User Engagement As A Function Of Audio Attributes Using Multivariate Regression

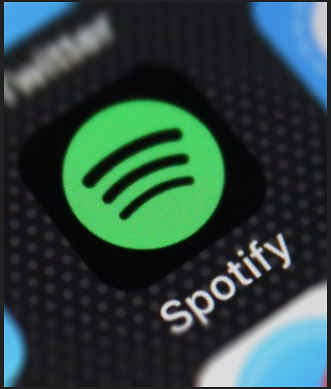
Jack Fahrnow
Luke Atkins
Akshika Rai
Yichen Gao

Background

- Spotify is the world’s largest streaming service provider that has over 381 million monthly active listeners and 171 million paid subscribers.
- Spotify had around 1.1 billion downloads on the Google Store in 2021 alone.
- Artist post their tracks on Spotify using a separate software Ditto, which helps artist release unlimited tracks in 100 different apps.
    - It varies how the Spotify Artist get paid
    - It depends on
        - The location of the listener
        - Whether the stream came from a paid or free subscription account

    In order for an artist to go viral on spotify, they would need to focus on a certain niche for their song,

        - They have the opportunity to explore and create content beyond music by incorporating their passions and interests into their daily conversation with their fans.



Goal

Given a set of songs, all sharing the relation of being from the same artist; use multi variable regression modeling to infer what attributes contribute most to the popularity of the songs.

Potential Uses

Aid in formally determining the niche of a given artist, help labels better understand what their audiences expect from an artist.

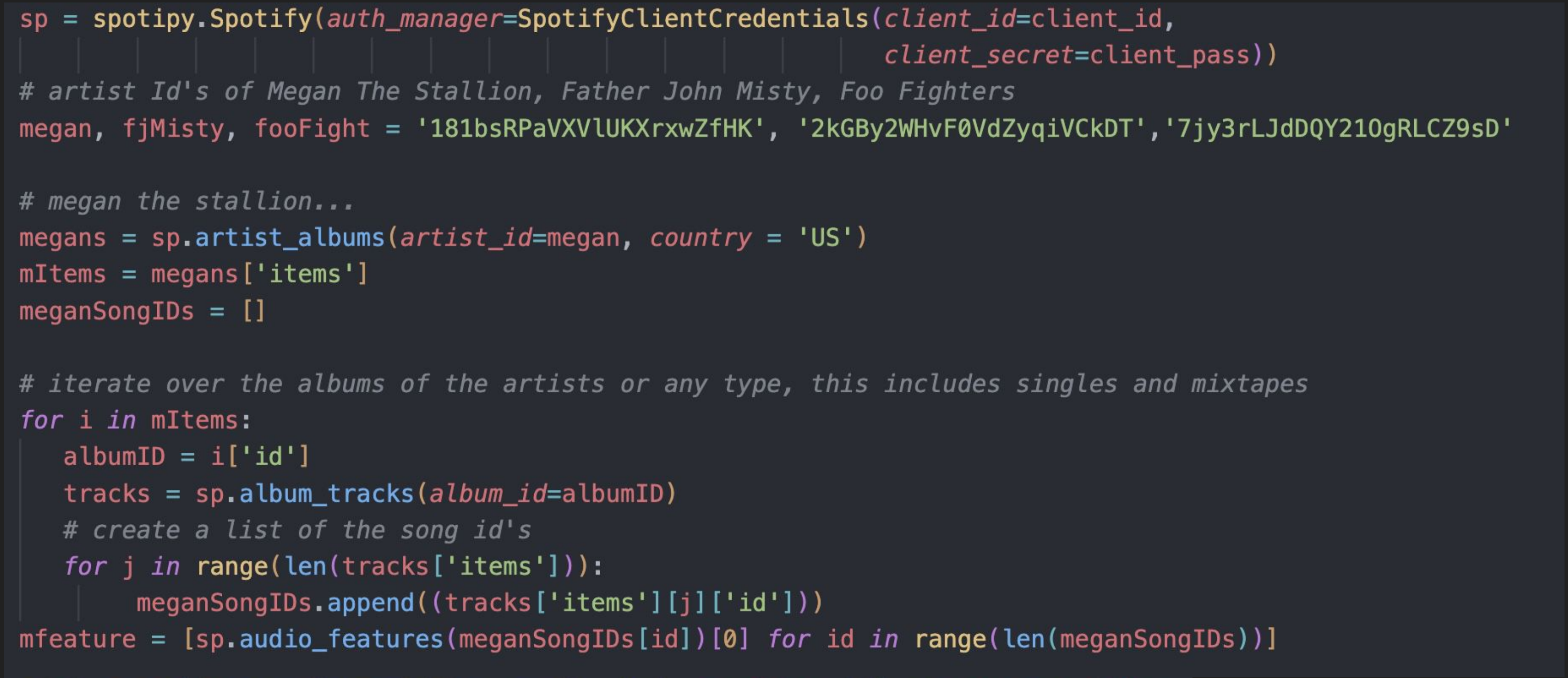
Dataset

We have 2 sets of data that we went through to see. 200 of the top songs from Foo Fighters, and Father John Misty

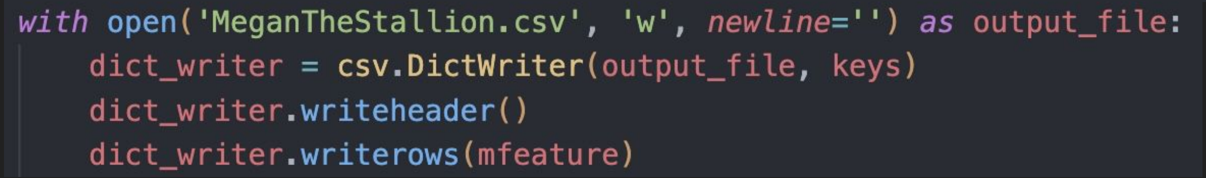
Rock

Alternative/Indie

Data Collection with Spotify API interacted with through the python library Spotipy, then written to a csv format, here is an example with Megan Thee Stallion!

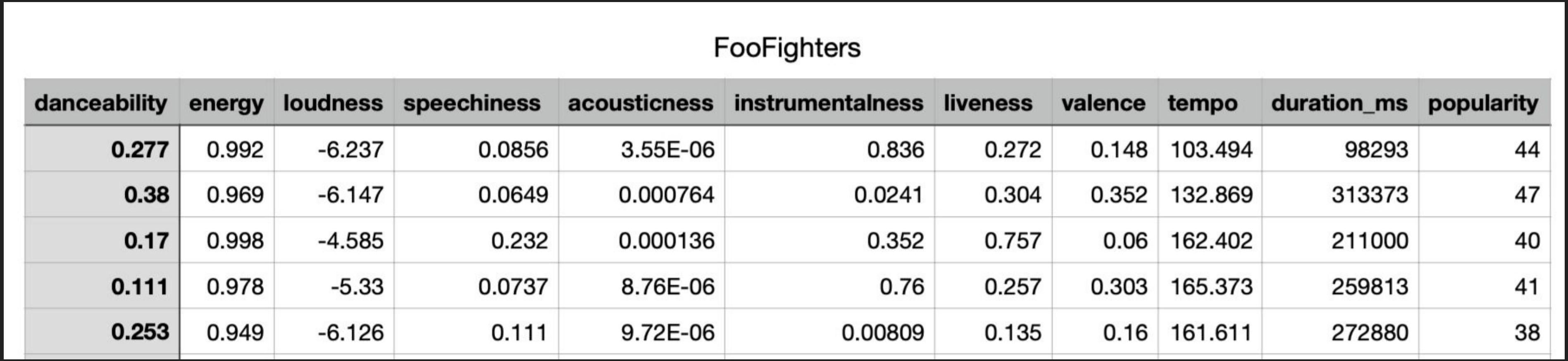


```python
sp = spotipy.Spotify(auth_manager=SpotifyClientCredentials(client_id=client_id,
                                                           client_secret=client_pass))
# artist Id's of Megan The Stallion, Father John Misty, Foo Fighters
megan, fjMisty, fooFight = '181bsRPaVXVlUKXrxwZfHK', '2kGBy2WHvF0VdZyqiVCkDT','7jy3rLJdDQY21OgRLCZ9sD'

# megan the stallion...
megans = sp.artist_albums(artist_id=megan, country = 'US')
mItems = megans['items']
meganSongIDs = []

# iterate over the albums of the artists or any type, this includes singles and mixtapes
for i in mItems:
   albumID = i['id']
   tracks = sp.album_tracks(album_id=albumID)
   # create a list of the song id's
   for j in range(len(tracks['items'])):
        meganSongIDs.append((tracks['items'][j]['id']))
mfeature = [sp.audio_features(meganSongIDs[id])[0] for id in range(len(meganSongIDs))]
```



```python
with open('MeganTheStallion.csv', 'w', newline='') as output_file:
    dict_writer = csv.DictWriter(output_file, keys)
    dict_writer.writeheader()
    dict_writer.writerows(mfeature)
```

Example of Pre-Cleaned Song Entry Data



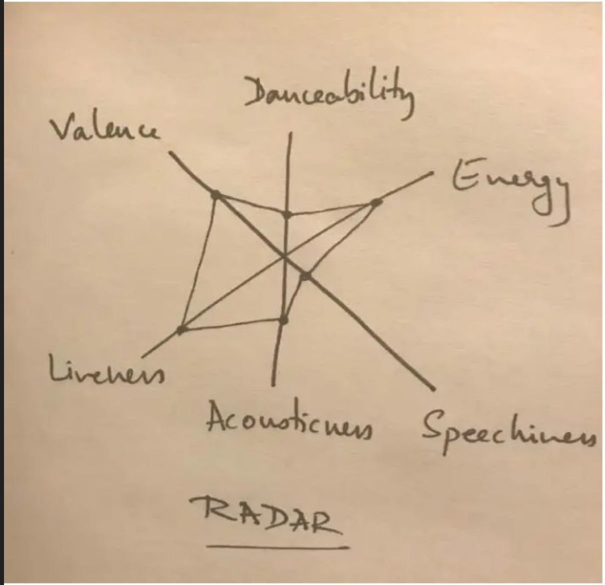
FooFighters

danceability	energy	loudness	speechiness	acousticness	instrumentalness	liveness	valence	tempo	duration_ms	popularity
0.277	0.992	-6.237	0.0856	3.55E-06	0.836	0.272	0.148	103.494	98293	44
0.38	0.969	-6.147	0.0649	0.000764	0.0241	0.304	0.352	132.869	313373	47
0.17	0.998	-4.585	0.232	0.000136	0.352	0.757	0.06	162.402	211000	40
0.111	0.978	-5.33	0.0737	8.76E-06	0.76	0.257	0.303	165.373	259813	41
0.253	0.949	-6.126	0.111	9.72E-06	0.00809	0.135	0.16	161.611	272880	38
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

This format follows for all sampled artists.

Attributes

- Danceability
- Energy
- Loudness
- Speechiness
- Acousticness
- Instrumentalness



- Liveness
- Valence
- Tempo
- Duration_ms
- Rank/Popularity

Spotify Attributes Meaning

- Danceability: describes how suitable a track is for dancing based on a combination of music elements
- Loudness: gives us the decibels of the track.
- Speechiness: detects the presence of spoken words.
- Acousticness: measures the confidence whether the track is acoustic
- Instrumentalness: Predicts if the track has no vocals.
- Liveness: detects presence of an audience, if the track was performed live.
- Valence: shows the musical positiveness conveyed by a track.
- Tempo: the overall beats per minute/pace in a track.
- Duration_ms: The length of the track.
- Rank/Popularity: the ranking of the track (how popular the song is)

Spotify Attributes that Weren’t being Used

- Mode
- Key
- URI
- E.t.c
- Mode: indicates if the track is major or minor
- Key: tells the pitch class with a few notations
  - The regression model works best if the attributes are continuous, so the variables that aren’t being used were removed for this model to work.

Data Preparations and Regression Methods

The code below is the regression model for the artist Father John Misty

```
x = read.csv2("FatherJohnMisty.csv",stringsAsFactors=FALSE,sep=",");
x = sapply(x, as.character)
x = data.matrix(x)
x = apply(x, 2, as.numeric) #changing the data type from character to numeric

new_data = scale(x[,9:11]) #Standardizing the dataset
View(new_data)
new_x = cbind(x[,1:8],new_data) #Binding the columns together
View(new_x)

X = as.matrix(new_x[,1:10]);
y = as.vector(new_x[,11]);
a = solve(t(X) %*% X, t(X) %*% y)
paste("coefficients are:")
a
```

Data Preparations and Regression Methods Contd.

```
yhat = X %*% a
sse = sum((y - yhat) ^ 2)
paste("sse =", sse) #Finding the sum of squared errors

new_sse = rep(0, 10)
diff = rep(0, 10)
for (i in 1:10) {
    newX = X[,-i]
    a = solve(t(newX) %*% newX, t(newX) %*% y)
    yhat = newX %*% a
    new_sse[i] = sum((y - yhat) ^ 2)
    diff[i] = abs(new_sse[i] - sse) #Computing the new sum of squared errors by omitting one attribute at a
time.
    cat(
        paste0(
            "Omit attribute", i , ": new sse = ", new_sse[i],", difference with old sse = ",diff[i],".\n" )
    )
```

Data Preparations and Regression Methods Contd.

```
}
cat("Most difference is" , max(diff), "\n") #Popularity most affected by this attribute

cat("Least difference is" , min(diff), "\n") #Popularity least affected by this attribute

plot(diff, main = "Effect on Popularity" , ylab = "Difference between old and new sse" , xlab = "Attributes")
#Goal of the model is to output the attribute which causes the most and least difference between the new sse and
the old sse
#Which will help us determine the attribute with the most and least effect on the popularity of an artist's
track
```

A similar regression model is made for Foo Fighters to attain the attribute that affects the popularity of their songs the most and the least.

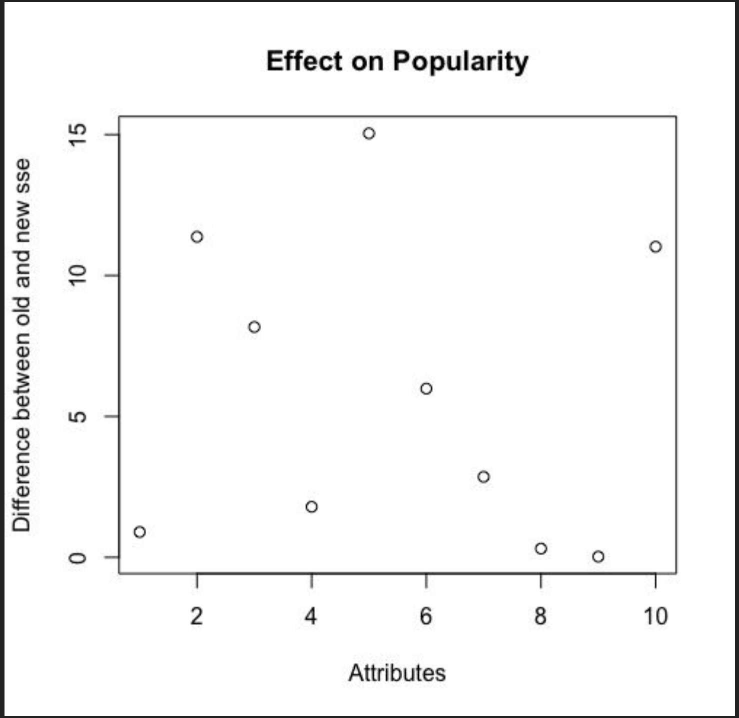
Results and Interpretation : Artist Father John Misty

```
Most difference is 1.911314
Least difference is 0.005982919
```

Father John Misty is an Alternative/Indie artist whose songs' popularity as observed in the plot, is most affected by the acousticness of his tracks. A track's popularity is least affected by the tempo.

As we have seen in the previous slides, acousticness is a measure of the confidence level of a track, is Acoustic or not. Tempo is the overall beats per minute/pace in a track.

As Father John Misty is an Alternative/Indie artist, his songs contain heavy use of acoustics, and the more acoustic the track is, the more popular it will be. An Alternative/Indie track's tempo does not affect its popularity because most have a low tempo (they are slow songs).



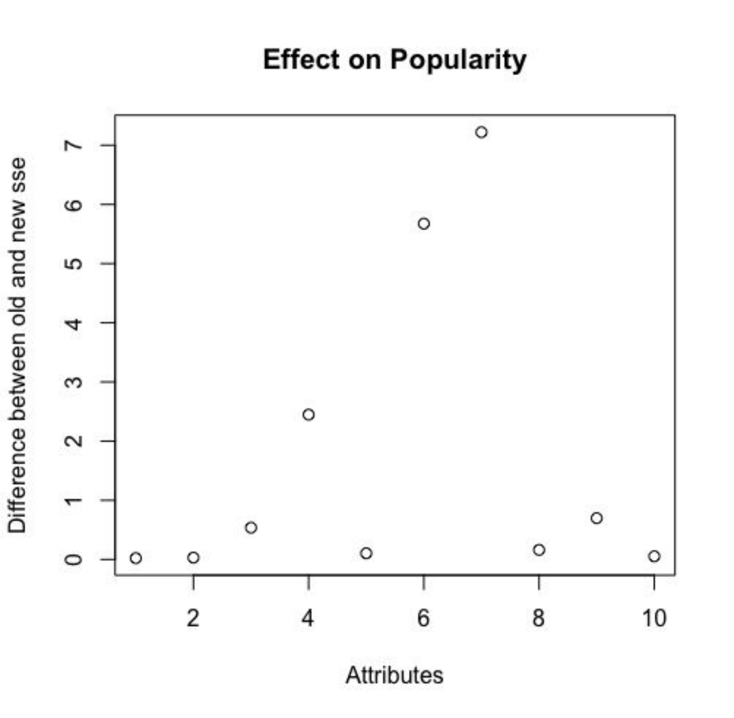
Results and Interpretation : Artist Foo Fighters

```
Most difference is 7.220158
Least difference is 0.02211429
```

Foo Fighters is a Rock Band whose songs' popularity, as observed in the plot, is most affected by the liveness of their tracks. A track's popularity is least affected by danceability.

As we have seen in the previous slides, liveness is an attribute that detects if the songs are performed live. Danceability is a measure of how danceable a track is.

The reason liveness affects the popularity of Foo Fighters' songs is because fans of the genre like live recorded rock music more.



Challenges

- Small data set
- Mapping numbers to artistic objects
- Dropped attributes have high impact on user engagement ex: (mode, key)
- Regression uses continuous random variables while music has discrete attributes

Future Directions

- Create new metrics for measuring the qualities of music…
    - Timber of sound, feeling, mood
- Improving Spotify's recommendation system and music promotion service
- Devise methods of modeling experience of a song.