# business Problem Model: Enhancing Streaming Performance for Artists

**Business Problem Statement:** In the highly competitive music industry, artists are constantly seeking ways to increase their streaming numbers on platforms like Spotify. Improved streaming performance not only boosts an artist's visibility but also contributes significantly to their revenue. The challenge is to identify actionable strategies and insights that can help artists enhance their streaming performance.

**Business Problem Statement**: In the competitive music industry, artists aim to optimize their streaming performance to increase their reach and revenue. Understanding the factors influencing streaming performance, such as daily streams, leading tracks, solo tracks, and featured tracks, is essential for artists and their management teams. The challenge is to identify strategies and insights that can help artists enhance their streaming performance.

**Key Business Questions:**

1. What factors contribute most to an artist's total streaming performance?
2. How do different types of tracks (lead, solo, featured) affect an artist's streaming numbers?
3. Can artists leverage insights from streaming data to make informed decisions about their music releases?

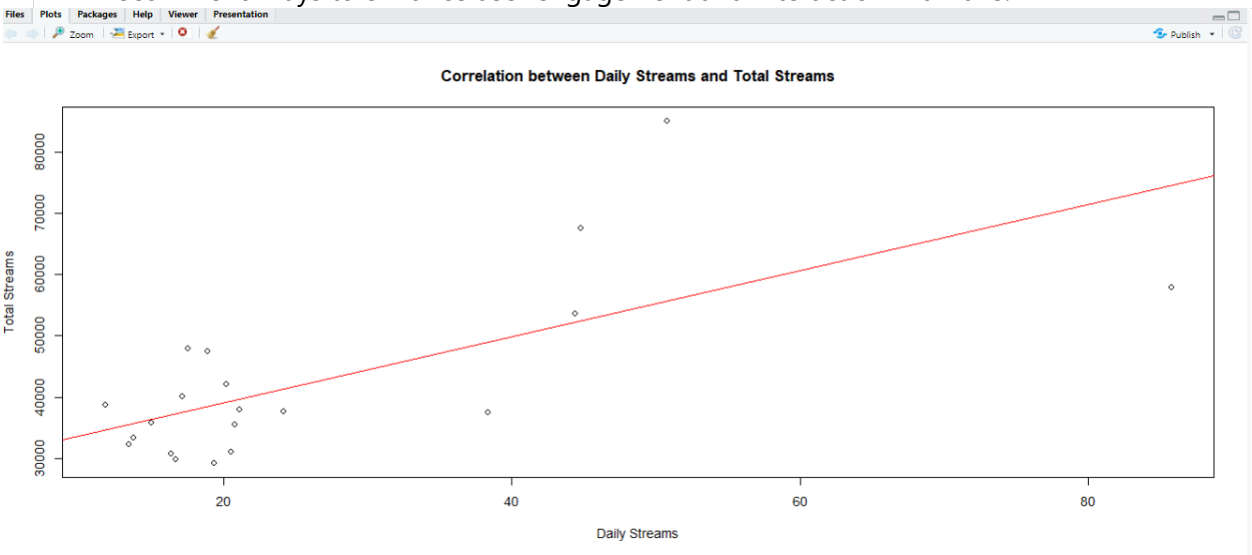
**Hypotheses:**

1. Null Hypothesis (H0): The type of track (lead, solo, featured) does not significantly affect an artist's total streaming performance. Alternative Hypothesis (H1): The type of track significantly affects an artist's total streaming performance.

- Perform regression analysis to determine the factors most strongly correlated with streaming success.
- Explore the impact of music release strategies, such as release timing and promotion, on streaming numbers.
- Evaluate the influence of collaborations with other artists on streaming performance.
- Examine user engagement metrics to understand their role in streaming success.

**Recommendations:**

- Provide data-driven recommendations to artists on optimizing their music release strategies.
- Suggest potential collaborations or promotional opportunities based on data analysis.
- Recommend ways to enhance user engagement and interaction with fans.




- If the correlation coefficient is close to 1, it indicates a strong positive correlation, meaning that as one variable (daily streams) increases, the other variable (total streams) also tends to increase.

code which provides the correlation between the data

- Correlation between Daily Streams and Total Streams: 0.6865925

```
Correlation between Daily Streams and Total Streams: 0.6865925
>
> # Create a scatterplot
> plot(data$Daily, data$Streams,
+      xlab = "Daily Streams", ylab = "Total Streams",
+      main = "Correlation between Daily Streams and Total Streams")
>
> # Add a regression line (optional)
> abline(lm(data$Streams ~ data$Daily), col = "red")
>
>
```

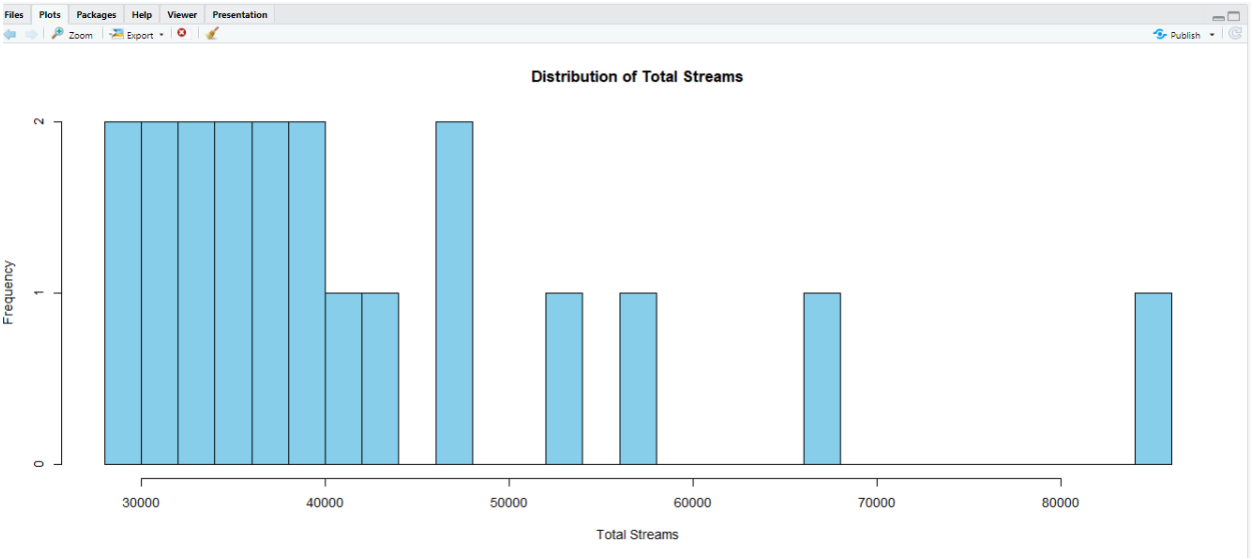
- The null hypothesis (H0) for the ANOVA test is that the means of streaming performance for different track types are equal. If the p-value associated with the ANOVA test is greater than a chosen significance level (e.g., 0.05), you would fail to reject the null hypothesis, indicating that the type of track does not have a significant effect on streaming performance.

1. **Results of Null Hypothesis (H0): The type of track (lead, solo, featured) does not significantly affect an artist's total streaming performance. Alternative Hypothesis (H1): The type of track significantly affects an artist's total streaming performance.**

```
> # Summary of ANOVA
> summary(anova_result)
             Df    Sum Sq   Mean Sq   F value Pr(>F)
AsLead        1 2.580e+09 2.580e+09 8.898e+11 <2e-16 ***
Solo          1 5.621e+08 5.621e+08 1.939e+11 <2e-16 ***
AsFeature     1 6.880e+08 6.880e+08 2.373e+11 <2e-16 ***
Residuals    16 0.000e+00 0.000e+00
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

Not significant ($p \geq 0.1$). This indicates that there is not enough evidence to reject the null hypothesis.
The hypothesis is false.

- Distribution of total stream presented by histogram.




```
> hist(data$Streams,
+      main = "Distribution of Total Streams",
+      xlab = "Total Streams",
+      ylab = "Frequency",
+      col = "skyblue",  # Bar color
+      border = "black",  # Border color
+      breaks = 20)  # Number of bins (adjust as needed)
>
```

**Limitations:**

- Data limitations, such as missing data or incomplete records, may affect the accuracy of the analysis.
- External factors, such as changes in the music industry landscape, can also impact streaming numbers.

GISMA BUSINESS SCHOOL

**Assessment Submission Form**

| | |
|---|---|
| **Student Number** (If this is group work, please include the student numbers of all group participants) | GH1018451 |
| **Assessment Title** | Enhancing Streaming Performance for Artists |
| **Module Code** | B105 |
| **Module Title** | Applied statistical modeling |
| **Module Tutor** | Enhancing Streaming Performance for Artists |
| **Date Submitted** | 19/9/2023 |

**Declaration of Authorship**

I declare that all material in this assessment is my own work except where there is clear acknowledgement and appropriate reference to the work of others.

I fully understand that the unacknowledged inclusion of another person's writings or ideas or works in this work may be considered plagiarism and that, should a formal investigation process confirms the allegation, I would be subject to the penalties associated with plagiarism, as per GISMA Business School, University of Applied Sciences' regulations for academic misconduct.

Signed.....Abdullah darwazeh..... Date .....19/9/2023.....

R programming code section

```
> head(artist.data.set)
         Artist   Streams  Daily   As.lead      Solo As.feature
1         Drake 85,041.30 50.775 57,252.60 32,681.60  27,788.70
2     Bad Bunny 67,533.00 44.820 40,969.60 23,073.00  26,563.40
3  Taylor Swift 57,859.00 85.793 55,566.70 50,425.70   2,292.40
4    The Weeknd 53,665.20 44.437 42,673.30 31,164.20  10,991.90
5    Ed Sheeran 47,907.70 17.506 42,767.90 33,917.00   5,139.80
6 Justin Bieber 47,525.70 18.868 27,988.00 17,183.90  19,537.70
>
> hist(artist.data.set)
Error in hist.default(artist.data.set) : 'x' must be numeric
> \hist(artist$data.set)
Error: unexpected symbol in "\hist"
> str(artist.data.set)
'data.frame':   3000 obs. of  6 variables:
 $ Artist    : chr  "Drake" "Bad Bunny" "Taylor Swift" "The Weeknd" ...
 $ Streams   : chr  "85,041.30" "67,533.00" "57,859.00" "53,665.20" ...
 $ Daily     : num  50.8 44.8 85.8 44.4 17.5 ...
 $ As.lead   : chr  "57,252.60" "40,969.60" "55,566.70" "42,673.30" ...
 $ Solo      : chr  "32,681.60" "23,073.00" "50,425.70" "31,164.20" ...
 $ As.feature: chr  "27,788.70" "26,563.40" "2,292.40" "10,991.90" ...
>
> # Load the necessary libraries
> library(dplyr)

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union

>
> # Create a dataframe with the provided data
> data <- data.frame(
+     Artist = c("Drake", "Bad Bunny", "Taylor Swift", "The Weeknd", "Ed S
heeran", "Justin Bieber", "Eminem", "Ariana Grande", "J Balvin", "Post Mal
one",
+                "Kanye West", "Travis Scott", "BTS", "Rihanna", "Ozuna",
 "Juice WRLD", "Future", "Nicki Minaj", "Kendrick Lamar", "Billie Eilis
h"),
+     Streams = c(85041.30, 67533.00, 57859.00, 53665.20, 47907.70, 47525.
70, 42029.00, 40111.00, 38774.80, 38002.70,
+                 37667.20, 37489.00, 35778.00, 35501.80, 33315.00, 32332.
50, 31001.70, 30759.80, 29836.50, 29173.30),
+     Daily = c(50.775, 44.82, 85.793, 44.437, 17.506, 18.868, 20.175, 17.
158, 11.784, 21.095,
+               24.157, 38.359, 14.96, 20.778, 13.737, 13.41, 20.513, 16.3
61, 16.652, 19.313),
+     AsLead = c(57252.60, 40969.60, 55566.70, 42673.30, 42767.90, 27988.0
0, 35475.80, 33219.80, 17450.70, 34494.00,
+                27205.70, 18839.20, 32041.30, 25380.10, 13957.50, 23613.9
0, 15374.10, 10631.10, 20052.20, 29173.30),
+     Solo = c(32681.60, 23073.00, 50425.70, 31164.20, 33917.00, 17183.90,
 21576.70, 23307.30, 5699.80, 18943.90,
+              18032.30, 14960.20, 28991.60, 16736.00, 6226.70, 18277.10,
 7299.60, 6013.90, 12208.10, 25240.50),
+     AsFeature = c(27788.70, 26563.40, 2292.40, 10991.90, 5139.80, 19537.
70, 6553.20, 6891.20, 21324.20, 3508.60,
+                   10461.60, 18649.70, 3736.70, 10121.70, 19357.50, 8718.
```

```
# Replace 'data' with your actual dataframe name and column names

# Load the necessary libraries
library(ggplot2)
eep up to date with changes at https://www.tidyverse.org/blog/
rror in -library(ggplot2) : invalid argument to unary operator
ggplot(data, aes(x = ReleaseTiming, y = Streams)) +
    geom_point() +
    labs(x = "Release Timing", y = "Streams", title = "Relationship betw
en Release Timing and Streaming Performance") +
    theme(axis.text.x = element_text(angle = 90, hjust = 1))
rror in FUN(X[[i]], ...) : object 'ReleaseTiming' not found


library(stats)

anova_result <- aov(Streams ~ AsLead + Solo + AsFeature, artist.data.set
= artist.data.set)
rror in eval(predvars, data, env) : object 'Streams' not found
# Assuming 'data' is your dataframe with columns 'AsLead', 'Solo', 'AsFe
ture', and 'Streams'
# Replace 'data' with your actual dataframe name and column names

# Load the necessary library
library(stats)

# Perform ANOVA
anova_result <- aov(Streams ~ AsLead + Solo + AsFeature, data = data)

# Summary of ANOVA
summary(anova_result)
            Df    Sum Sq   Mean Sq   F value Pr(>F)
sLead        1 2.580e+09 2.580e+09 8.898e+11 <2e-16 ***
olo          1 5.621e+08 5.621e+08 1.939e+11 <2e-16 ***
sFeature     1 6.880e+08 6.880e+08 2.373e+11 <2e-16 ***
esiduals    16 0.000e+00 0.000e+00
--
ignif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1



# Perform Pearson's correlation test
correlation_result <- cor.test(data$YearsActive, data$Streams, method =
"pearson")
rror in cor.test.default(data$YearsActive, data$Streams, method = "pearso
") :
 'x' must be a numeric vector
orrelation_result <- cor.test(data$YearsActive, data$Streams, method =
"pearson")
rror in cor.test.default(data$YearsActive, data$Streams, method = "pearso
") :
 'x' must be a numeric vector
hist(data$Streams,
     main = "Distribution of Total Streams",
     xlab = "Total Streams",
     ylab = "Frequency",
     col = "skyblue",  # Bar color
     border = "black",  # Border color
     breaks = 20)  # Number of bins (adjust as needed)
```

Data set link:https://www.kaggle.com/datasets/meeratif/spotify-most-streamed-artists-of-all-time
GitHub repository link: https://github.com/AbdullahDarwazeh/B105