Dataset

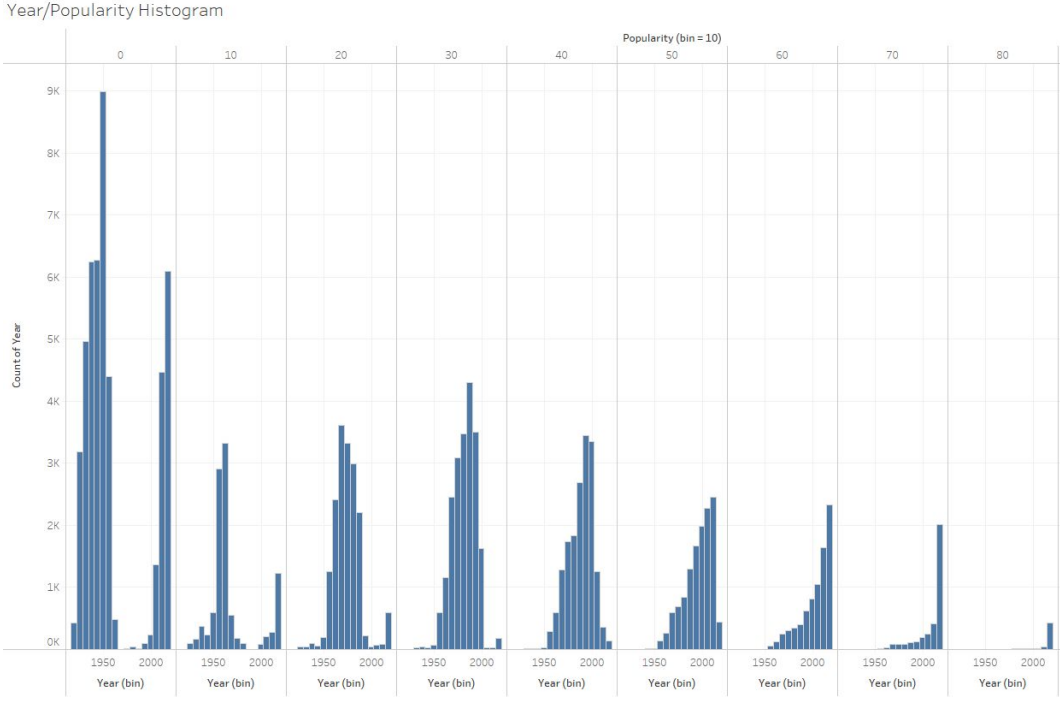
Size: 170K+ Songs → 137K+ Songs

Fields include

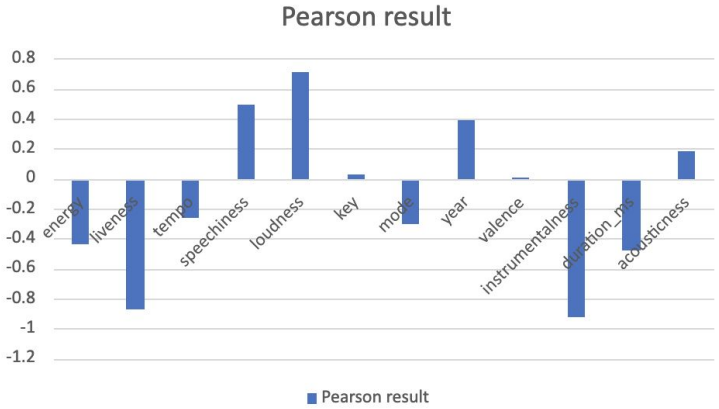
- Continuous [0,1]: Acousticness, Danceability, Energy, Instrumentalness, Liveness, Speechiness, Valence
- Continuous: Duration, Loudness, Tempo
- Discrete/Binary: Explicit, Mode, Key, Release Month, Year, Popularity
- Identifiers: Artists, Name, ID

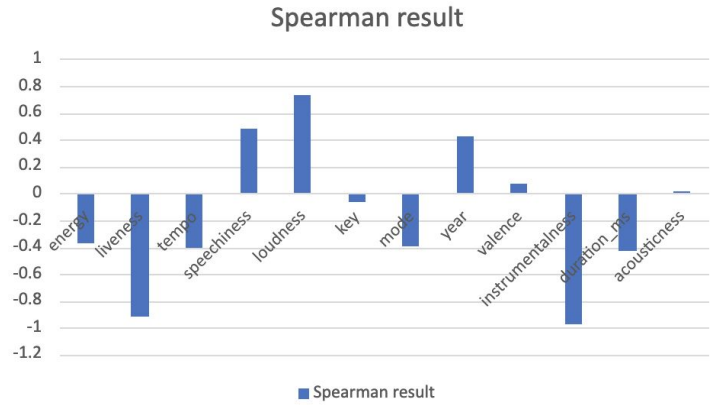
Source: Github (spags093) via Spotify API

Exploratory Data Analysis



Statistical Analysis





- Pearson & Spearman Rank Correlation Coefficient

Strong Relationship:

- Liveness (-0.91)
- Speechiness (0.48)
- Loudness (0.73)
- Instrumentalness (-0.97)
- Duration_m (-0.48)

Can we predict whether a song will be popular?

Classification Methodology:

- Mean Dataset Popularity: 37.7
- Popular: 38+ popularity (36K songs)
- Not Popular: 37- popularity (29K songs)

Parameters

- Input: Acousticness, Danceability, Energy, Instrumentalness, Liveness, Speechiness, Valence, Duration, Loudness, Tempo
- Output: 0 (not popular) or 1 (popular)

Decision Tree Classifier

- K-fold Validation: 0.726
- Testing F1: 0.725

Logistic Regression

- K-fold Validation: 0.715
- Testing F1: 0.724

Can we predict the exact popularity score of a song?

Parameters

- Input: Acousticness, Danceability, Energy, Instrumentalness, Liveness, Speechiness, Valence, Duration, Loudness, Tempo
- Output: Popularity [0,100]

Gradient Boosting Regressor

- K-fold Validation: 0.329
- Testing R^2: 0.334
- Pairwise ranking accuracy: 0.69

Linear Regression

- K-fold Validation: 0.163
- Testing R^2: 0.167

Next Steps

- Popularity Prediction
  - Adjust the year threshold filter of the data set to reach a model with better accuracy
  - Adjust the classification methodology to see if there is an improvement in classifying popular and not popular songs
  - Improve the accuracy of the popularity score prediction models with parameter tuning
  - Explore other supervised learning models/techniques such as support-vector machine, ensemble learning
- Explore: Unsupervised learning models to provide song recommendations based on similarity, e.g. clustering algorithms