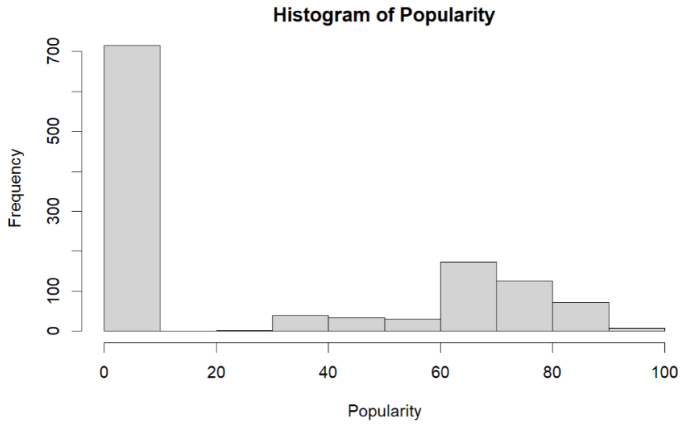
Introduction

Given a music data set, we can use the provided metrics to predict popularity of songs. The given training data contains 15 features, and 1200 samples. The response, popularity, is a continuous variable that ranges from 0 to 98. Ten of the features are continuous predictors, and the other five are categorical, each providing some information about the song qualities.

The first step I took was examining missing data in the training set. Fortunately, there was only one song that was missing data (tempo and time signature). Given that the training data included 1200 samples, I simply removed this song from the training set, as the sample was sufficiently large. This reduces the training set to 1199 samples.

From the now reduced training sample, I will perform 5-fold cross validation, and use average MSE to determine which model performed the best. The criteria used to determine the optimal model is lowest average MSE on the five folds. Once that model is decided, I will retrain that model on the entire training set for the purpose of predicting popularity on the testing set.



The response variable, popularity, has a notably unusual distribution. It is severely right skewed, where 44.6% of the training samples have a popularity of 0. As a result, its summary statistics exhibit this skewedness; the first quartile is 0, the median is 1, and the third quartile is 66. Given this unusual distribution of data, I expect linear models to have a decreased performance, as an assumption for linear regression is a uniform distribution of the response.

I took steps to preprocess the data before my analysis. First, I standardized the continuous features using mean normalization in the model, to decrease the bias that the features with large coefficients could have on the model. This is important for the regularized regression, as it will penalize the terms fairly. This also addresses the issue of extreme values, as they are now reduced to a reasonable scale. Next, I created dummy variables for the categorical features. This greatly increased the feature space; originally there were 15 features, and after creating dummy variables there are 28 features. Because of the high dimensionality introduced, efforts to mitigate the effects of this will be introduced.

Methods Overview

My first instinct was to start with a ridge regression as a baseline, followed by lasso and a relaxed lasso model. This is because the model has 28 features, after creating the dummy variables for categorical predictors. I started with these models, as they are fairly simple and should serve as a starting point to analyze the effects of regularization on the average MSE.

As mentioned in the introduction, I observed that most of the data points have a popularity of 0. For this reason, I attempted KNN and a classification tree to determine if the data were easily separable. Next, I attempted a series of ensemble methods on decision tree models. I attempted bagging, boosting, and random forests. Lastly, I created a model with a neural network. This model was more so a test for viability for a potentially more complex model. These models will be compared based on the average MSE on the testing set.

Summary of Result

The first model I selected, ridge regression, resulted in an average MSE of 918.4 on the validation sets. Lasso regression marginally improved these results, resulting in an average MSE

of 913.1. This is not as large of an improvement as I was hoping for over the baseline ridge regression. Additionally, the sparse model given by Lasso after tuning for optimal lambda still includes 20 features, indicating most of the features are important in the linear model. Relaxed Lasso yields an average MSE of 917.9 which is a slight decrease of the Lasso MSE.

Moving on to the non-parametric models, I tested K Nearest Neighbors for different values of K and a classification tree. After tuning the KNN model and classification tree, I received a top accuracy of 0.704 for KNN and an accuracy of 0.75 with the tree model; however, after examining the misclassified data, the average absolute error was 34.4. Unfortunately, the zeros are not easily enough separable that both classification and regression can be used.

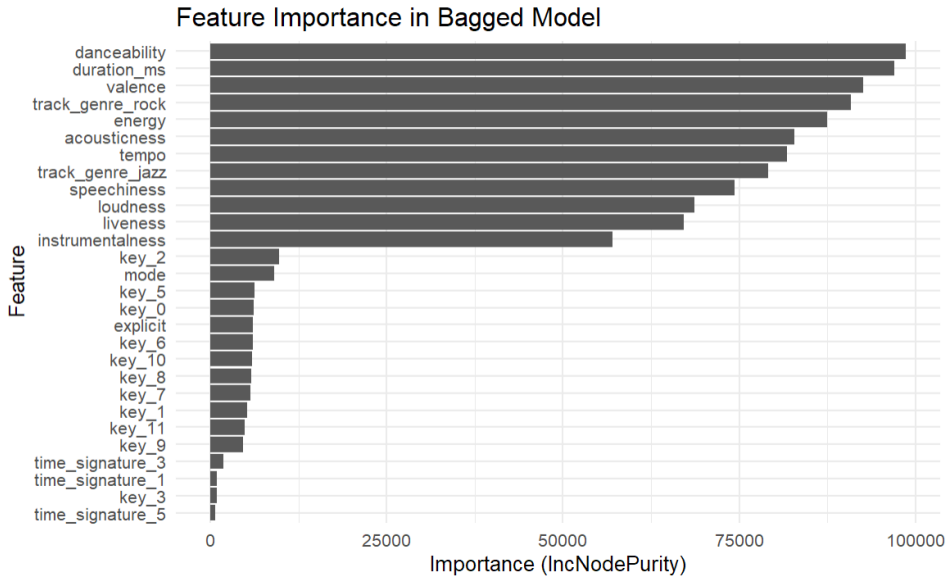
The ensemble methods provided the best results out of all of the models created. The bagging ensemble had an average MSE of 787.3, boosting yielding an average MSE of 877.2, and the Random Forest model yielded an MSE of 789.8. The bagging ensemble is the best model, followed closely by the Random Forest model. Based on the criteria of MSE, the bagging ensemble model has the best performance, which has a corresponding RMSE of 28.1.

The neural network performed the worst, with an MSE of 986.0. The structure of this model was a single hidden layer, with only four nodes. I find this result to be impressive, considering that there was only one hidden layer, as it is performing just slightly worse than the linear models. Building a more complex neural network can have better results, given that this fairly simple neural network had comparable performance to the regularized linear models, and some added complexity could have comparable performance to the ensemble methods.

	Ridge	Lasso	Relaxed	Bagging	Boosting	RandomForest	NeuralNetwork
MSE	918.4	913.1	917.9	787.3	877.1	789.8	986.0

There were a couple different metrics used to determine variable importance. The first of which is those obtained from Lasso regularization. Lasso shrinkage reduced the model to these features: duration, explicit, danceability, energy, mode, instrumentalness, valence, tempo, key 10, key 0, key 6, key 2, key 7, key 9, key 5, key 11, time signature 3, time signature 3, track genre rock, and track genre jazz. This was the shrunk model with the optimal lambda value, and it only removed eight features from the model.

From the tree ensembles, we can get a better understanding of feature importance. Using the total decrease in node impurities, we can determine the top five most important features for the bagged model, and the random forest model. In descending order, the most important features in the bagged model are danceability, duration, valence, energy, and acousticness. For the random forest model, the most important features in are danceability, duration, energy, loudness, and valence.



Based on these three estimates, the most important variables are danceability, duration, valence, and energy. Each of these features show up in the shrunk model and the top five by increase in node impurities of the bagged and random forest models. According to the bagged trees model, there is a significant drop off in importance after a certain point, as shown in the figure. From this, we can determine that key, time signature, explicitness, and mode are not relatively

important features. An interesting contradiction is that the Lasso model included key and time signature in its sparse model, while it is not important in the bagged model.

Conclusions

In conclusion, there were two models that performed the best in terms of mean squared error: bagged trees and Random Forests. These models performed about the same and had significantly less error than the next best model, boosting.

The most challenging aspect of this dataset was that 44% of the data had a popularity of 0. This makes the task of predicting popularity, a continuous variable, more difficult, as the response is skewed right. I believe this explains the worse performance of the linear models, as it fails the assumption that the response variable is normally distributed. I tried to mitigate this by attempting classification on the data first, to separate the zero and nonzero values. However, the classifier accuracy was too low, in my opinion, to attempt using the predictions for regression.

I am fairly confident in my best model, the bagging ensemble. It predicts with an average error of 28.1, which should be sufficient for identifying songs that are overly undervalued or overvalued in the market, which is the original goal. Considering the distribution of song popularity, most songs are either popular or not. With an RMSE of 28.1, on average the model should be able to differentiate between the two modes of the popularity distribution.

Something that would be very useful to improving the success of the model is expanding the classes in the track genre. There are plenty more genres than just rock, pop, and jazz, and these features were very important, as shown in the figure of bagged model feature importance. Adding additional categories of track genre would provide more information in a category that has shown very important in making a good model.