Song Opinions Capstone Final Report (Data Science HarvardX)

Daniela Garcia

10/27/2020

Introduction

Dataset Description

The music/emotions dataset used in this project contains data on a range of songs and reviews that listeners have given them. During the study that produced the dataset, listeners were able to choose the genre of the song and could skip songs. The study included 400 song excerpts (1 minute long) in 4 genres (rock, classical, pop, electronic).

Each row in the data set contains a song ID and its genre, in addition to the listener's age, gender, native tongue, the emotions they felt when listening, their mood before listening, and their opinion of the song (liked, disliked, no opinion).

The "mood" variable denotes the listener's mood before listening and is based on a Likert scale with 1 being very bad and 5 being very good.

Listeners indicated values of 1 or 0 for whether or not they felt each of the 9 emotions, respectively, when listening. The emotions are described as follows:

- Amazement: Feeling of wonder and happiness

- Solemnity: Feeling of transcendence, inspiration. Thrills

- Tenderness: Sensuality, affect, feeling of love

- Nostalgia: Dreamy, melancholic, sentimental feelings

- Calmness: Relaxation, serenity, meditativeness

- Power: Feeling strong, heroic, triumphant, energetic

- Joyful: activation Feels like dancing, bouncy feeling, animated, amused

- Tension: Nervous, impatient, irritated

- Sadness: Depressed, sorrowful

Project Goal

The goal of this project is to predict the opinion a user will have about a song after listening to it: whether they liked it, disliked it, or had no opinion.

Key Steps

The key steps of this project are as follows:

- Download and clean the dataset ("music_opinion")

- Explore the variables in the dataset ("music_opinion")

- Split the dataset into a training ("music_train") and testing ("music_test") set

- Generate predictive models using only the training ("music_train") data

- Select a final model (the most accurate) after generating predictions for the testing ("music_test") set

Methods and Analysis

Data Cleaning

The finalized music_opinion dataset is generated first by downloading the data from its url. Then, the "genre" and "mother.tongue" variables are converted into factors. The "liked" and "disliked" variables are combined together into one "opinion" factor variable, which equals "liked" if the variable "liked" is 1, "disliked" if the variable "liked" is 0, and "no opinion" if both the variable "liked" and "disliked" are 0.

The music_opinion dataset is then split into a testing ("music_test") and training ("music_train") set, where 90% of the music_opinion data is used for the training set. The testing set is filtered so that all track IDs, genres, ages, moods, genders and mother tongues in the testing set are also in the training set. This prevents any errors when making predictions on the testing set. Any rows removed from the testing set during this process are then added back into the training set.

**Data Exploration and Visualization**

First, I explored how the average age and mood before listening differs between different opinions. Based on the table below, it appears as though people who liked their songs were slightly older on average and tended to be in a little bit of a better mood before listening.

```
##      opinion  avg_age avg_mood
## 1   disliked 31.53946 3.435807
## 2      liked 32.30933 3.560620
## 3 no opinion 31.56349 3.458455
```
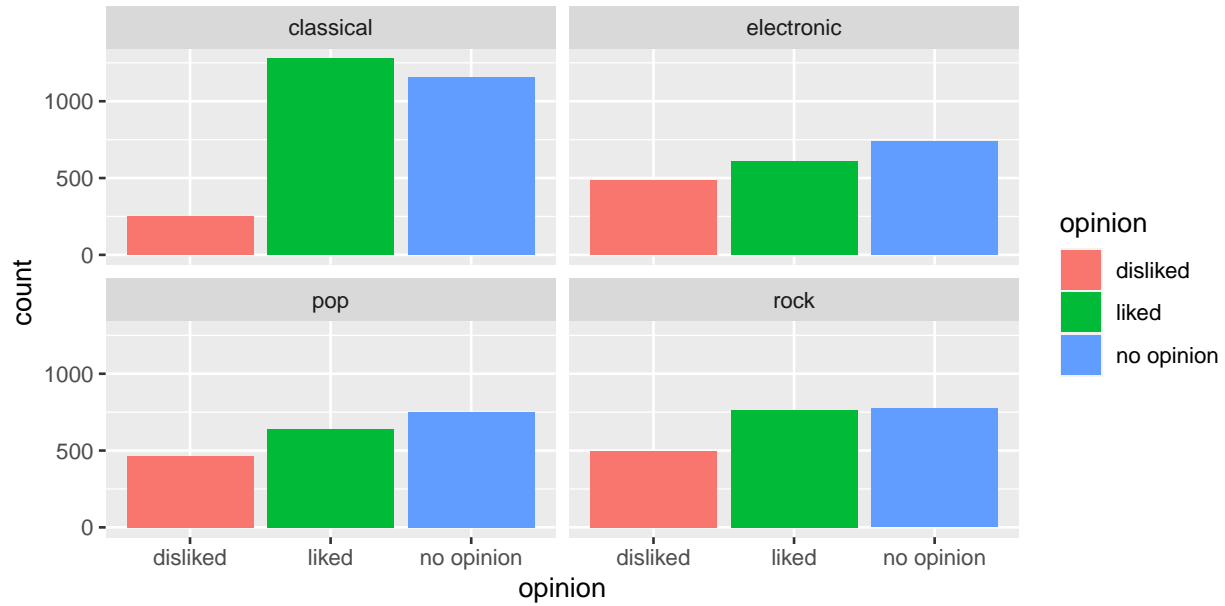
I then plotted the distribution of opinions for each emotion. Based on the plot below, it appears as though listeners often liked calm, joyful, solemn, and tender songs more than they disliked them, while lots of listeners disliked tense songs.
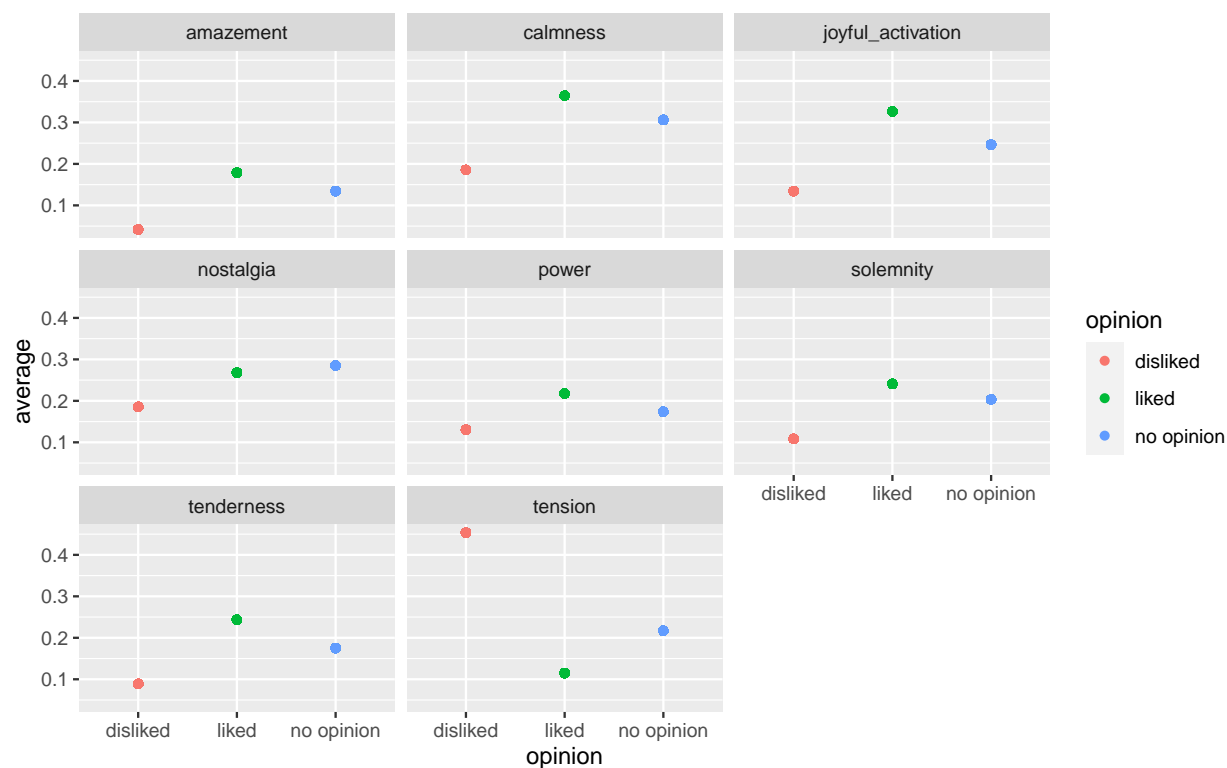

Next, I plotted the distribution of opinions for each gender. Both men and women liked more songs than disliked them, and fairly similar amounts of listeners either liked or had no opinion about a song for each gender.
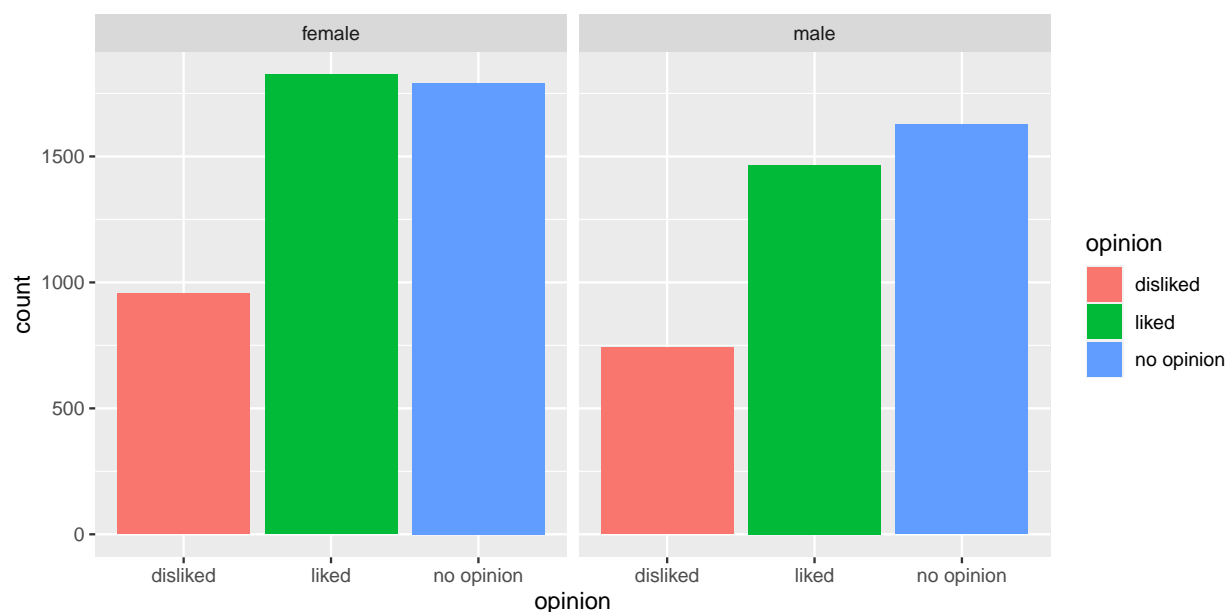
Next, I plotted the distribution of opinions for each song genre. The difference between likes and dislikes was most prominent for classical music, with likes being much more common than dislikes.

Finally, I plotted the distribution of opinions for each mother tongue (with a log10 transformation of the y-axis). The majority of mother tongues had more likes than dislikes, with the exception of Portuguese.
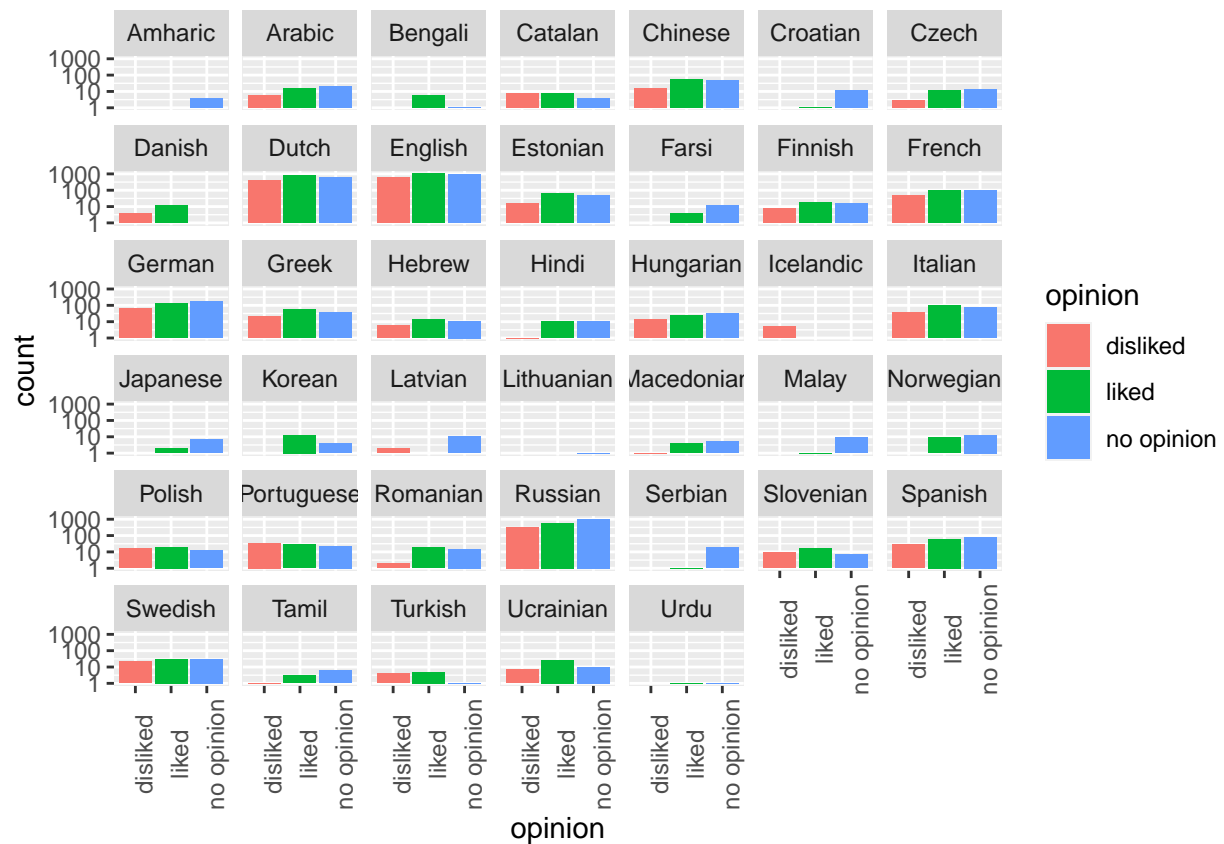
Insights Gained

Based on the exploration conducted above, it appears that the variables "age", "mood", "emotion", and "genre" all have somewhat of an effect on the opinion a listener has of a song. That being said, I plan to include all of the predictors in the training process, in case I misinterpreted or underestimated any of the associations in my exploration process.

Modeling Approach

Based on the size of the training data, it is possible to train classification machine learning models on the training data using the "caret" package in R.

The models I selected to train are rpart, lda, naive_bayes, knn, multinom, Rborist, rf and svmLinear, which are common machine learning algorithms. I generated a function that performs the "train()" function for each algorithm on the training data. I then saved these generated models in a variable called "fits".

Next, I generated predictions for the testing set ("music_test") opinions using each of the models and saved the predictions in a variable called "predictions". I compared each of these sets of opinion predictions to the actual opinions in the testing set to find the accuracy of each model. I saved these accuracies in a variable called "accuracies". The table below shows the accuracy for each of the models.

```
##           model  accuracy
## 1         rpart 0.4596200
## 2           lda 0.4774347
## 3   naive_bayes 0.4821853
## 4           knn 0.5023753
## 5      multinom 0.4833729
## 6       Rborist 0.5154394
## 7            rf 0.5201900
## 8     svmLinear 0.4643705
```

I then decided to combine the 8 models into an ensemble model. For each entry in the testing ("music_test") set, the ensemble prediction was set equal to the most common prediction among all 8 models. I determined the accuracy of the ensemble model by comparing its opinion predictions to the real opinions in the testing set. The accuracy of the ensemble model was $\boxed{0.51188}$. Among the 8 models, Rborist and rf were the only models that were more accurate than the ensemble, suggesting that random forest algorithms are the best at making predictions on this data set.

Separately, before making predictions on the testing data, each predictive model estimated its own accuracy. I found the average of these estimated accuracies among the 8 models and built a second ensemble model using only the models whose estimated accuracy was greater than or equal to the mean estimated accuracy. The accuracy of this second ensemble model is $\boxed{0.52732}$.

Results

Based on the accuracies of the 8 models separately in addition to the accuracies of the first and second ensemble models, I decided to use the second ensemble model as my final model, which takes into account the lda, knn, multinom, Rborist, and rf predictions. The final model accuracy was $\boxed{0.52732}$.

Conclusions

Brief Summary

This project took data on songs and reviews that listeners have given them and produced a model that predicts listeners' opinions regarding the songs. Listeners can either like, dislike, or have no opinion about a song. The predictive models I subsequently generated take into account the effects that the listener's age, gender, mother tongue, mood before listening, emotions felt whilst listening, and the song's ID and genre have on the opinion the listener has of the song in order to generate predictions. The models were trained on training data that provides opinions given different songs and listeners. After training on this data, I generated predictions on the testing data using each of the models. I determined the accuracy of each model and selected my final model based on the model that most accurately predicted the opinions of the testing data.

Potential Impact, Limitations, and Future Work

The predictive quality of this model is moderately strong. It is certainly stronger than a model that simply guesses the opinion of each user, which has an accuracy of about 31.2% (randomly select between 3 opinions), but it is still subject to limitations.

For example, the model is trained on a fairly small set of data. With limited data, it is harder to find trends and associations between listeners, songs, and opinions. Therefore, the model's predictive quality is not as strong. If it had more data to train on, it could make more astute predictions about future listeners. Another limitation of the model is that it takes all variables into account when generating predictions. This could be unnecessary, as not every predictor may be significant enough in predicting a listener's opinion about a song. A stronger model might only take into account the most significant predictors, which would also make the model-generating process faster.

In addition, I originally started the model-building process with just 8 popular machine learning algorithms from the "caret" package. There are most likely more accurate algorithms within the package that I did not run, either because I was not familiar with them or their computing speed was too long. In a future version, given more knowledge, time, and computing power, I could include more accurate algorithms in the model-building process to make my final ensemble model more accurate.

This model can be useful for determining what opinions different listeners will have regarding different songs. Given information on the listener's age, gender, mother tongue, mood before listening, emotions felt whilst listening, and the song's ID and genre, companies (like Spotify, Pandora, or Apple Music) and music producers can predict how their listeners will react to their songs. This can be useful for determining, for example, which songs will be liked by a wide range of listeners, which songs will be liked by smaller, niche groups of listeners, and which songs they should recommend to increase ratings and listener engagement.