Building the Perfect Song

By Asher Palmer

The Dataset

I found this dataset while looking for sets on music. I found a subset of the Million Song Dataset. This subset is 10000 songs or 1% of the original dataset. Originally, I did not want to use this dataset as it had many missing values and needed quite a bit of cleaning.

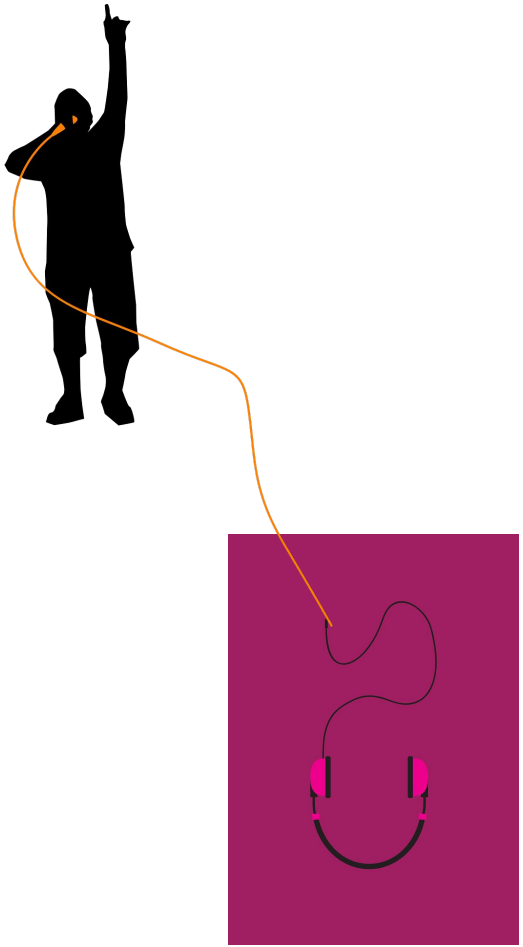
The Music dataset has a number of great attributes such as artist hotness, song hotness, location, latitude & longitude, key the song was made in, the mode, the familiarity, the duration as well as the artist name and song title.

The data ranges from year 0 (these songs did not have input for year) through 2010 and is 10000 rows x 35 columns.

Why Did I Choose This Dataset?

I choose this dataset because I have a very personal connection with music. I've played in band in school and have seen and experienced the way it moves people and can completely change their outlook and provide a safe space.

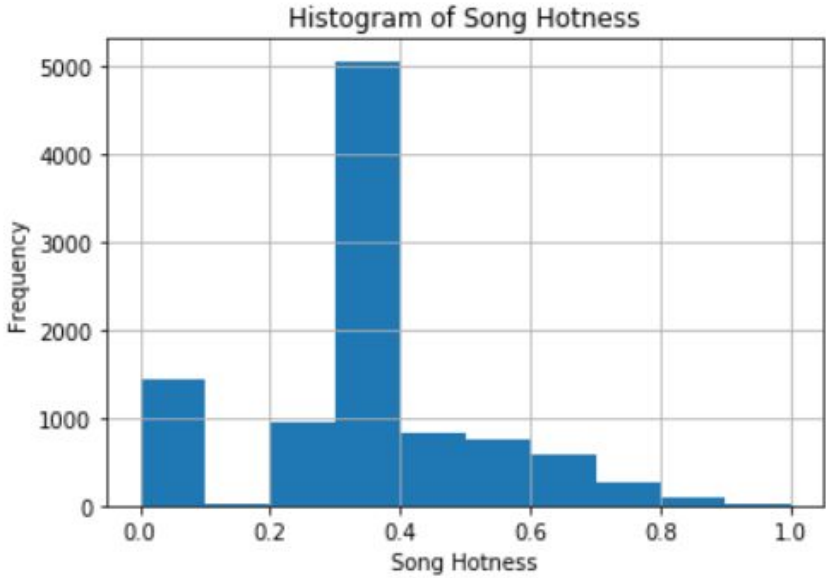
Because music is so influential, it has become a multibillion dollar business and being able to produce the perfect song (particularly multiple songs on a consistent basis) can be very lucrative.



The Parts and Pieces

Many of the features are continuous. One of which, we will focus on is ‘song.hotttnesss’. This variable is how we define a great song.

As you can see with the histogram to the right, the majority of songs have a rating of 0.3. However, we want a song that scores 1.0 consistently.



Attributes Of A Perfect Song

Looking at the data, there are only two songs in my dataset that have a rating for song hotness of 1.

Nothin' On You by B.o.B. made in 2010. The artist hotness is 0.71253495 and plays for 269.6355 seconds. The familiarity is 0.768224 and played in the key of 10 (B flat). The tempo is 104.038 and is -5.388 decibels loud.

Immigrant Song by Led Zeppelin made in 1970. The artist hotness is 0.63441239 and plays for 145.0575 seconds. The familiarity is 0.787098 and played in the key of 11(B). The tempo is 150.569 and is -10.544 decibels loud.

The Eye Ball Comparison

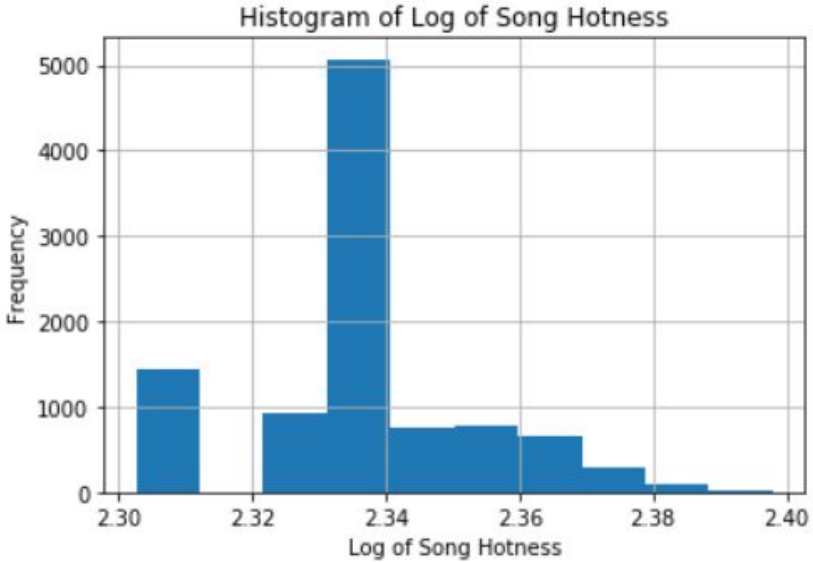
Just looking at these comparisons, you would think that there isn't much between these two songs other than a similar familiarity and key that they are played in. They are played in different countries and have so many more dissimilarities that I'm not sure data science will find a way to find the perfect song.

Lucky for us, we have technology to analyze data and be able to make predictions about future songs.

Normalizing the Data

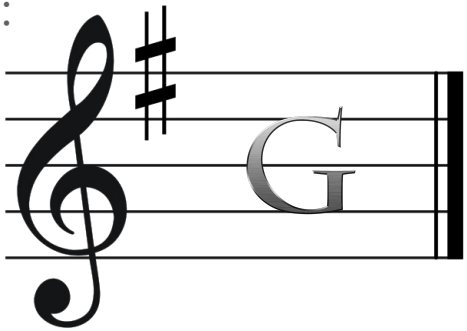
I thought about using a log version of my target variable as a means of making the distribution more normal but all I got was essentially the same distribution, which is close to normal. That's why I decided to use the normal version of my variable.

I did decide to use a variable in my classifications called 'song_hotttnesss' that split my data into songs with a hotness of 0.5 or greater.

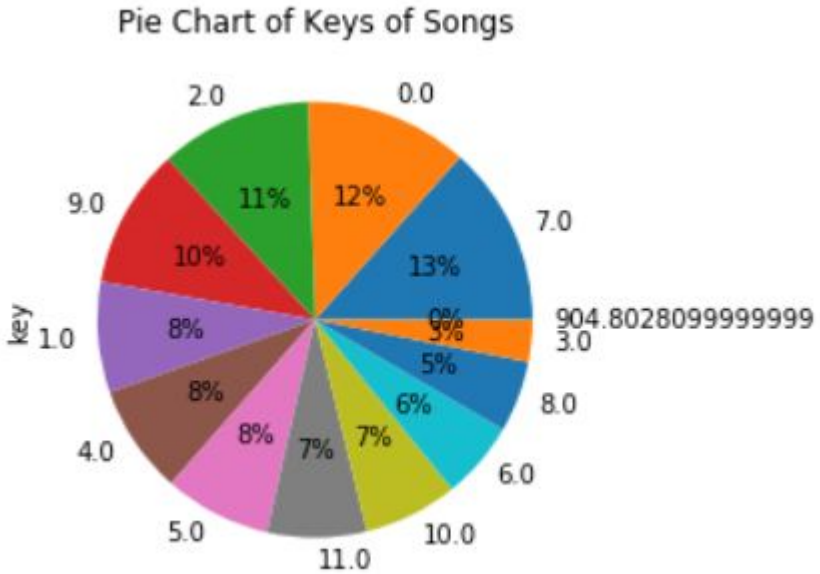


The Frequently Used Key

The key that is used the most is:



Followed by; C, D, & A



Why Is G The Most Used Key?

According to Spotify, “More mysterious, at first glance anyway, is the order of the notes. Why is G Major the top key on all of Spotify? And why is C Major number two?

Much like electricity going through a circuit, songwriters often take the path of least resistance. On a keyboard or a guitar — both incredibly popular instruments for composing western music — that path is through G Major.”

Put simply:

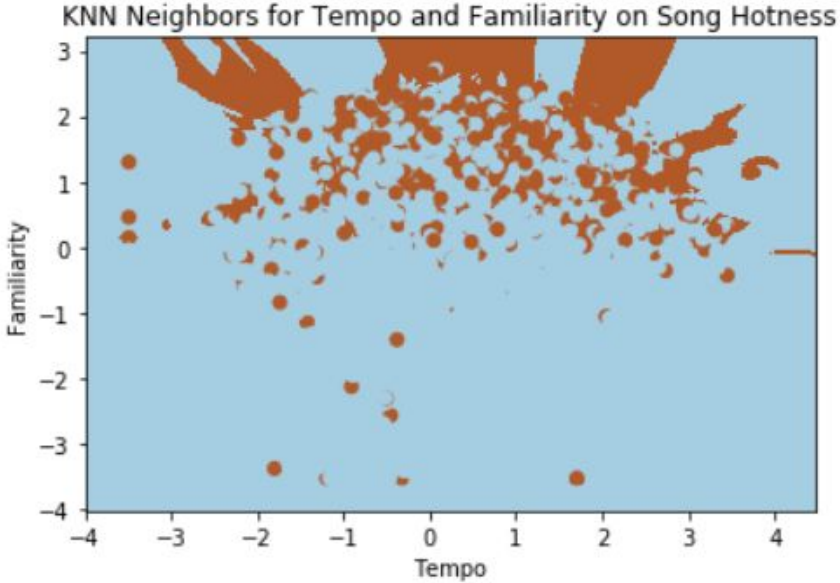
E is convenient for guitar, but not piano.

C is convenient for piano, but not guitar.

G is convenient for both guitar and piano.

Let the Testing Begin

Let’s take a look at KNN. Looking at Tempo and Familiarity, I wanted to see if there were any clear and decisive lines of what to call a “good song”. As you can see, there’s no real correlation between tempo and song hotness but the higher the familiarity, the greater the occurrence of song hotness.



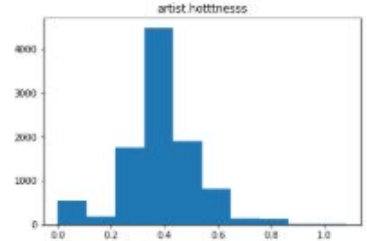
Random Forest

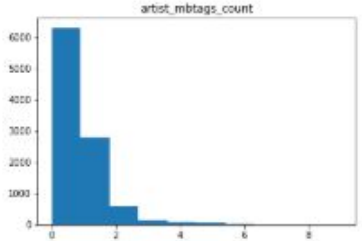
Using Random Forest, my model performs at around 81% and took 32 seconds.

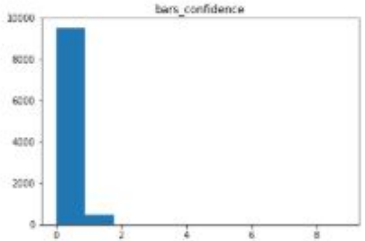
```
Depth: 1
Model Performance: 82.87%.

Depth: 2
Model Performance: 82.87%.

Depth: 3
Model Performance: 82.87%.

Depth: 4
Model Performance: 82.79%.

Depth: 5
Model Performance: 82.61%.

Depth: 6
Model Performance: 82.23%.

Depth: 7
Model Performance: 81.87%.

Depth: 8
Model Performance: 81.55%.

Depth: 9
Model Performance: 80.92%.

dot: graph is too large for cairo-renderer bitmaps. Scaling by 0.96399 to fit

Depth: 10
Model Performance: 80.20%.

--- Runtime: 32.1027059 seconds. ---
```

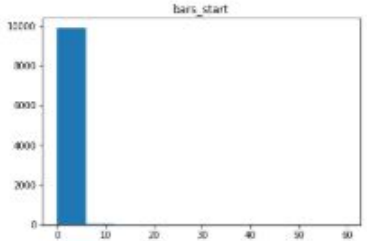
Subplots

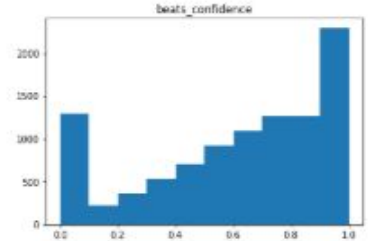
On the following slide are the subplots of my numeric variables. The features have mostly normal distributions but there are some variables that are skewed. I did consider winsorizing these skewed variables in an attempt to normalize them. However, I decided not to as a means of “preserving” the original data.

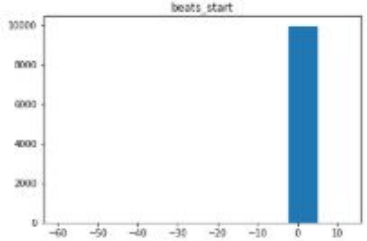

artist.hotttnesss

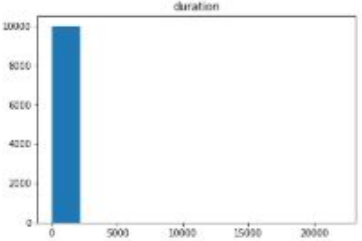

artist_mbtags_count

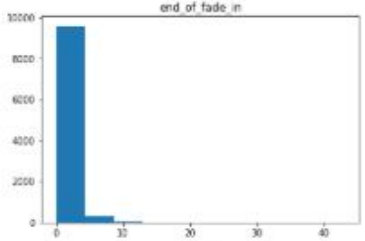

bars_confidence

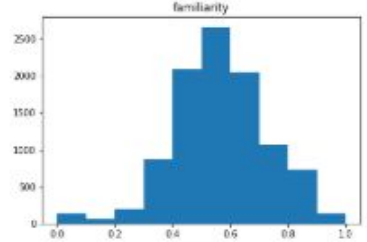

bars_start

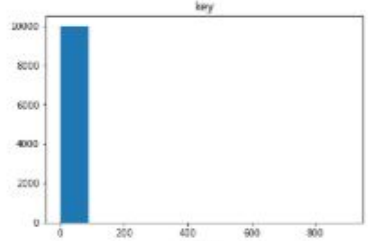

beats_confidence

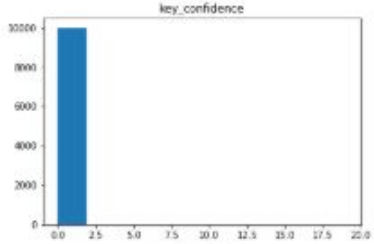

beats_start

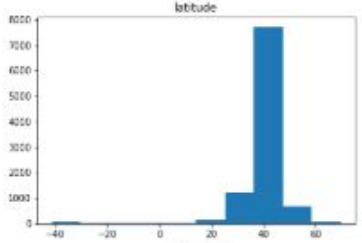

duration

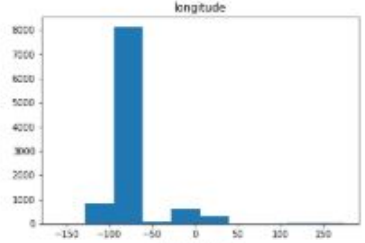

end_of_fade_in

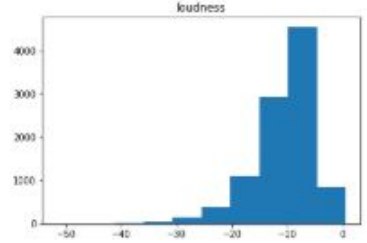

familiarity

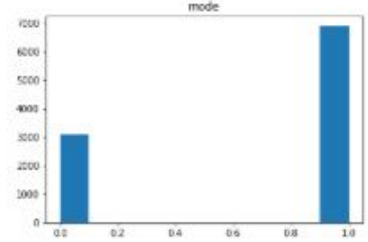

key

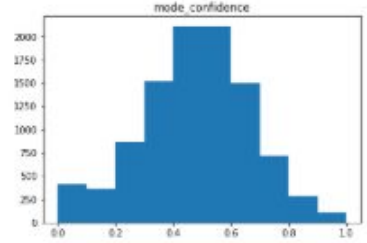

key_confidence

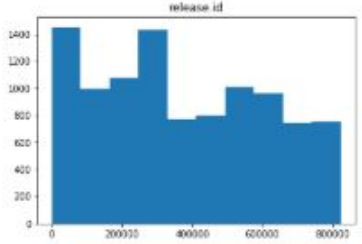

latitude

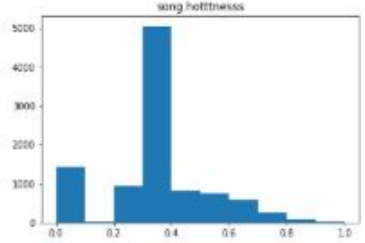

longitude


loudness


mode


mode_confidence


release id


song.hotttnesss

Preprocessing

Here are the preprocessing numbers for those who are interested.

	0	1	2	3	4	5	6	\
0	-0.568257	-1.200206	-1.818560	-0.766981	1.205503	0.121417	0.006727	
1	0.397776	1.249813	-0.058316	-0.960125	0.983171	-1.380964	-1.404991	
2	0.232170	1.064861	0.451228	-0.166727	-0.331387	1.464795	1.209015	
3	-1.061513	-1.677932	-0.340460	-0.704076	-1.353145	0.161600	0.867720	
4	1.515170	-0.502899	0.451228	0.188552	1.111954	-0.892179	0.852207	

	7
0	-0.463082
1	0.946452
2	-1.566491
3	0.928234
4	-0.834219

Cross-Validation Scores

Here are my cross-validation scores. My model has an average of 83.40% and features of importance showing as artist hotness, familiarity and the year.

Also with Gradient Boosting, you can see my table with my classifier.

```
Cross-Validation Scores: [0.83316683 0.83483483]

Cross-Validation Score Averaged Across Folds: 83.40%.

Selected Features: ['artist.hotttnesss', 'familiarity', 'year']
```

col_0	0	1	All
song_hotttnesss			
0	6484	136	6620
1	700	680	1380
All	7184	816	8000

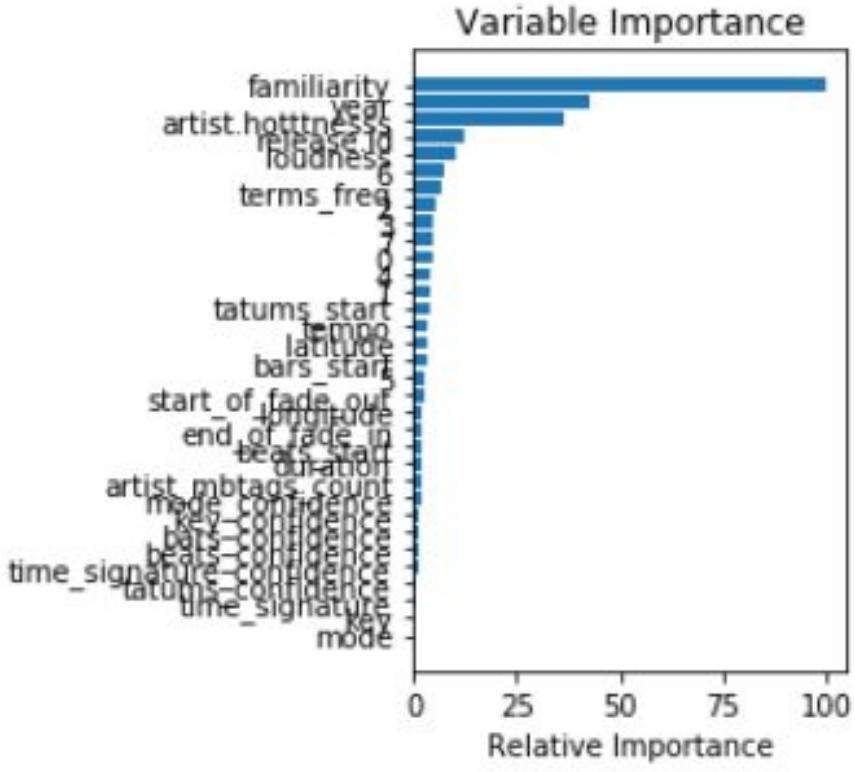
Training Set Accuracy

Here are my training and test set accuracies. This state that my Type I and Type II errors are mostly low.

```
Training set accuracy:
Percent Type I errors: 0.017
Percent Type II errors: 0.0875

Test set accuracy:
Percent Type I errors: 0.0435
Percent Type II errors: 0.1145
```

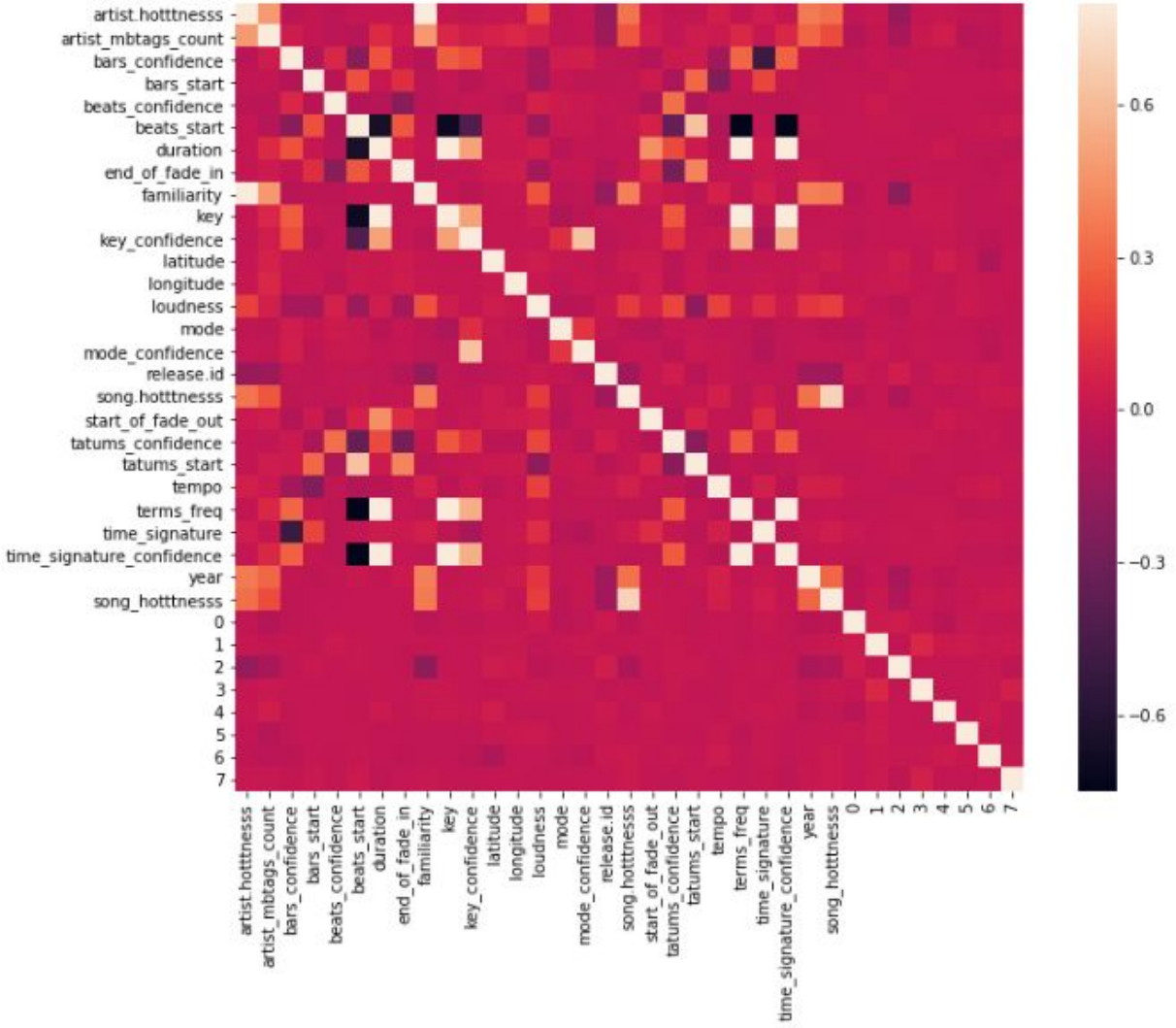
Variable Importance

As you can see, familiarity, year and artist.hotttnesss are the top three important features. I thought the duration and key of a song would be more influential on song hotness.



Heatmap

On the next slide, you will find the heatmap. It’s filled with over 40 features so it’s hard to find song.hotttnesss and follow along the axis. However, song.hotttnesss is highly correlated with artist.hotttnesss, familiarity, and year with some slight correlation to loudness. This is all shown with the variable importance image from the last slide but with more pizzazz.


artist.hotttnesss
artist_mbtags_count
bars_confidence
bars_start
beats_confidence
beats_start
duration
end_of_fade_in
familiarity
key
key_confidence
latitude
longitude
loudness
mode
mode_confidence
release.id
song.hotttnesss
start_of_fade_out
tatums_confidence
tatums_start
tempo
terms_freq
time_signature
time_signature_confidence
year
song_hotttnesss
0
1
2
3
4
5
6
7
0.6
0.3
0.0
−0.3
−0.6

StatsModel Regression

I have included a statsmodel regression for those that would like a closer look at the numbers and how I am forming my hypothesis.

```
Optimization terminated successfully.
         Current function value: 0.357650
         Iterations 19
                           Logit Regression Results
==============================================================================
Dep. Variable:        song_hotttnesss   No. Observations:                 8000
Model:                          Logit   Df Residuals:                     7966
Method:                           MLE   Df Model:                           33
Date:                Tue, 03 Sep 2019   Pseudo R-squ.:                 0.2222
Time:                        22:47:04   Log-Likelihood:                -2861.2
converged:                       True   LL-Null:                       -3678.6
                                        LLR p-value:                     0.000
============================================================================================
                               coef    std err          z      P>|z|      [0.025      0.975]
--------------------------------------------------------------------------------------------
artist.hotttnesss            1.1166      0.465      2.402      0.016       0.206       2.028
artist_mbtags_count         -0.0900      0.038     -2.377      0.017      -0.164      -0.016
bars_confidence              0.0122      0.148      0.083      0.934      -0.278       0.302
bars_start                   0.0227      0.027      0.851      0.395      -0.030       0.075
beats_confidence            -0.0656      0.112     -0.585      0.559      -0.285       0.154
beats_start                 -0.1873      0.236     -0.792      0.428      -0.651       0.276
duration                     0.0203      0.005      3.881      0.000       0.010       0.030
end_of_fade_in              -0.0159      0.021     -0.757      0.449      -0.057       0.025
familiarity                  5.4545      0.448     12.177      0.000       4.577       6.332
key                         -0.0088      0.010     -0.920      0.357      -0.028       0.010
key_confidence               0.0465      0.193      0.240      0.810      -0.333       0.425
latitude                     0.0087      0.004      2.266      0.023       0.001       0.016
longitude                   -0.0008      0.001     -0.730      0.466      -0.003       0.001
loudness                     0.0536      0.008      6.796      0.000       0.038       0.069
mode                        -0.1169      0.073     -1.591      0.112      -0.261       0.027
mode_confidence              0.2164      0.281      0.770      0.441      -0.334       0.767
release.id             -5.273e-07   1.47e-07     -3.591      0.000   -8.15e-07   -2.4e-07
start_of_fade_out           -0.0209      0.005     -3.925      0.000      -0.031      -0.010
tatums_confidence            0.0537      0.120      0.448      0.654      -0.181       0.288
tatums_start                 0.2321      0.243      0.956      0.339      -0.244       0.708
tempo                        0.0017      0.001      1.607      0.108      -0.000       0.004
terms_freq                  -2.1828      0.663     -3.294      0.001      -3.482      -0.884
time_signature               0.0345      0.035      0.996      0.319      -0.033       0.102
time_signature_confidence    0.0077      0.097      0.079      0.937      -0.183       0.198
year                         0.0006   3.94e-05     15.564      0.000       0.001       0.001
0                           -0.0755      0.034     -2.250      0.024      -0.141      -0.010
1                            0.0454      0.033      1.366      0.172      -0.020       0.111
2                            0.0154      0.032      0.482      0.629      -0.047       0.078
3                            0.0053      0.034      0.158      0.875      -0.061       0.072
4                           -0.0385      0.033     -1.155      0.248      -0.104       0.027
5                           -0.0080      0.034     -0.238      0.812      -0.074       0.058
6                            0.0610      0.034      1.810      0.070      -0.005       0.127
7                            0.0736      0.033      2.202      0.028       0.008       0.139
intercept                   -4.0590      0.738     -5.496      0.000      -5.506      -2.612
============================================================================================
```

Areas of Improvement

1. I could add more features to test different theories on song hotness
2. I could do more normalizing techniques for all of my variables
3. I could find more ways to reduce my bias towards the data

References

The Most Popular Keys of All Music on Spotify

https://insights.spotify.com/us/2015/05/06/most-popular-keys-on-spotify/