# Predicting Popularity Of A Song using Spotify Data

By Sandhya Kiran Reddy Donthireddy

# Data

- **Rows -** 174389 **,Features –** 18
- **Features -** ['key', 'mode', 'explicit', 'acousticness', 'danceability', 'duration_ms', 'energy', 'instrumentalness', 'liveness', 'loudness', 'speechiness', 'tempo', 'valence', 'year', 'artists' , 'release_date', 'name' , 'id']
- **Target -** 'Popularity'

# Question?

- **Determine how well the features explain the popularity of songs on Spotify and Predict the popularity of a song based on these features**

# Pre-Processing

**Categorical Features:**

- Missing values
- One Hot Encoding

**Continuous Features:**

- Missing Values
- Standard Scaler

# Algorithms

- **Ridge -** ['alpha','solver']
- **Extra Trees Regressor -** ['n_estimators', 'max_depth', 'min_samples_leaf', 'max_features', 'bootstrap']
- **Random Forest Regressor -** ['n_estimators', 'max_depth', 'min_samples_leaf', 'max_features', 'bootstrap']

# Best Model

- **RandomForestRegressor(max_depth=20, min_samples_leaf=5, n_estimators=30)**

# Interpretation

- R-Squared value - 0.68 and it implies that model explains 68% of the fitted data and that gives an indication that the features in the Spotify data could be used to interpret the popularity of the songs.

# Summary

- Worked because the data set has diverse and predictive features.
- Useful in business setting when trying to understand which songs could become popular based on features such as tempo, danceability etc.,
- Could also be used further to recommend songs to users based on the common features in the songs they listen to.

# Next Steps

- Improving the model fit by using other exhaustive methods like Grid Search.
- Finding if there is multicollinearity between the features and dropping those features.

# ThankYou