Maddie Bauer
DSC 550
Original Case Study

Topic: Analyze Features of Songs from Spotify to Understand a Song's Popularity Score

Hypothesis:

What features of a song can lead to a higher popularity score on Spotify? Are any particular features of a song highly correlated to one another and/or have a significant effect on their popularity score?

Background:

Music plays a big part in many peoples lives, including mine. For this analysis, I wanted to explore data from my favorite music platform, Spotify, to understand what features of a song leads it to being more popular. Features of songs include items such as tempo, energy level, mood, loudness and more. Most people simply enjoy the music they are listening to without thinking in depth about the features mentioned above, which is why I wanted to explore this data to see if any of these features play an important role in determining the overall popularity of a song. This information can be useful for song producers, artists, music marketers and other professionals within the music industry.

The Data:

I found my data set on Kaggle. It consists of 603 observations and 15 variables (listed below) for the most popular song titles between the years of 2010 and 2019.

Song Number – The song's number in a set
Title – Title of the song
Artist – Artist of the song
Top Genre – The genre of the track
Year – The song's year in the Billboard
Bpm – (Beats Per Minute) The tempo of the song
Nrgy – (Energy) The energy of a song – the higher the value, the more energetic song
Dnce – (Danceability) The higher the value, the easier it is to dance to this song

dB – (Loudness) The higher the value, the louder the song
Live – (Liveness) The higher the value, the more likely the song was recorded live
Val – (Valence) The higher the value, the more positive mood for the song
Dur – (Length) The duration of the song
Acous – (Acousticness) The higher the value, the more acoustic the song is
Spch – (Speechiness) The higher the value, the more spoken word the song contains
Pop – (Popularity) The higher the value, the more popular the song is
(Source: http://organizeyourmusic.playlistmachinery.com/)

<u>Graph Analysis:</u>

Step 1: Load data into a dataframe

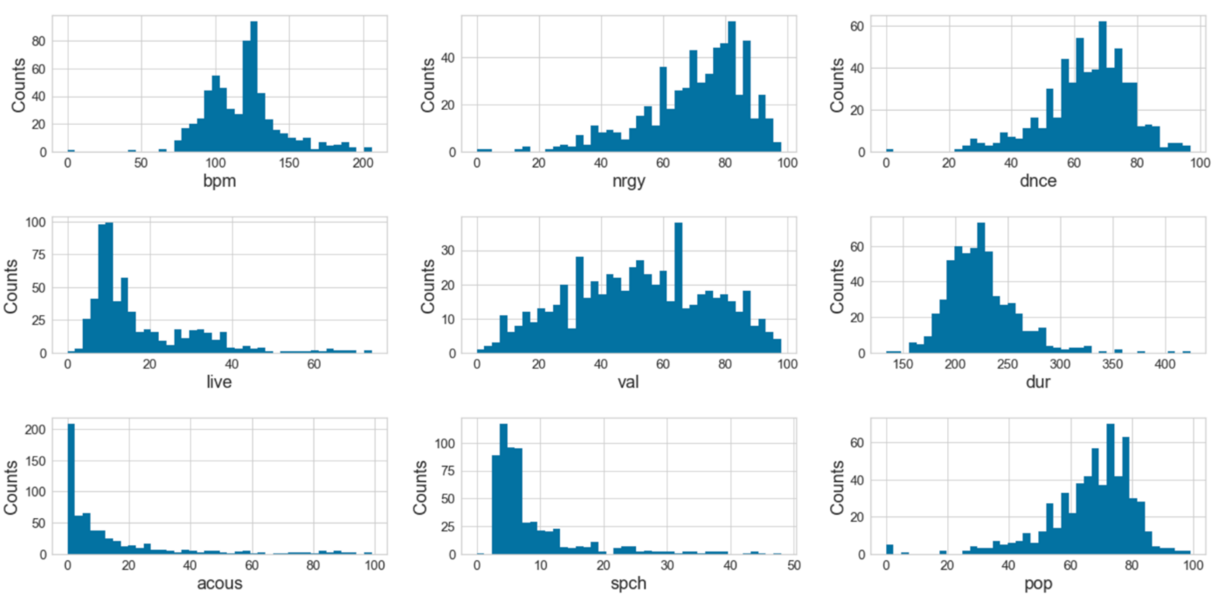
Step 2: Check the dimensions of the dataframe

Step 3: View the data

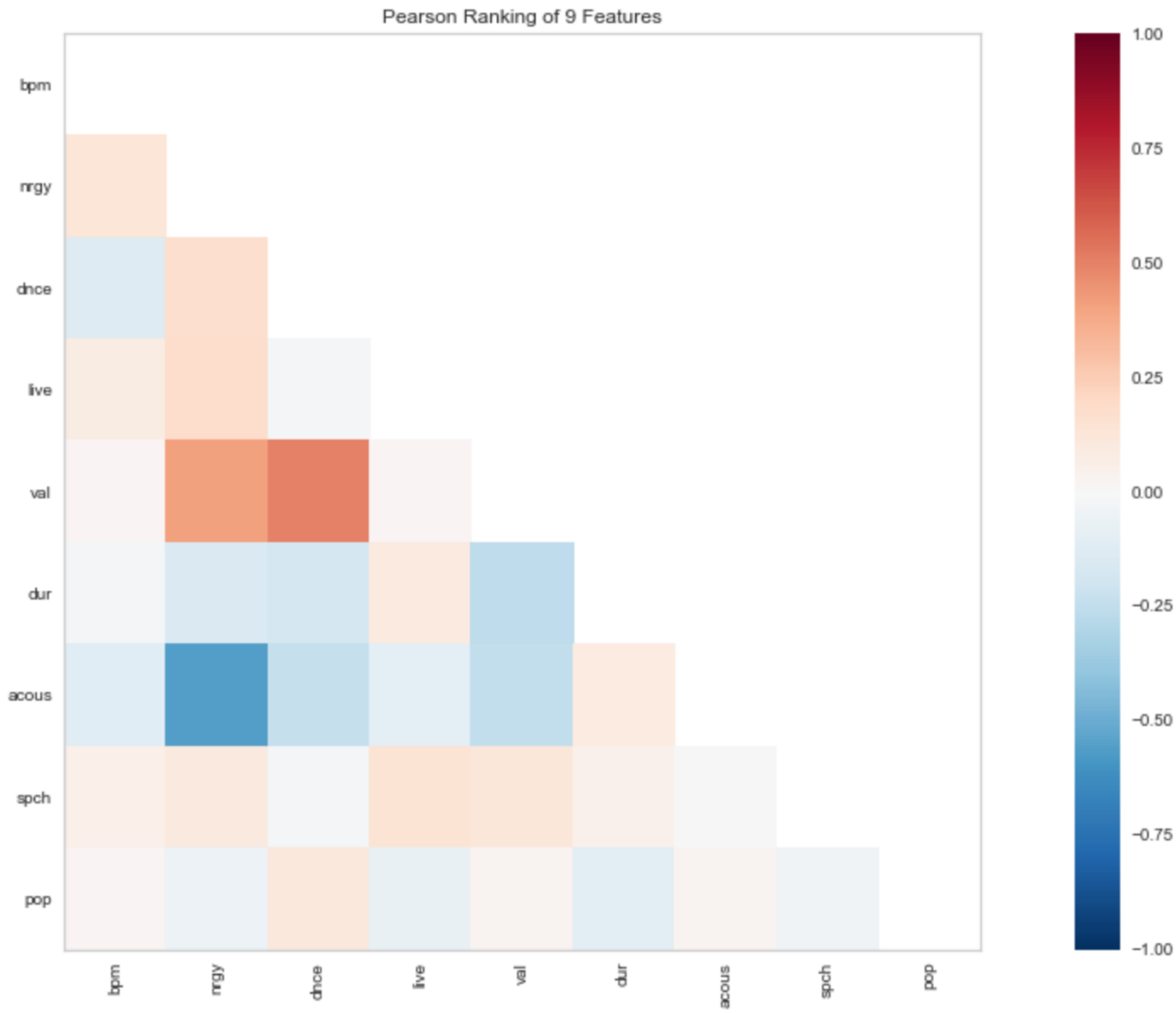
Step 4: Look at the different types of variables and summary information

Step 5: Clean up data (check for missing values, delete columns as needed, etc.)

Step 6: View distribution of variables with histograms



Step 7: Pearson Correlation Ranking Chart

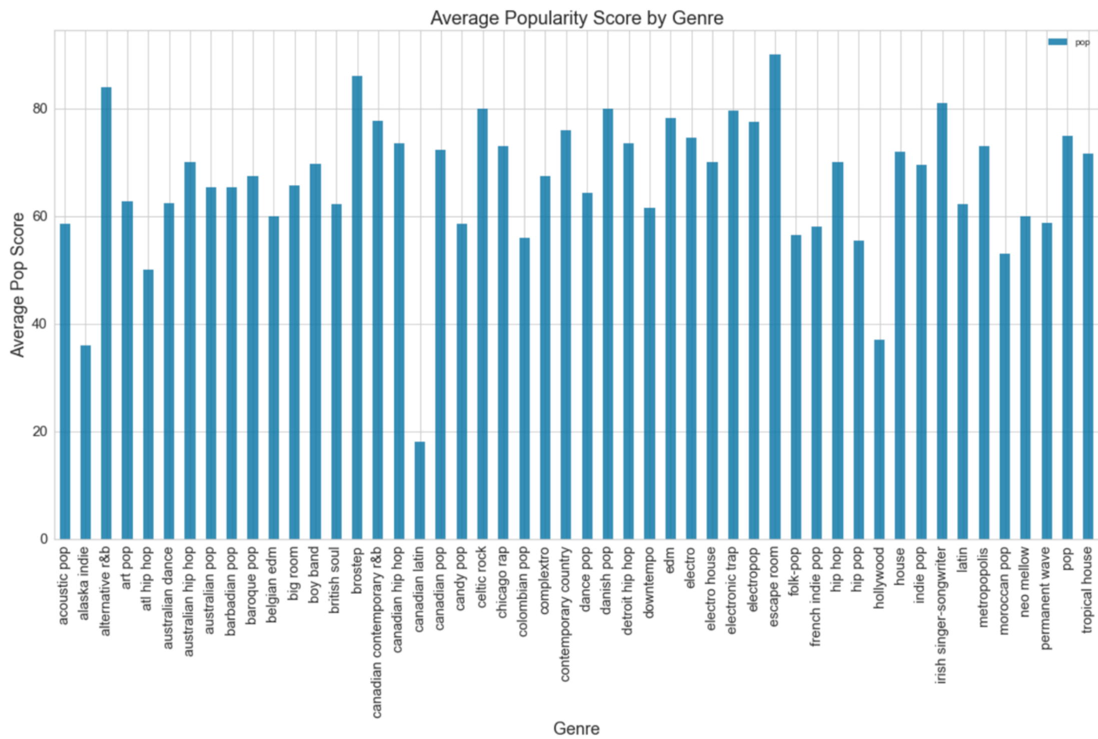


Dnce (danceability) and val (valence) are positively correlated.

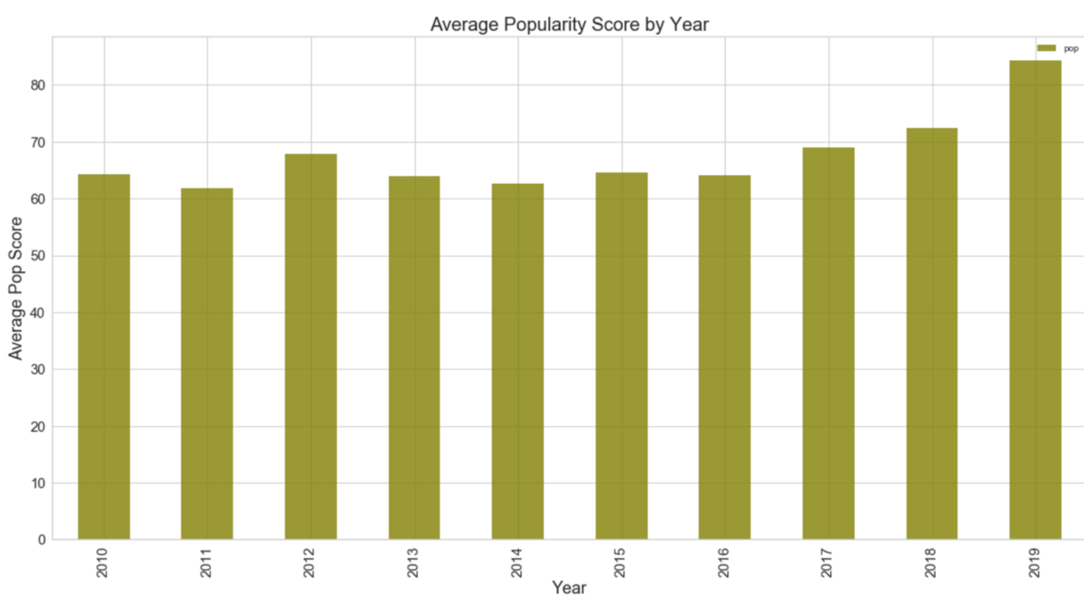
Acous (acousticness) and nrgy (energy) are negatively correlated.

There is no multicollinearity within this dataset.

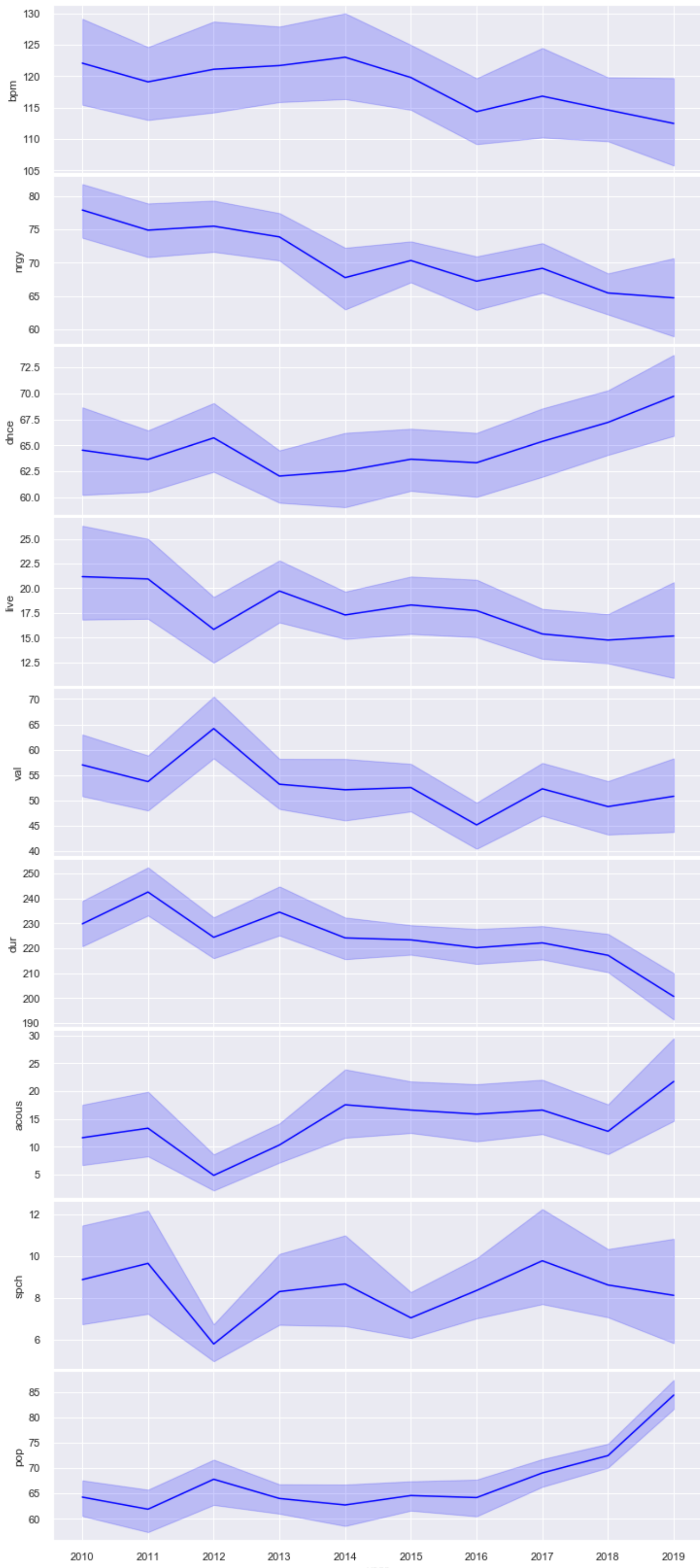
Step 8: Explore some visualizations.



The genre with the highest popularity average is escape room. I have never heard of this genre before!



The year with the highest average popularity score average is 2019.



From this plot I am noticing the following:

1. bpm (tempo) is slightly decreasing over time

2. energy is consistently decreasing over time

3. danceability is consistently increasi ng over time

4. valence (mood) is consistent over time

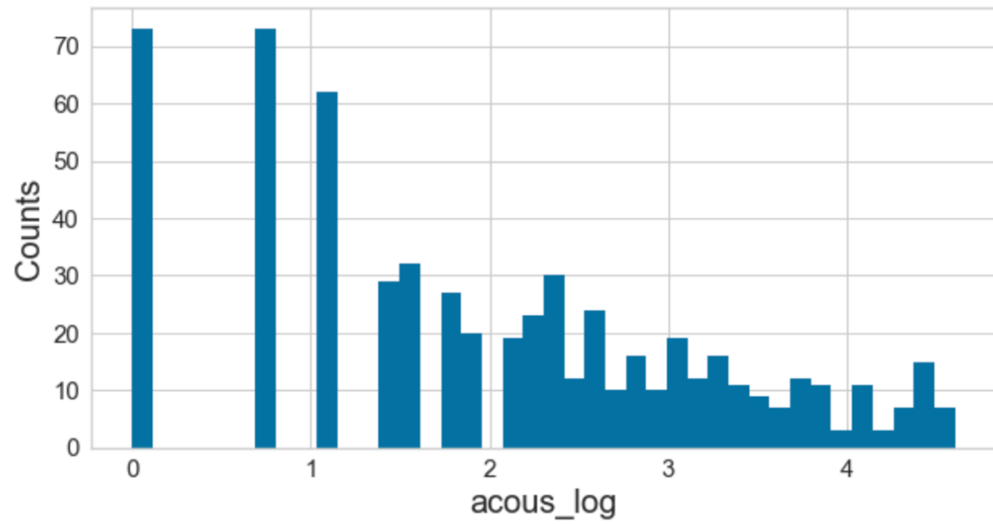
5. duration is consistently decreasing

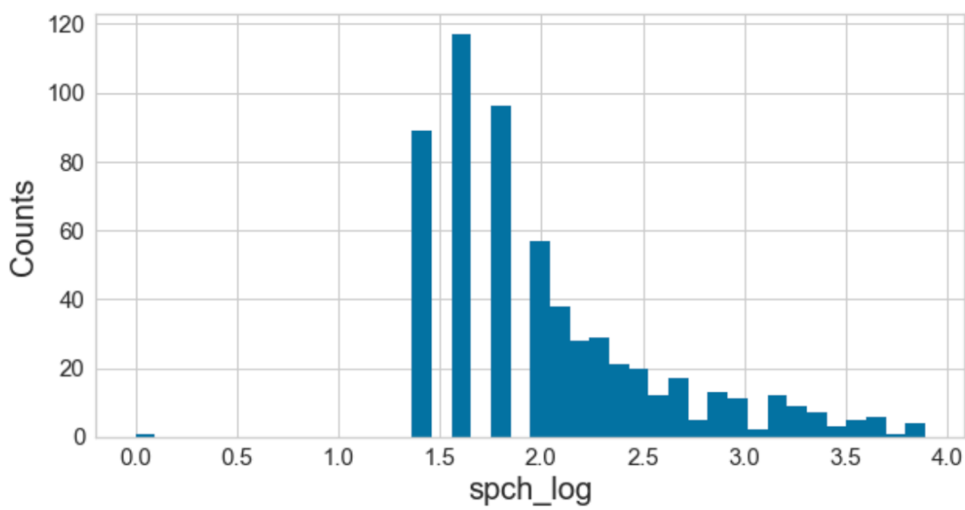
6. acousticness is consistently increas ing over time

5. popularity is consistently increasin g over time

Dimensionality Reduction:

Step 9: Perform log transformation on skewed variables (acous & spch)





Step 10: View the variance of the features.

Several features have high variance which mean the spread in the data is quite large.

year	6.796747
bpm	614.809794
nrgy	266.037773
dnce	178.990088
dB	7.828917
live	171.676622
val	506.836091
dur	1164.860950
acous	431.233621
spch	55.997719
pop	210.764935
acous_log	1.609541
spch_log	0.335319

Step 11: View the correlation to target variable for each feature. Five features are negatively correlated with the popularity score while seven features are positively correlated. None of the correlations are very high.

```
year         0.241261
bpm          0.018983
nrgy        -0.057645
dnce         0.116054
dB           0.156897
live        -0.075749
val          0.038953
dur         -0.104363
acous        0.026704
spch        -0.041490
pop          1.000000
acous_log    0.080561
spch_log    -0.005666
```

Step 12: Feature Selection with SelectKBest

The data now has 7 variables which are shown below.

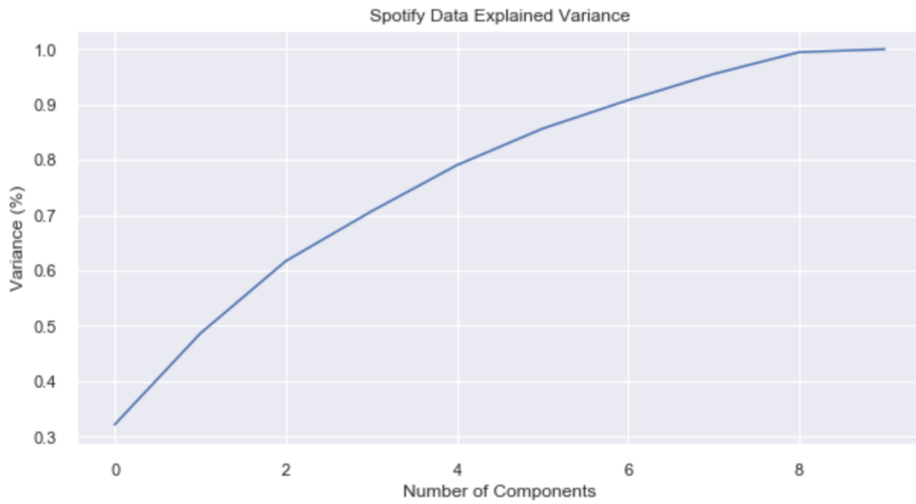
```
[[ 67   8 217]
 [ 75  52 263]
 [ 76  29 200]
 [ 70   8 295]
 [ 64   9 221]]
   spch  bpm  nrgy  dnce  live  val  dur
0     4   97    89    67     8   80  217
1    23   87    93    75    52   64  263
2    14  120    84    76    29   71  200
3     4  119    92    70     8   71  295
4     4  109    84    64     9   43  221

The new shape of the data is:  (603, 7)
```

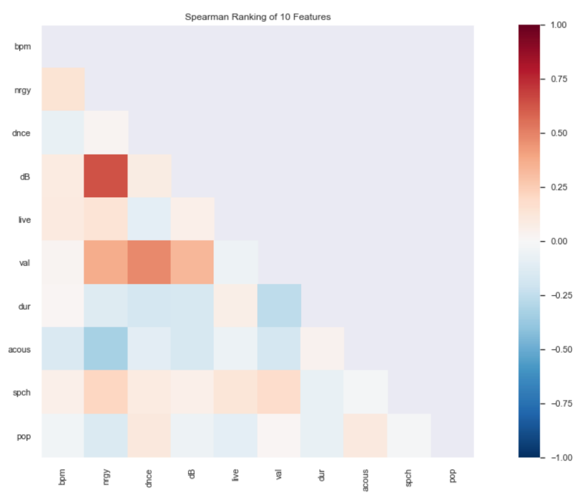
Step 13: Normalize data between 0 and 1 to help with PCA and linear regression

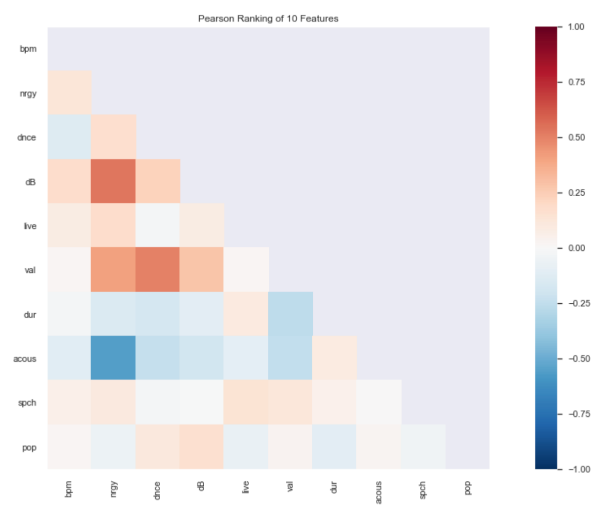
	bpm	nrgy	dnce	dB	live	val	dur	acous	spch	pop
count	603.000000	603.000000	603.000000	603.000000	603.000000	603.000000	603.000000	603.000000	603.000000	603.000000
mean	0.575464	0.719430	0.663709	0.938297	0.240195	0.532914	0.312672	0.144714	0.174129	0.671927
std	0.120366	0.166435	0.137925	0.048242	0.177061	0.229725	0.117690	0.209759	0.155899	0.146644
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.485437	0.622449	0.587629	0.931034	0.121622	0.357143	0.234483	0.020202	0.083333	0.606061
50%	0.582524	0.755102	0.680412	0.948276	0.162162	0.530612	0.300000	0.060606	0.104167	0.696970
75%	0.626214	0.836735	0.752577	0.965517	0.324324	0.704082	0.363793	0.171717	0.187500	0.767677
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

Step 14: View the explained variance. The explained variance increases as the number of components (features) increases. I will use all 8 features.



Step 15: View Spearman and Pearson correlation heatmaps with normalized data frame.

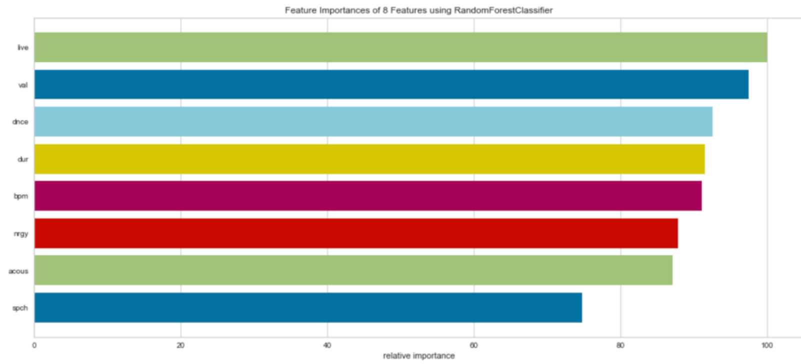




<u>Model Evaluation & Selection</u>

Step 16: Split the data into two sets (training 80% and testing 20%).

Step 17: Create random forest classifier model and view feature importance



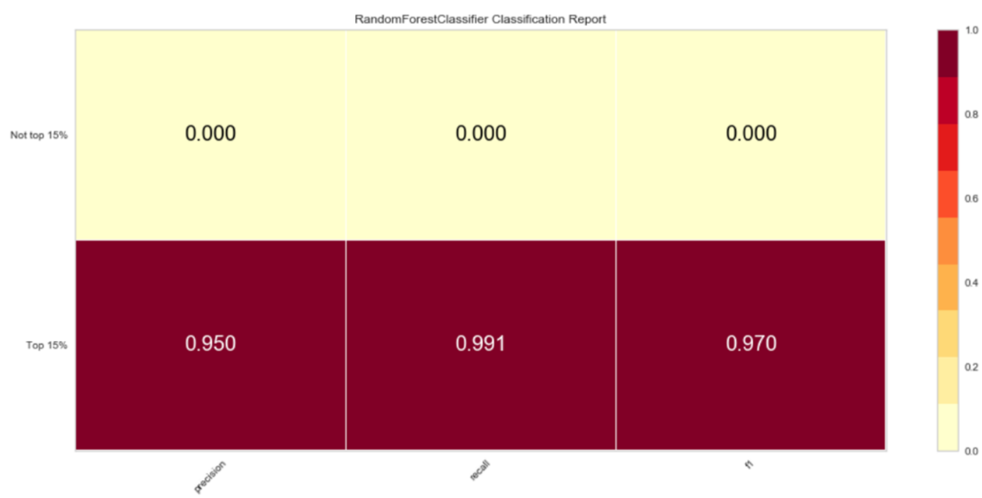
Step 18: Display number of songs in the top 15% of each set and not in the top 15%

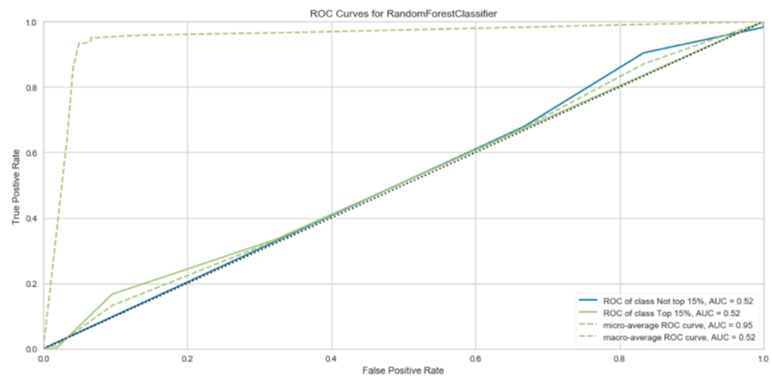
```
No. of samples in training set:  482
No. of samples in testing set: 121


No. of songs in the top 15% of pop in the training set:
Not top 15%    462
Top 15%         20
Name: pop, dtype: int64


No. of songs in the top 15% of pop in the testing set:
Not top 15%    115
Top 15%          6
Name: pop, dtype: int64
```

Step 19: Classification Report and ROC





<u>**Results of 1st Random Forest**</u>:

Precision: 0.950
Recall: 0.991
F1 Score: 0.970

Step 20: Try other models

<u>**Results/Score**</u>

Support Vector Machine: 0.6363636364
Decision Tree: 0.925619834
Random Forest #2: 0.446280992

Step 21: Define and use function that will find similar songs to any randomly chosen song from the dataframe.

Step 22: Linear Regression

<u>**Results of Linear Regression:**</u>

$R^2 = 0.628$

62.8% of the variance of the popularity score can be explained through the features of this dataset.