598: Machine Learning Project

Application of Classification Methods on Spooftify Data

Antonio Campbell, Sinta Sulistyo, Atta Ullah, Penny Wu

4/30/2021

1 Introduction

In this project we have considered a dataset of 2017 songs from Spotify. This data was provided by Spotify and posted by user GeorgeMcIntire on Kaggle ("https://www.kaggle.com/geomack/spotifyclassification"). Within the data, each song has 16 features. The feature of interest for classification is called `target` which indicates whether the creator of the dataset liked or disliked a song. A song is labeled "1" if it is liked and "0" when it is disliked. The other features include `acousticness`, `danceability`, `durationMs (duration in milliseconds)`, `energy`, `instrumentalness`, `key`, `liveness`, `loudness`, `mode`, `speechiness`, `tempo`, `time-signature`, `valence`, `songname`, `artist`.

The goal of the project is to build several classifiers for prediction that is based on the rest of the features to determine whether the individual would like a song. We begin by preparing the data to fit possible models appropriately.

2 Data exploration and feature selection:

Data Sample

X	acousticness	...	target	song_title	artist
0	0.0102	...	1	Mask Off	Future
1	0.199	...	1	Redbone	Childish Gambino
2	0.0344	...	1	Xanny Family	Future
3	0.604	...	1	Master Of None	Beach House
4	0.18	...	1	Parallel Lines	Junior Boys
5	0.00479	...	1	Sneakin'	Drake
⋮	⋮	⋮	⋮	⋮	⋮
2011	0.000586	...	0	Brightside - Borgeous Remix	Icona Pop
2012	0.00106	...	0	Like A Bitch - Kill The Noise Remix	Kill The Noise
2013	0.0877	...	0	Candy	Dillon Francis
2014	0.00857	...	0	Habit - Dack Janiels & Wenzday Remix	Rain Man
2015	0.00164	...	0	First Contact	Twin Moons
2016	0.00281	...	0	I Wanna Get Better	Bleachers

Data Preparation

We ensure the data is adequate for fitting before we begin modeling. In particular we are looking for features which have missing values, the songs that are duplicated in the dataset, the types of features (if a feature is numerical or categorical), and as well as the variables of importance for the fit.

Data Dimensions

	rows	columns
count	2017	17

We drop the first column, which is just for indexing and has no use to us here. The data has the remaining features

```
##  [1] "acousticness"     "danceability"     "duration_ms"      "energy"
##  [5] "instrumentalness" "key"              "liveness"         "loudness"
##  [9] "mode"             "speechiness"      "tempo"            "time_signature"
## [13] "valence"          "target"           "song_title"       "artist"
```

Next we check if there are any missing values within the data and we find that there are 0 records with missing data in them; however, there are 5 duplicate records:

	acousticness	danceability	...	song_title	artist
268	0.096200	0.654	...	River	Ibeyi
509	0.024600	0.586	...	Her Fantasy	Matthew Dear
895	0.000334	0.907	...	Jack	Breach
928	0.934000	0.440	...	Episode I - Duel of The Fates	John Williams
982	0.036900	0.448	...	Myth	Beach House

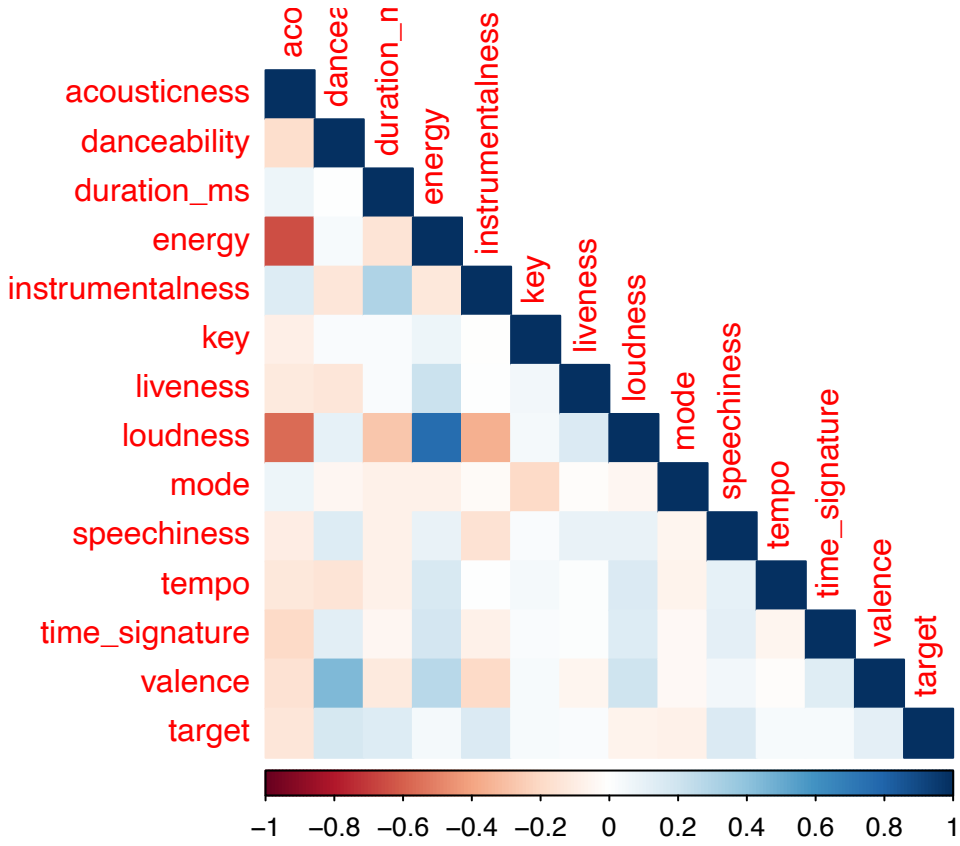
We remove the 5 remaining data points the data that remains has the following dimensions,

Data Dimensions

	rows	columns
count	2012	16

Looking for Correlations (Correlation Matrix Heatmap):

For this analysis we have drooped the last two features 'song_title' and 'artist_name' as well.



Looking for correlation of other feautre with 'target':

```
##
## Correlation method: 'pearson'
## Missing treated using: 'pairwise.complete.obs'
```

3 Features selection based on correlations with 'target':

acousticness

danceability

duration_ms

instrumentalness

speechiness

valence

```
##   target acousticness danceability duration_ms instrumentalness speechiness
## 1      1      0.01020        0.833      204600          0.021900      0.4310
## 2      1      0.19900        0.743      326933          0.006110      0.0794
## 3      1      0.03440        0.838      185707          0.000234      0.2890
## 4      1      0.60400        0.494      199413          0.510000      0.0261
## 5      1      0.18000        0.678      392893          0.512000      0.0694
## 6      1      0.00479        0.804      251333          0.000000      0.1850
##   valence
```

```
## 1   0.286
## 2   0.588
## 3   0.173
## 4   0.230
## 5   0.904
## 6   0.264
```

```
## [1] 7
```

4 Models

The potential methods to build a calcification model for this project include:

Logistic Regression (Penny)

Naive Bayes (Sinta)

Decision Trees (Atta)

Random Forests (Sinta)

Bossting (Penny)

Test Train split

```
## dimension of train data: 1509 7
```

```
## dimension of test data: 503 7
```

```
##    predict_unseen
##       0   1
##   0 166  95
##   1  64 178
```

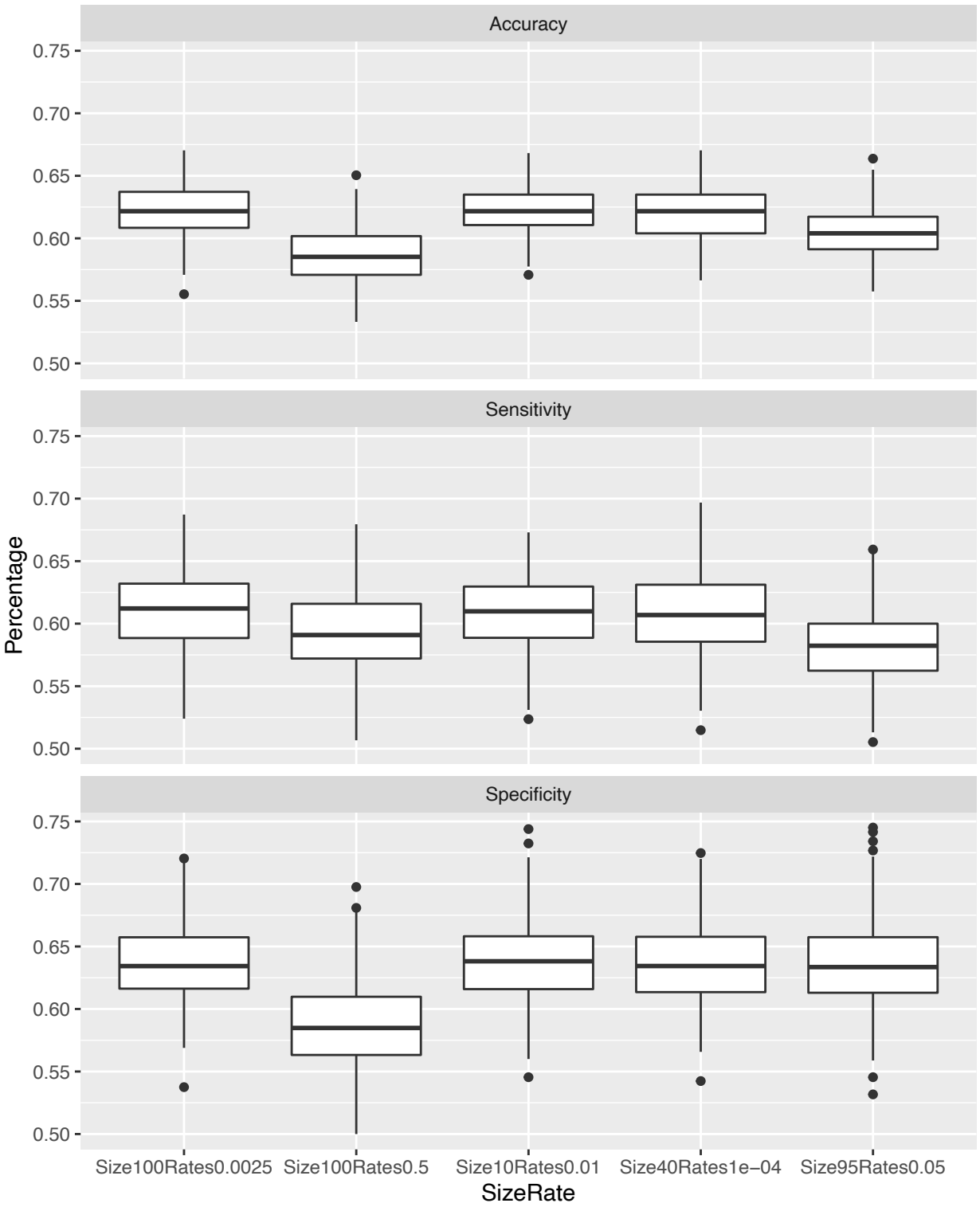
```
## [1] "Accuracy for test data is:  0.68389662027833"
```

Neural Nets

to do: explanations coming soon, commenting code

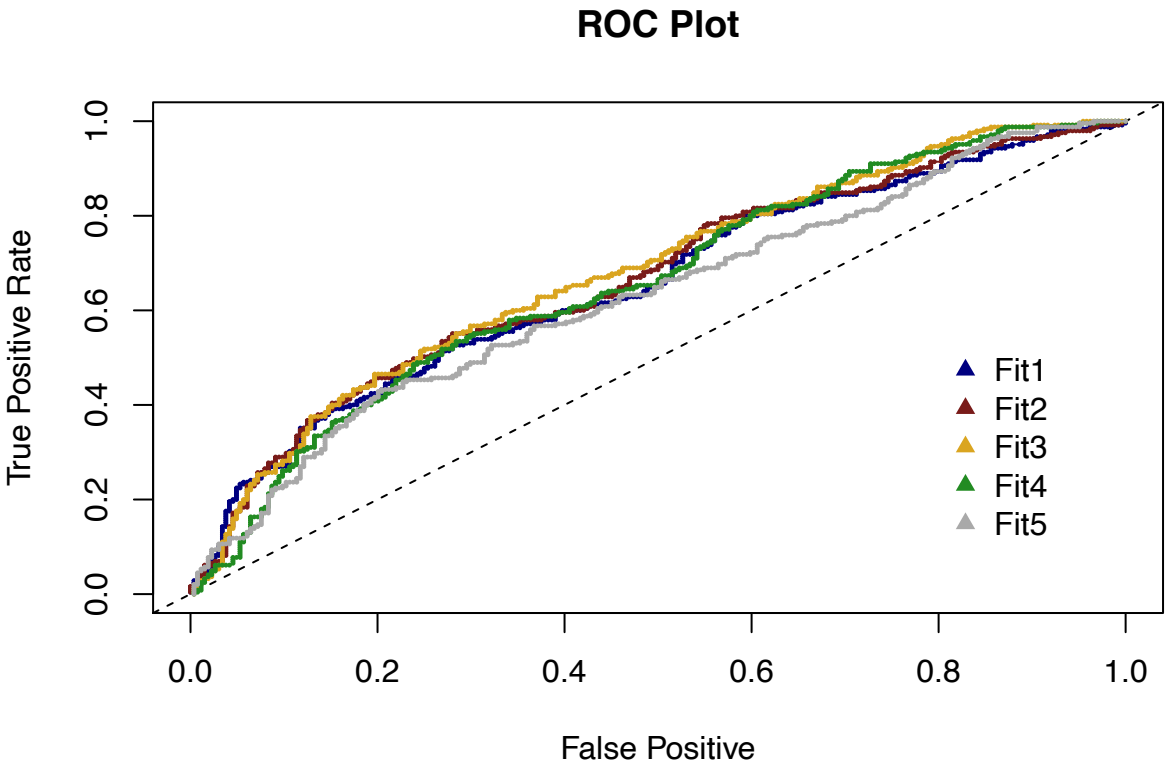
We begin with the simplest model we can fit based on the three main predictors that we have used for the previous models, `danceability`, `speachiness`, `acousticness`. We will use a single-layer neural network to fit on the training data. For optimization we will begin by using a grid search accross of layer size. We also want to introduce regularization so we will include those in our grid as well.

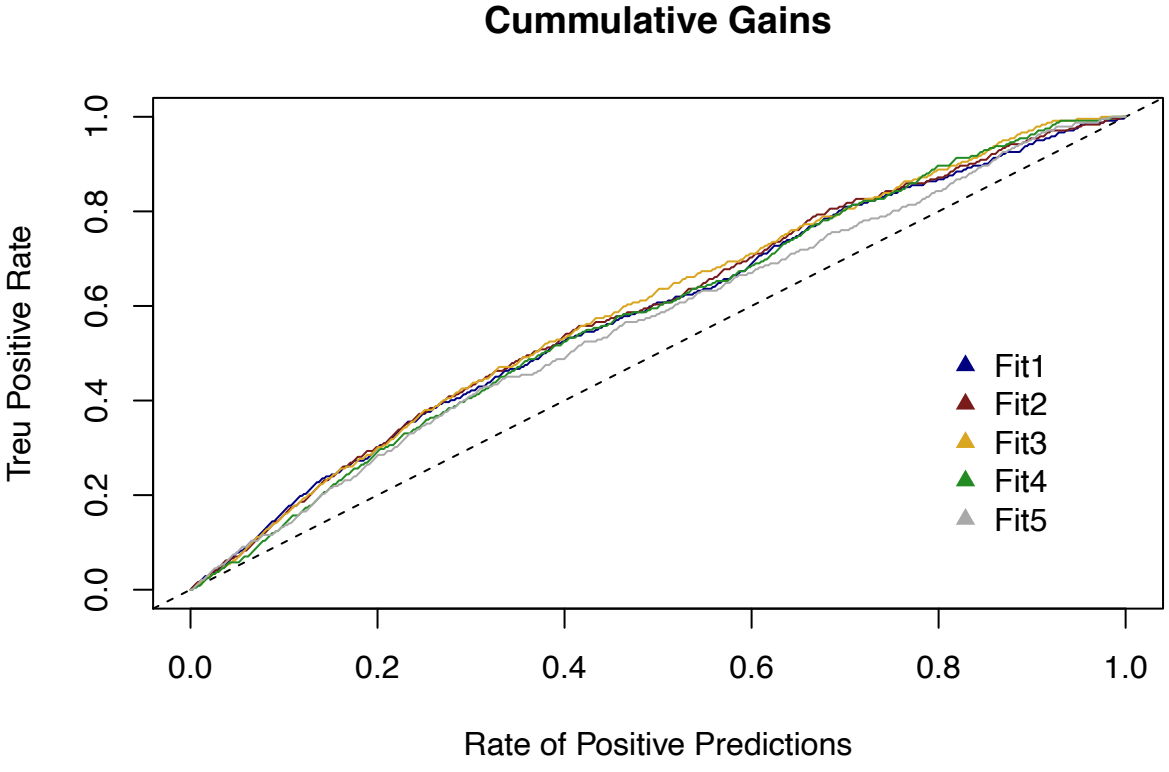
```
##     size  decay
## 1      5 0.0001
## 2     10 0.0001
## 3     15 0.0001
## 4     20 0.0001
## 5     25 0.0001
## 6     30 0.0001
## 7     35 0.0001
```



explain how biased fits are as one of the 'best fits' of a single split doesn't operate the same for splits. other splits

explain why spotify might care more about specificity vs sensitivity. If the person is a music lover in general who dooens't stick to one genre or music style they care way more about Specificity because the use probably likes more than they don't like so finding the don't likes are more important vs. somoneone who has limited music taste. Getting the likes categorized are the most important.





to do: Choose different cut off for classification than 0.5 and explain why spotify might do this... based on specificty/sensitivity. We want to reduce fpr, better to make a playlist of more good songs that letting in more songs they wouldn't like. Minimize fpr on sensitivity vs. specificity. Refit using new classification rule. pick best fit from these and show confusion matrix, show accuracy.

To do: fit NN on all predictors and smaller grid of sizes/rates...try regularization decay rates = c(0.001, 0.01, 0.05, 0.1, 0.25) sizes = c(5,15,25,75). Say/show something about why regularization should provide similar fit to above. (Consider fitting model over several splits again again on different models and view box plot?? might take too long if I use more thatn 100 again..)

ROCs/Gains/Classificiation rule/Confusion matrix on best

Add cost functions for fits as well???

Confusioon matrix plots??

Deep Neural Nets (Antonio)

play with 2-3 layers with smallest ammount of predictors. If worse than 1 layer neural net stop there.

Summary oof metrics of thre

5 Conclusions