Do Musicians learn from Feedback? Examination of Popular songs from Spotify

Dr Preet Deep Singh

09/09/2021

Abstract

Using a dataset of 5,86,672 songs from Spotify, we evaluate whether artists learn from the response their songs get or not. We find that popularity of songs improves with iterations. We plot whether artists have their biggest hits in their first year or subsequent years.

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.6
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

##
## Please cite as:

##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables

##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer

## Rows: 170633 Columns: 151

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (11): Country, Uri, Title, Artist, Album/Single, Genre, Album, Release_...
## dbl (139): Popularity, Artist_followers, Track_number, Tracks_in_album, danc...
## lgl   (1): Explicit

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

## New names:
## * `` -> ...1
```

```
## Rows: 101939 Columns: 32
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (15): album_id, analysis_url, artists_id, available_markets, country, hr...
## dbl (17): ...1, acousticness, danceability, disc_number, duration_ms, energy...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## New names:
## * `` -> ...1
## Rows: 115062 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): Artist, Date1
## dbl  (3): ...1, Year1, LastYear
## date (3): GoldenYear, GoldenDate, LastDate
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

In a series of papers, I try to examine the following questions #. . . . . # . # What does the average album look like: how many sad, what balance # Do albums with similar songs do better # Do albums have multiple hit songs or only 1/2 songs make it ? # Do artists experiment after a hit or do more of the same? Are they scare or overconfident? # Do negative songs do better? # Do winter songs do better? # When does a artist shine to fame: first or next or last? # Is the same song a hit in multiple places? What are unique geographic hits? In this paper, I look at the most popular hits of artists. I examine whether artists get their biggest hits in the first go and spend the rest of their lives trying to match that success or whether artists enter the market, release music, take feedback and improve in order to produce their most popular song. In case artists are learning from feedback, the popularity of their songs would have an upward trend. We regress

In case artists do not take feedback, their most popular songs would be spread randomly across their careers. For artists who had their most popular song in their first year, the subsequent years would entail a lot of experiments, most of which would fail and eventually they would retire. This means we should see artists with their best hits coming in the first year, dropping off in a couple of years post that. The year/date of the last song by a particular band can be used to check this hypothesis.

Our data spans a total of 586672 songs by 115062 across 102 years. This includes 25864 explicit songs.

##WHen do you know you have failed?

1 Data

1.1 Summary

- Total songs
- Total Artists
- Total Albums
- Average Popularity
- Year wise distribution

```
##
## % Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac
## % Date and time: Sun, Nov 07, 2021 - 20:38:34
## \begin{table}[!htbp] \centering
##   \caption{This table shows the summary of the dataset.}
##   \label{}
## \begin{tabular}{@{\extracolsep{5pt}}lccccccc}
## \\[-1.8ex]\hline
## \hline \\[-1.8ex]
## Statistic & \multicolumn{1}{c}{N} & \multicolumn{1}{c}{Mean} & \multicolumn{1}{c}{St. Dev
## \hline \\[-1.8ex]
## \hline \\[-1.8ex]
## \end{tabular}
## \end{table}
```

```
## New names:
## * `` -> ...1
```

```
## Rows: 115062 Columns: 8
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): Artist, Date1
## dbl  (3): ...1, Year1, LastYear
## date (3): GoldenYear, GoldenDate, LastDate
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu % Date and time: Sun, Nov 07, 2021 - 20:38:34

#.....
We look at the year of the first song by the artist and the year of the most popular song by the artist. For example, if the artist released the first song in 1997, and their most popular song was released in 2001, then we note 2001-1997

Table 1: This table

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max

= 4 as the time till the first hit. In case the most popular song was released in the same year, this number would be 0.

This analysis allows us to answer the following questions # Who got it soonest # How many got it soonest # How many in 1 year # Who got it late # How many after 5 years

1.2 Regression

Sequence of songs Only the most popular song on any date, incase there were multiple songs on the same date Only artists with more than 1 song Get details of the music Regression on Popularity on All the factors Find that sequence is a major contributor along with Explicit

```
## New names:
## * `` -> ...1
```

```
## Rows: 211986 Columns: 11
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): id, name, artists, id_artists
## dbl  (6): ...1, popularity, duration_ms, explicit, YearRelease, seq
## date (1): release_date
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu % Date and time: Sun, Nov 07, 2021 - 20:39:16

2 Limitations

Limited dataset: It is possible that there are some songs by certain artists that are not part of the half a million songs I use. In some cases, the biggest hit of a few artists might have existed in a particular year and is not accounted for in this study. However, our dataset is not biased in terms of vintage or recency. Therefore, I do not expect this to be systematically lopsided. This data is from Spotify. The popularity of the app has gone up in each of the previous few years. It is possible that songs that were not hits in the first go, in the first geography subsequently became hits. This would not be accounted for in our study.

Table 2: This table shows the results of regression of popularity on sequence and control variables.

	Dependent variable:	
	popularity.x	
	(1)	(2)
seq	0.010***	0.009***
	(0.001)	(0.001)
duration_ms.x	−0.00000***	0.00000**
	(0.00000)	(0.00000)
danceability	0.177	3.512***
	(0.150)	(0.173)
energy	1.721***	14.882***
	(0.157)	(0.178)
key	−0.033***	0.006
	(0.004)	(0.005)
loudness	0.226***	0.569***
	(0.006)	(0.007)
mode	0.512***	0.136***
	(0.033)	(0.038)
speechiness	−4.433***	−19.589***
	(0.085)	(0.088)
acousticness	0.126*	−9.101***
	(0.074)	(0.082)
instrumentalness	−7.788***	−18.410***
	(0.074)	(0.080)
valence	−0.486***	−11.589***
	(0.096)	(0.106)
tempo	−0.003***	−0.006***
	(0.0005)	(0.001)
time_signature	0.145***	0.145***
	(0.016)	(0.019)
explicit.x	8.430***	13.420***
	(0.115)	(0.132)
Constant	−778.496***	38.088***
	(2.013)	(0.234)
Year	Yes	No
Observations	494,991	494,991
R^2	0.732	0.641