Group Assignment 4 - Alternative Model - Spotify2

TASK

You summary should include the following information:

- Justify your model choice based on how your response is measured, any observations you may have made in your EDA, and the first model you have estimated. When selecting an alternative model, choose one that is appropriate for your response. Do **not** change the response to estimate an alternative model.
- Report the model's test error rate using one of the techniques we discussed in lecture. Justify your choice.
- Based on the estimated test error rate, discuss how well the model fits the data.
- Use the model to make predictions for at least three cases of interest.

INTRODUCTION

The response variable, popularity, is distributed nearly evenly across the popularity index, but there are a maximum number of songs that are unpopular, or with a popularity index of zero. This is illustrated by the frequency distribution in the histogram plot of the response variable.

Previously, a logistic regression model was implemented to fit the apparent non-linear relationship in the data set. However, it was observed that the r-squared value was too low for the model to be considered a decent fit, at r-squared = 0.4.

The random forest model was selected as a suitable alternative and was expected to produce better results and better predictions. Random forests are known for their robustness and ability to fit non-linear data better than some others. Additionally, the data set used in this study contains both numerical and categorical features, and random forests are known to be better in such cases. Lastly, the random forest was chosen for the ease in interpretability.

MODEL SUMMARY: RANDOM FOREST

- randomForest library was used to fit the model.
- Total number of predictors = 10, and since this is for regression and not classification, m = sqrt(p) was used, giving m = 3.
- Number of trees was left at default, ntree = 500.

The table below summarizes the parameters and their values used in this study.

Parameter	**Value**
Sample size	10,000
Predictors	10
mtry	3
ntree	500 (default)
Training set	7500
Testing set	2500

DISCUSSION

The percent of variability explained by the models (repeated 10 times) is, on average 56%. This is an improvement from the logistic regression model previously built.

For sample size = 10,000:

%Var	**MSE Test**	**MSR Train**
56.14	195.12	204.89
55.93	203.13	206.51
55.34	216.09	207.3
58.08	196.17	198.55
54.82	213.63	209.10
57.64	205.55	193.35
55.28	206.22	206.69
56.5	197.30	204.44
54.46	191.99	212.99
55.84	201.13	208.64

For sample size = 20,000:

%Var	MSE Test	MSR Train
56.95	201.2415	204.0528
56.2	205.5447	202.4685
57.28	212.8946	202.2643
55.01	200.5142	212.4289
53.92	207.5451	212.5884
54.55	205.948	207.1142
53.95	206.8427	212.9464
55.81	200.0225	205.5912
54.95	205.7717	214.4911
56.4	207.2242	205.0833

PREDICTIONS

Case 1

The following values have been chosen as the "higher" threshold for all variable ranges. The popularity index, as was expected, is low at 13.596. This is expected because some of the variables are negatively correlated with popularity. As values in such variables increase, popularity decreases.

Acousticness = 0.99
Danceability = 0.99
Instrumentalness = 0.99
Key = 7
Speechiness = 0.99
Loudness = -1
Liveness = 0.99
Duration_ms = 180000
Tempo = 100

Popularity = 13.596

Case 2

The following values were chosen as the "lower" threshold, where the variables are given lower values within their range. This case predicted a popularity index of 33.407. This is reasonable because of the aforementioned reason that some predictors are correlated with higher popularity by having lower values.

Acousticness = 0.1
Danceability = 0.15
Instrumentalness = 0.15
Key = 7
Speechiness = 0.15
Loudness = -55
Liveness = 0.15
Duration_ms = 180000
Tempo = 55

Popularity = 33.407

Case 3

The following values were chosen selectively. Based on the EDA, predictors that were observed to have no correlation at all with popularity were given "middle-ground" values. Predictors that were somewhat skewed were given values on the end of the spectrum where they correlated with high popularity. However, surprisingly, the "middle-ground" values seem to have affected the song's audio features in such a way that predictors with values that correlated with higher popularity indexes were overshadowed, therefore concluding with a low popularity index.

Acousticness = 0.5
Danceability = 0.5
Instrumentalness = 0.85
Key = 7
Speechiness = 0.3
Loudness = -10
Liveness = 0.2
Duration_ms = 180000
Tempo = 30

Popularity = 28.621

```
> predict(rf1,newdata )
     1
13.596
> predict(rf1,newdata1 )
       1
33.40703
> predict(rf1,newdata2 )
      1
28.6217
>
```

CONCLUSION

This model did much better than the previous logistic regression model. However, if the entire data set were trained and tested, it is predicted that the results would be better.

Additional to random forests, the support vector machine model was also implemented. However, the random forest demonstrated better results.

BIBLIOGRAPHY

R

https://rstudio-pubs-static.s3.amazonaws.com/71575_4068e2e6dc3d46a785ad7886426c37db.html
https://www.rdocumentation.org/packages/rfUtilities/versions/2.1-5/topics/rf.crossValidation
https://www.blopig.com/blog/2017/04/a-very-basic-introduction-to-random-forests-using-r/
https://www.r-bloggers.com/2018/01/how-to-implement-random-forests-in-r/
https://stackoverflow.com/questions/13956435/setting-values-for-ntree-and-mtry-for-random-forest-regression-model#:~:text=The%20randomForest%20function%20of%20course,of%20500%20quite%20a%20bit.
https://sebastiansauer.github.io/Three_ways_recoding_cutting/
https://www.rdocumentation.org/packages/caret/versions/6.0-86/topics/train
https://www.datacamp.com/community/tutorials/support-vector-machines-r
https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989
https://www.edureka.co/blog/support-vector-machine-in-r/#Non-linear%20SVM
https://rcompanion.org/handbook/E_05.html

Python

https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python#building
https://scikit-learn.org/stable/modules/svm.html#kernel-functions
https://medium.com/all-things-ai/in-depth-parameter-tuning-for-svc-758215394769
https://www.bitdegree.org/learn/train-test-split