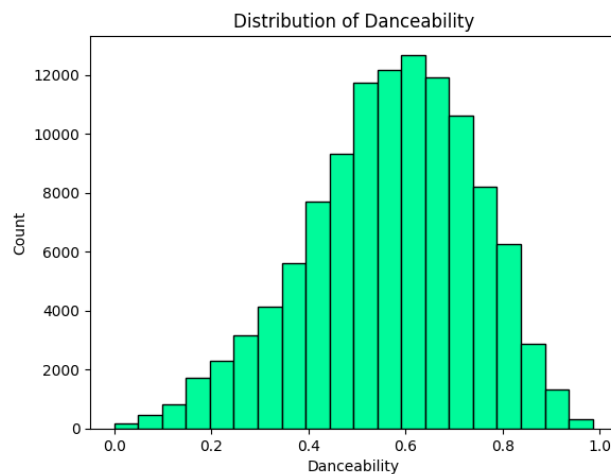
It is clear that most of the songs are not explicit and the proportion of the quantity was represented by a pie chart.



This could mean that the public prefers to listen to music that does not contain explicit content, so that is why artists do not have many songs with that content, perhaps these results have to do with platform policies or cultural aspects.

Column danceability

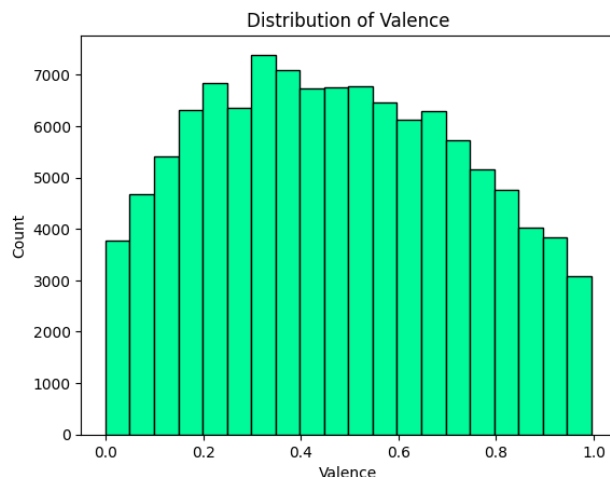
Danceability is a measure of how likely a song is to be danced to. So let's look at the distribution of the danceability in the dataset.



The graph shows that most of the songs have a danceability between 0.4 and 0.6. This means that most of the songs are moderately danceable. There are some songs with very high danceability (above 0.8) and some songs with very low danceability (below 0.2).

Column valance

This column turns out to be important since it allows us to evaluate the emotions expressed in the songs, allowing to be an important factor at the time of analysis.

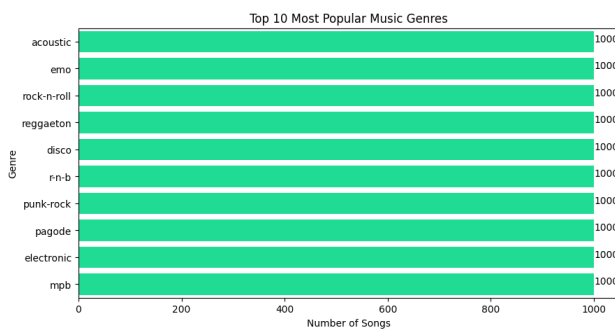


The dataset contains a mix of songs with different levels of valence.

- Most of the songs have a high valence value (between 0.4 and 1), meaning that they are perceived as positive and cheerful.
- There are a significant number of songs with a medium valence value (between 0.2 and 0.4), meaning that they are perceived as neutral.
- A smaller number of songs have a low valence value (between 0 and 0.2), meaning that they are perceived as negative and sad.

Column track_genre

We start by seeing that there are 114 genres, where we initially see which are the 10 most frequent genres:



A bar diagram was made to verify which are the 10 most frequent genres and the information is completely the same and standardized in all the observed genres. We then checked and found that the genders are very even, ranging in frequency from 904 to 1000.

Additional Information

We decided to see if there is some kind of relationship between the variables and to give possible conclusions

1. Average relationship between the variables 'genre' and 'danceability'.

| Genre | |
|----------------|----------|
| electronic | 0.643523 |
| global sounds | 0.587254 |
| instrumental | 0.469326 |
| jazz and soul | 0.560806 |
| latino | 0.638980 |
| mood | 0.529618 |
| pop | 0.579015 |
| rock and metal | 0.491721 |
| single genre | 0.592143 |
| urban | 0.642655 |
| varied themes | 0.556550 |

The results show whether a song is danceable varies significantly between different music genres. For example, electronic, Latin and urban music tend to have a higher average danceability compared to genres such as instrumental music or rock and metal. Which is quite logical because the genres with higher danceability have components that make them more danceable, such as instrumental or speechiness.

2. valence and popularity -0.04109661109347727

Since the correlation is close to zero and negative, it indicates that there is no strong or significant relationship between valence and popularity. In other words, there is no clear trend to suggest that the higher or lower the 'valence' of a song, the higher or lower its overall popularity.

3. danceability and valence 0.47675537653797145

Since the correlation is close to 0.5, it indicates that there is a moderate tendency: the higher the value of 'valence' (musical positivity), the more likely the song will be perceived as more suitable for dancing (higher 'danceability'). However, this is not a perfect relationship; there is variability and other factors at play.

Transformations

These transformations are intended to make it easier to analyze the data, allowing a better understanding and also serve for the analytics implemented later.

Grammys transformations

- We remove the columns that we are not going to use and the justification was mentioned in the eda. The columns to remove are img, published_at and update_at.
- Fill in the null values using regular expressions, mentioned earlier in the eda
- Remove the null values that were not filled in the previous process.
- Clean up the artists column by removing the parentheses resulting from the completion of the null values.
- Delete the workers column (it was not deleted at the beginning because it was useful to fill in the null values).
- Rename the name of the winner column to nominated so that in the future the information in this column will not be lost, since later on we have to perform a merge and it could be helpful.

Spotify Transformations

- Delete columns: as mentioned earlier in the eda, in this case the Unnamed : 0 column was deleted, and justifications were given as to why.
- Eliminate duplicates (mentioned in the eda).
- Popularity column: in this one the decision was made to make a transformation and create a new column with that. Then the values were categorized, remembering that the range goes from 0 to 100 as follows:

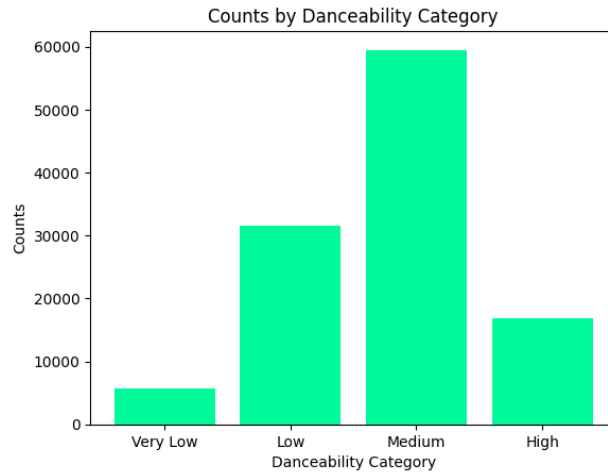
| Ranges |
|-------------------|
| Very Low: 0-20 |
| Low: 21-40 |
| Medium: 41-60 |
| High: 61-80 |
| Very High: 81-100 |

And the categories were left with the following amount of data:

| popularity_level |
|------------------|
| Very Low: 31736 |
| Low: 33264 |
| Medium: 33737 |
| High: 13613 |
| Very High: 1199 |

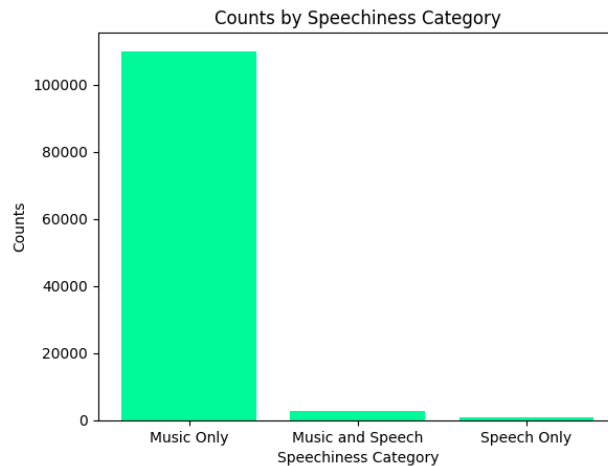
- Column duration_ms: As we observed in the eda we saw that this column is in milliseconds and it is very difficult to interpret it, then what will be done is to create a new column where the duration is in minutes per second. And the result is that there are songs from minute 00:00 to 55:40 minutes.
- Column danceability: Like the previous columns, a new column was categorized as follows:

| Ranges |
|---------------------|
| Very Low: 0 - 0.25 |
| Low: 0.26 - 0.5 |
| Medium: 0.51 - 0.75 |
| High: 0.76 - 1.0 |



- Most of the counts are in the medium bailability category.
- There are fewer counts in the very low and very high bailability categories.
- Danceability can be affected by factors such as rhythm, tempo, and instrumentation.
- Column speechiness: The same procedure was done in this column, categorizing and creating a new column as follows:

| Ranges |
|-------------------------------|
| Music Only: 0 - 0.33 |
| Music and Speech: 0.34 - 0.66 |
| Speech Only: 0.67 - 1.0 |

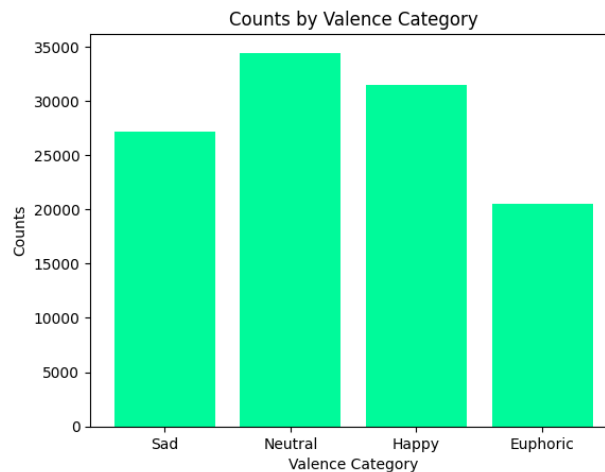


- The majority of the songs (about 90%) are from the "Music Only" category.
- A small percentage of the songs (about 2%) are from the "Music and Speech" category.
- An even smaller percentage of the songs (about 0.7%) are from the category "Speech Only".

The dataset is primarily composed of songs that have no spoken content. There are a small number of songs that have a mixture of music and speech. And there are a very small number of songs that are speech only.

- Column valence: This column turns out to be very important since it is the one that covers how much emotion a song has, so to understand it better it was also categorized:

| Rangos |
|----------------------|
| Sad: 0 - 0.25 |
| Neutral: 0.26 - 0.5 |
| Happy: 0.51 - 0.75 |
| Euphoric: 0.76 - 1.0 |



Sad: Songs with a melancholic or depressive tone.

Neutral: Songs with a neutral tone or without strong emotions.

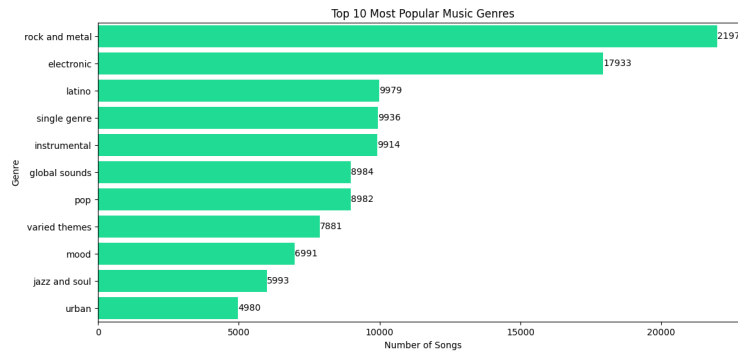
Happy: Songs with a happy or positive tone.

Euphoric: Songs with an extremely cheerful or festive tone.

- Most of the songs on Spotify have a neutral tone.
- There are a significant number of songs with positive valence (happy and euphoric).
- Sad songs are less common than songs with positive valence.
- Column track genre: as it was seen in the eda there were many genres, so the decision was made to group the genres by category taking into account their similarity, culture, rhythm, among other things.
 - **instrumental**: acoustic, classical, folk, guitar, piano, singer-songwriter, songwriter, world-music, opera, new-age,
 - **electronic**: afrobeat, breakbeat, chicago-house, club, dance, deep-house, detroit-techno, dub, dubstep, edm, electro, electronic, house, idm, techno, minimal-techno, trance, hardstyle,
 - **rock and metal**: alt-rock, alternative, british, grunge, hard-rock, indie, metal, metalcore, punk-rock, rock, rock-n-roll, black-metal, death-metal, hardcore, heavy-metal, industrial, psych-rock, rockabilly, goth, punk, j-rock, garage,
 - **pop**: anime, cantopop, j-pop, k-pop, pop, power-pop, synth-pop, indie-pop, pop-film,
 - **urban**: hip-hop, j-dance, j-idol, r-n-b, trip-hop,
 - **latino**: brazil, latin, latino, reggaeton, salsa, samba, spanish, pagode, sertanejo, mpb,
 - **global sounds**: indian, iranian, malay, mandopop, reggae, turkish, ska, dancehall, tango,
 - **jazz and soul**: blues, bluegrass, funk, gospel, jazz, soul,

- **varied themes:** children, disney, forro, grindcore, kids, party, romance, show-tunes,
- **mood:** ambient, chill, happy, sad, sleep, study, comedy,
- **single genre:** country, progressive-house, swedish, emo, honky-tonk, french, german, drum-and-bass, groove, disco

So now there are 11 gender categories with the following results:



We observe that we already have a much more valuable distribution to carry out analyzes and visualizations, this thanks to the grouping into 11 general categories for the genres. And here we can see that the most frequent genre is rock and metal, this may be because it is possibly the group that has the most genres, causing this high level of frequency to be generated.

Some additional important transformations were the following, these were performed mostly at the time of the merge so that it would not have incomplete or erroneous data or information:

- Replaces None values with NaN in integer columns of the DataFramer.
- Fills null values in the specified columns with the word 'Not nominated'.
- Fill the nulls in the nominated column with 0 and change the TRUE value to 1, in order to make the column easier to plot for the dashboard.

Merge

For the merge process between the Spotify data sets and the Grammy Awards data, the analytical strategy that we are developing in this Workshop was taken into consideration. Our objective is to evaluate the characteristics or parameters that influence the nomination of a song for the Grammy Awards. Following this purpose, we identify that the key column to carry out this process is 'track_name' in the Spotify data set, which contains the name of each song. Likewise, in the Grammy data set, we use the 'nominee' column, which indicates the nominated songs, albums or artists. This merge allows us to analyze in depth what musical characteristics, popularity metrics and other factors could influence the probability of a song being nominated for these prestigious awards. Once the data fusion has been performed, we apply the necessary transformations to prepare the data for analysis.

Airflow

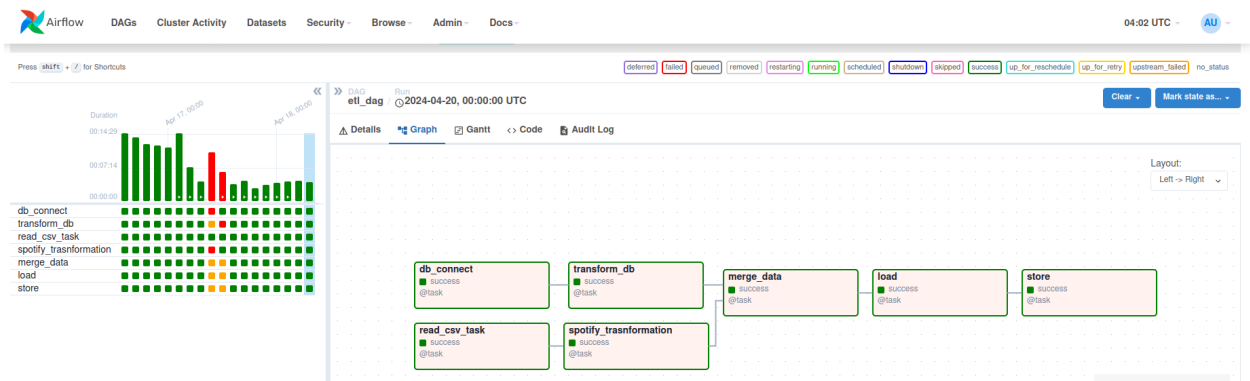
Now we demonstrate the airflow process, for this case an Ubuntu virtual machine was used, the reasons for using it is that it is much easier to handle and the airflow process. (The video of the installation of the machine is in the references). The project has a folder called dag, where there are 2 files: etl.py and etl_day.py, in the first one there are all the functions defined for the tasks specified for the airflow, and in the second file there are all the names defined for the tasks and also the functions created in the previous file are called, in this case we use decorators to create these tasks. To run the airflow it is important to have a virtual environment created, after this the following two commands are executed:

```

manuela@manuela-VirtualBox:~/prueba$ export AIRFLOW_HOME=$(pwd)
manuela@manuela-VirtualBox:~/prueba$ airflow standalone
standalone | Starting Airflow Standalone
standalone | Checking database is initialized
INFO [alembic.runtime.migration] Context impl SQLiteImpl.
INFO [alembic.runtime.migration] Will assume non-transactional DDL.
INFO [alembic.runtime.migration] Context impl SQLiteImpl.
INFO [alembic.runtime.migration] Will assume non-transactional DDL.
WARNI [airflow.models.crypto] empty cryptography key - values will not be stored encrypted.
standalone | Database ready

```

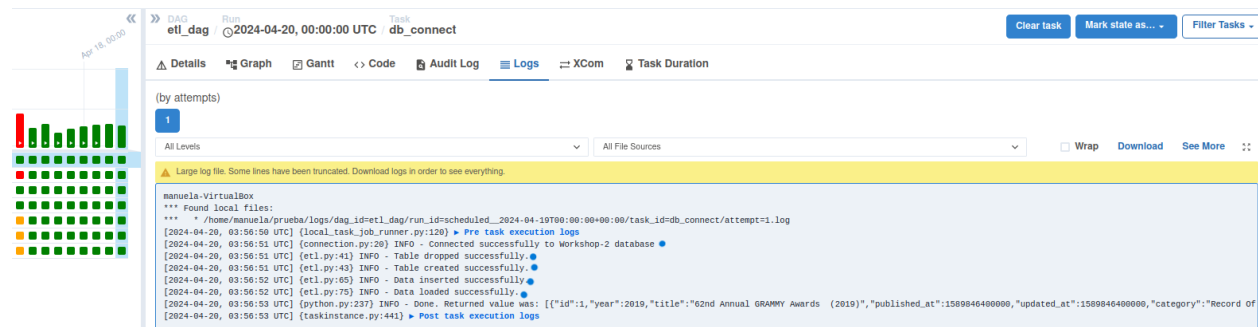
Then it will automatically create a file called 'standalone_admin:password.txt' where the password to access the local host to observe the airflow will be. After logging in and running the airflow, we can see that all the tasks are running effectively.



Now let's review what each task does and its respective log that evidences if the task is running correctly.

db_connection

In this task is the loading of the data of the grammys dataset, here the connection to postgres is made, the table is created, in case the table is already created it deletes it and recreates it, then inserts the data and finally loads the data to Postgres.



| id | year | title | published_at | updated_at | category | nominee |
|----|------|-----------------------------------|--------------|------------|--------------------|--|
| 1 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Bad Guy |
| 2 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Hey, Ma |
| 3 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | 7 rings |
| 4 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Hard Place |
| 5 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Talk |
| 6 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Old Town Road |
| 7 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Truth Hurts |
| 8 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Record Of The Year | Sunflower |
| 9 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Album Of The Year | When We All Fall Asleep, Where Do We Go? |
| 10 | 2019 | 62nd Annual GRAMMY Awards (201... | 2020-05-19 | 2020-05-19 | Album Of The Year | I |

Total rows: 1000 of 4810 Query complete 00:00:01.056 Ln 1, Col 1

We notice that the data is already loaded

transform_db

This taks is in charge of performing all the corresponding transformations for the grammys dataset, these transformations were already explained in the transformations part of the document, here we are going to prove that the taks was executed in the correct way.

etl_dag 2024-04-20, 00:00:00 UTC transform_db -1

Clear task Mark state as... Filter Tasks

Details Graph Gantt Code Audit Log Logs XCom Task Duration

(by attempts)

1

All Levels All File Sources Wrap Download See More

Large log file. Some lines have been truncated. Download logs in order to see everything.

```

manuela-VirtualBox
*** Found local files:
***   /home/manuela/prueba/logs/dag_id=etl_dag/run_id=scheduled__2024-04-19T00:00:00+00:00/task_id=transform_db/attempt=1.log
[2024-04-20, 03:57:58 UTC] [local_task_job_runner.py:120] ▶ Pre task execution logs
[2024-04-20, 03:57:58 UTC] [logging_mixin.py:180] WARNING - /home/manuela/prueba/transformation/grammys.py:26 UserWarning: This pattern is interpreted as a regular expression, and has match groups. To actually get the
[2024-04-20, 03:57:58 UTC] [etl.py:183] INFO - Data transformed successfully
[2024-04-20, 03:57:58 UTC] [python.py:237] INFO - Done. Returned value was: [{"year":2019,"category":"Record Of The Year","nominee":"Bad Guy","artist":"Billie Eilish","nominated":true}, {"year":2019,"category":"Record O
[2024-04-20, 03:57:58 UTC] [taskinstance.py:441] ▶ Post task execution logs

```

read_csv_task

This taks is in charge of loading the Spotify dataset from a csv file.

etl_dag 2024-04-20, 00:00:00 UTC read_csv_task

Clear task Mark state as... Filter Tasks

Details Graph Gantt Code Audit Log Logs XCom Task Duration

(by attempts)

1

All Levels All File Sources Wrap Download See More

Large log file. Some lines have been truncated. Download logs in order to see everything.

```

manuela-VirtualBox
*** Found local files:
***   /home/manuela/prueba/logs/dag_id=etl_dag/run_id=scheduled__2024-04-19T00:00:00+00:00/task_id=read_csv_task/attempt=1.log
[2024-04-20, 03:56:57 UTC] [local_task_job_runner.py:120] ▶ Pre task execution logs
[2024-04-20, 03:56:59 UTC] [etl.py:117] INFO - Data loaded successfully
[2024-04-20, 03:56:59 UTC] [python.py:237] INFO - Done. Returned value was: [{"Unamed": "0", "track_id": "5SuoIkwiRyPwvIQ0Jug5v", "artists": "Gen Hoshino", "album_name": "Comedy", "track_name": "Comedy", "popularity": 73, "dura
[2024-04-20, 03:57:03 UTC] [taskinstance.py:441] ▶ Post task execution logs

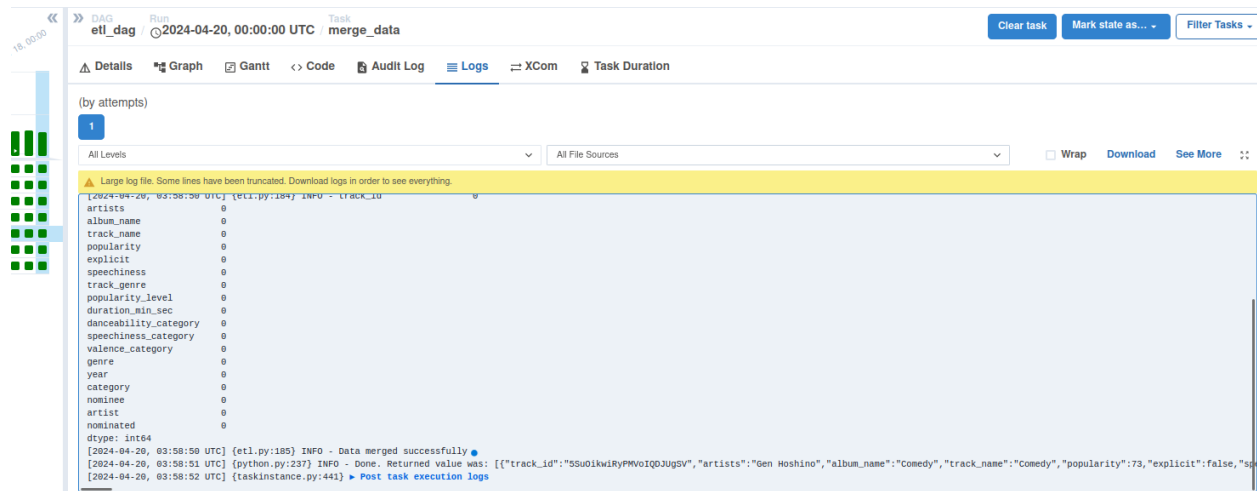
```

spotify_transformation

In this taks are all the transformations corresponding to the Spotify dataset, which were specified previously, for this case it was not possible to touch the evidence because due to resource problems the log of the taks does not load, but in the taks of the load of the merge we are going to show the new columns that were created from the transformations of this dataset. It is worth mentioning that the log did load before but due to different inconveniences the evidence could not be registered.

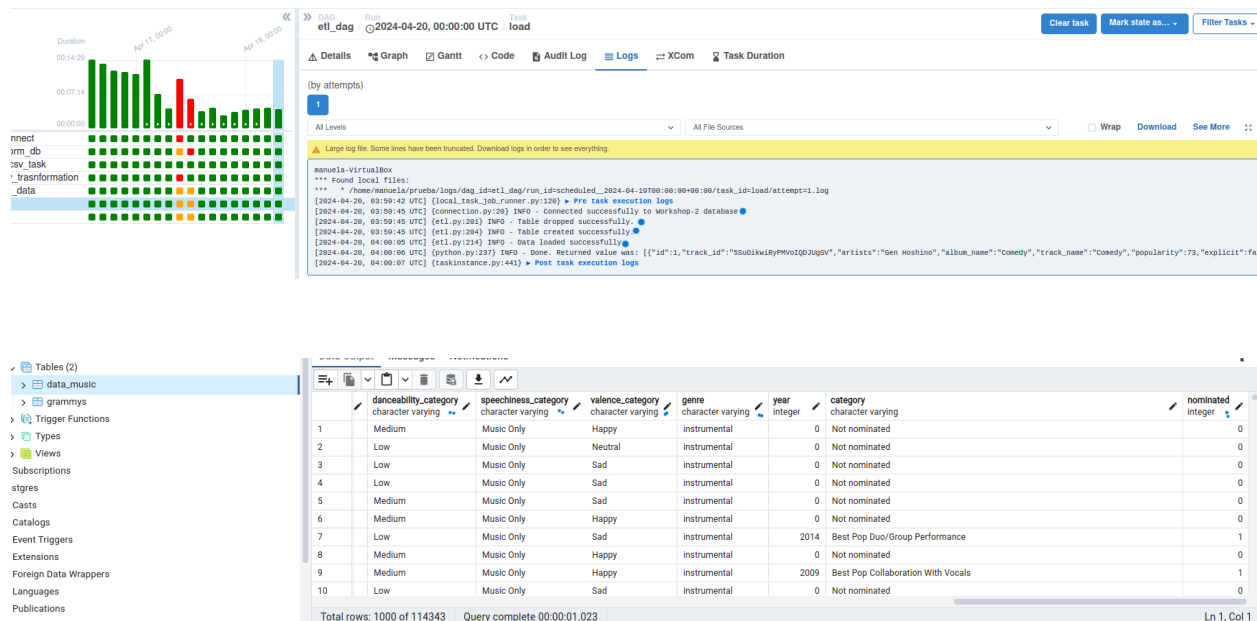
merge_data

The merge procedure was already explained above, and in this task we merge the data and see that the task is successful.



load

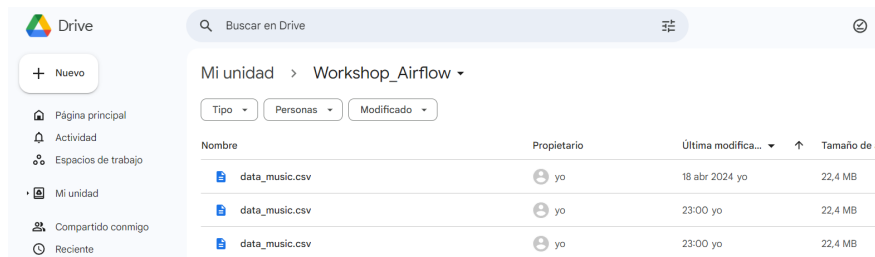
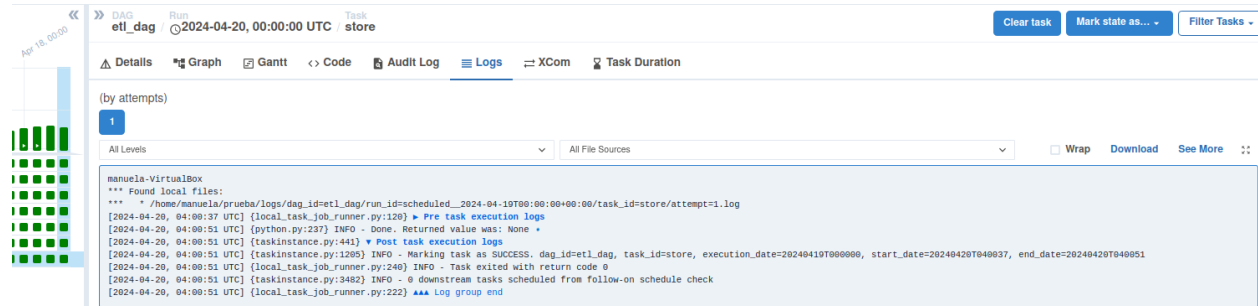
In this task the result of the merge is loaded to postgres, the process of connection, creation or deletion (if applicable) of the table, insertion of the data to the created table and finally the loading to Postgres, as it was done in the procedure of the first task.



Here we can see that the data is loaded in Postgres and as we could not take the evidence of the spotify transformations, I decided to show some of the columns that were generated after these transformations to show that they had been completed successfully (they are the ones with the blue dot).

store

The operation of this task is to upload the dataset resulting from the merge to Drive using the google Drive API to perform this procedure, in the etl file is all the function to perform this procedure.

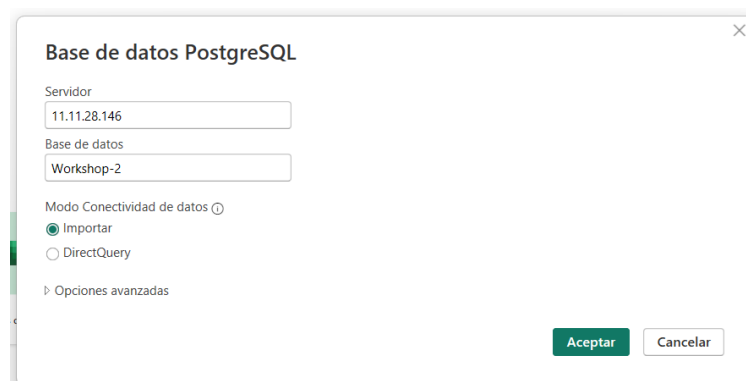


Here we can see that it was loaded correctly, and several files are visible because the whole dag has been executed several times.

Power BI Visualization

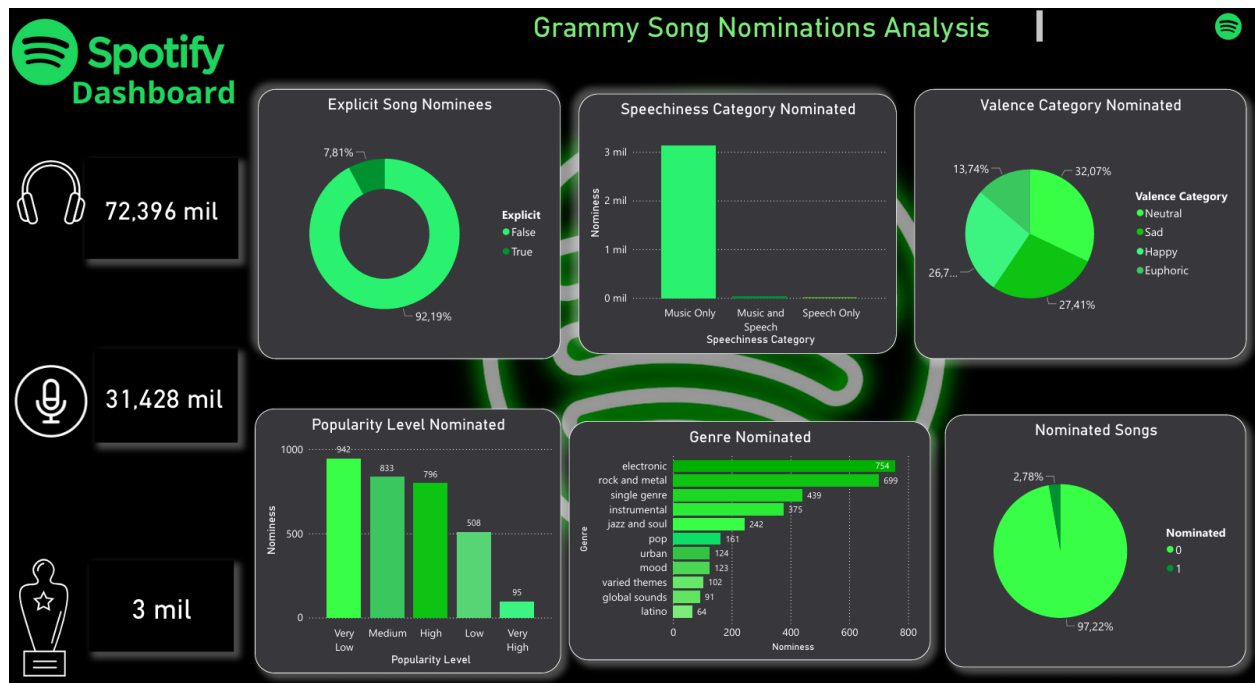
As mentioned above, we are working from the Ubuntu virtual machine, and to display the data in Power BI, a Power BI connection process to the localhost was carried out. To do this, port 5432 was opened to remote connections, and We change the virtual machine to bridge mode and from there we look at the IP address shown in ifconfig.

We see that the IP is 11.11.28.146 and the following configuration is made from Power Bi, from the 'obtain data from another source' option we select 'PostgreSQL Database' and configure it as follows, the base is placed of data you have been working on:

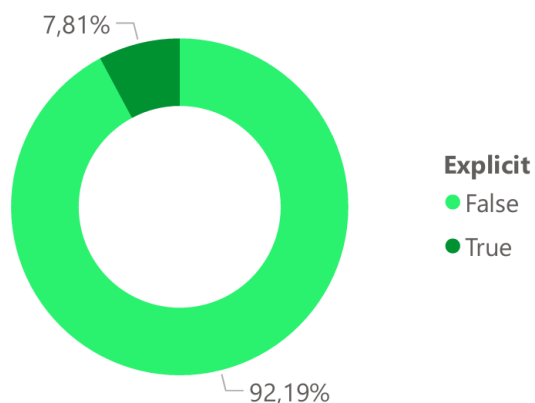


Dashboard

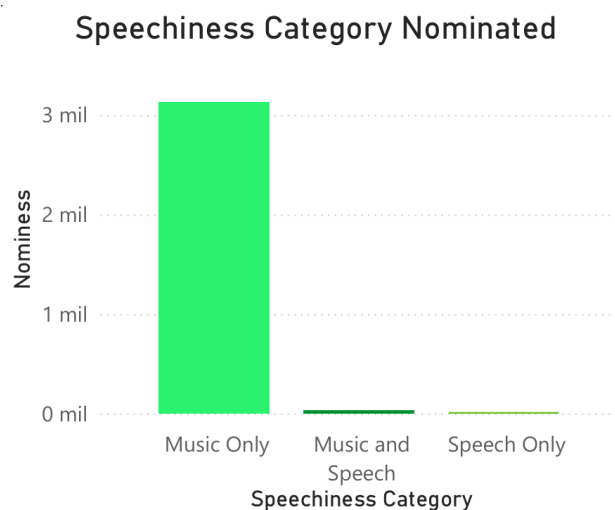
The analysis chosen for this Workshop is to analyze the metrics or characteristics that make a song nominated for the Grammys.



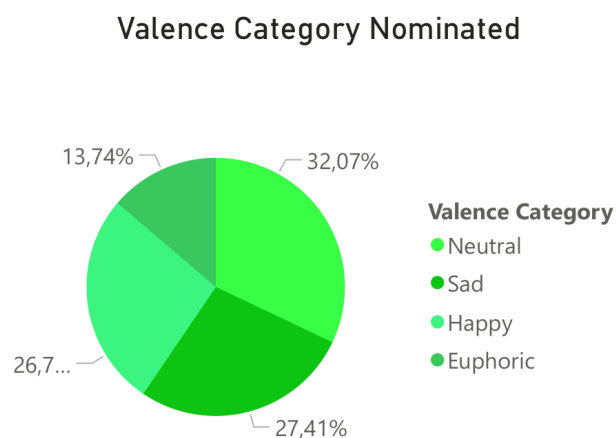
Explicit Song Nominees



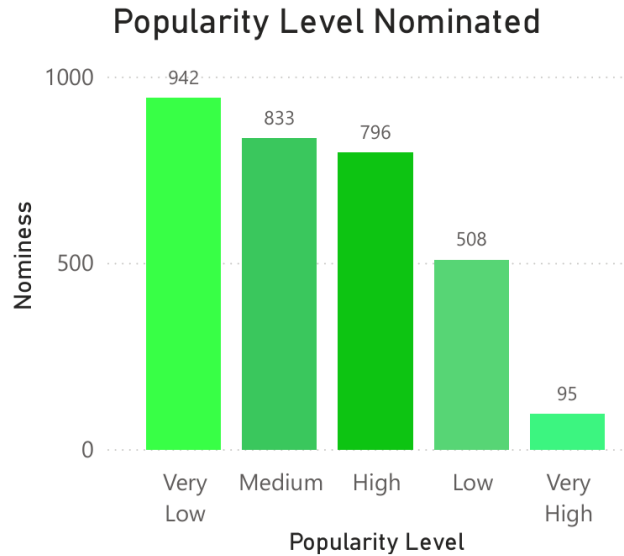
The explicit songs chart shows the distribution of Grammy-nominated songs according to their explicit content. A song is considered explicit if it contains lyrics or content that is considered offensive or inappropriate. From this data, we can interpret that the Grammy Awards have a bias toward songs that are not explicit. That is, songs with offensive or inappropriate lyrics or content are less likely to be nominated for awards. This suggests that Grammy voters value music that is appropriate for a general audience.



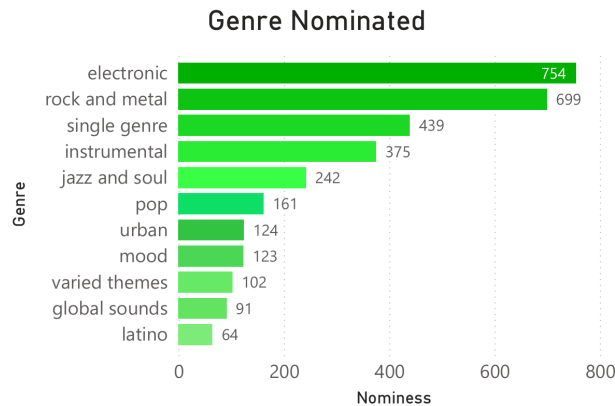
For this column a transformation was made which consisted in categorizing in 3 different groups the speechless which qualifies if a song has only music or only speech, in this case we see that the most nominated songs are the ones that have only music. It is quite curious and this may be related to its level of danceability or the instrumental it has, and that makes these songs to be considered for nomination.



This graph was considered quite important to perform the analysis since it contains information about the level of feeling or emotion transmitted by the songs, since the valence is a measure of the positivity or negativity of an emotion, in this case a transformation was also performed which consists of separating the information into 4 groups, where the group with the highest number of nominations is Neutral. From this data, we can interpret that the Grammy Awards do not have a bias towards any particular type of emotional valence. The songs nominated for the awards span a wide range of emotions, from sadness to elation. This suggests that Grammy voters value musical diversity and reward songs that convey a wide range of emotions.

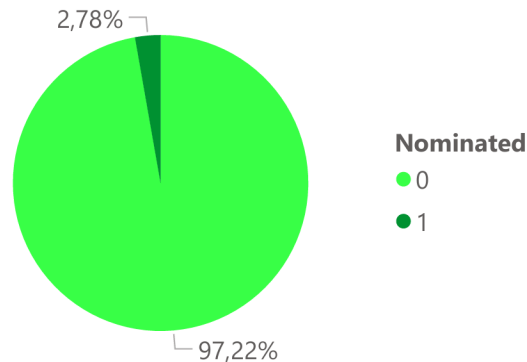


The chart of Grammy nominations by popularity level shows the distribution of Grammy nominated songs according to their level of popularity. In this case, the songs were classified into four categories by making a transformation. We can interpret that the Grammy Awards do not have a bias towards popular songs. That is, songs with a higher number of plays on streaming platforms are not more likely to be nominated for awards than songs with a lower number of plays. This suggests that Grammy voters value factors other than popularity when selecting nominated songs.



The Grammy nominations by genre chart shows the distribution of Grammy nominated songs according to their musical genre. In this case, the genres were grouped into 11 categories. We can interpret that the Grammy Awards are quite diverse in terms of the musical genres they nominate. No single genre dominates the nominations, and a wide range of genres are represented.

Nominated Songs



From this graph we observe that of the number of songs in the dataset there are very few songs that have been nominated for the grammys, this may be due to the fact that there are very few categories or the number of songs nominated per category.

References

For the virtual machine and some configurations:

- GroverTec. (2024, 24 febrero). *Como Instalar Python en Linux Ubuntu: Guía Sencilla para Principiantes* [Video]. YouTube. <https://www.youtube.com/watch?v=88np4KkfD08>
- Enreta Services. (2022, 22 abril). *Cómo instalar UBUNTU 22.04 PASO a PASO desde cero! TUTORIAL* [Video]. YouTube. <https://www.youtube.com/watch?v=8MRibUo9VAA>
- kipuna ec. (2024, 25 febrero). *Instalar git en Ubuntu* [Video]. YouTube. <https://www.youtube.com/watch?v=4M8cL-1XANQ>
- Roelcode. (2022, 13 septiembre). *Instalar PostgreSQL 14 y PgAdmin4 en Linux Ubuntu 22.04 y distros basados en Ubuntu* [Video]. YouTube. <https://www.youtube.com/watch?v=5sP36Hdh4wU>

EDA

- 2nd Annual GRAMMY Awards — GRAMMY.com. (s. f.). <https://www.grammy.com/awards/2nd-annual-grammy-awards>
- Grammy Awards. (2020, 16 septiembre). Kaggle. <https://www.kaggle.com/datasets/unanimad/grammy-awards>
- Pandas DataFrame describe() Method. (s. f.). https://www.w3schools.com/python/pandas/ref_df_describe.asp
- Python RegEx. (s. f.). https://www.w3schools.com/python/python_regex.asp
- Spotify Tracks Dataset. (2022, 22 octubre). Kaggle. <https://www.kaggle.com/datasets/maharshipandya/-spotify-tracks-dataset>

PyDrive

- MoonCode. (2021, 7 marzo). *Aprende a usar Google Drive con Python en 20 minutos -Learn Python and Google Drive in 20 minutes* [Video]. YouTube. <https://www.youtube.com/watch?v=ZI4XjwbpEwU>

- Welcome to PyDrive's documentation! — PyDrive 1.2.1 documentation. (s. f.). <https://pythonhosted.org/PyDrive/>