CS5033 - SL Project Checkpoint

Airi Shimamura Khoi Trinh

1. Introduction

For our Supervised Learning project this semester, we want to create a few models to predict a user's preference for a song on the Spotify streaming platform.

Each song on Spotify has their own set of 11 features. These features are: danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, and tempo.

A user can request their streaming history from Spotify via the P rivacy section of their account. For the purpose of this project, Khoi's one-year streaming history from February 2022 to February 2023 will be used as the 1 class, for songs we like; and about 4200 random songs which Khoi has never heard will be used as the 0 class, for songs we do not like.

The dataset has 56,210 songs in class 1, and 4,170 songs in class 0. However, the songs in class 1 have duplicates, due to the nature of streaming a song multiple times. We expect this number will be much lower once the duplicates are removed.

2. Hypotheses

Our first hypothesis is that our supervised learning models will be able to classified these songs as likes or dislikes with at least 90% accuracy.

Our second hypothesis is that out of the four chosen algorithms; Random Forest will give us the best performance; followed by Decision Tree, then Logistic Regression, and finally Naive Bayes. We will evaluate each algorithm's performance by comparing the value of Precision, Recall, and produce ROC curves for each algorithm.

3. Experimental Progress

3.1. Data Preprocessing

For the preprocessing of the data, we removed duplicates, the preference column, as well as scaling and shuffling the data before passing it into the model.

3.2. Khoi's Progress

For this project, Khoi is implementing two methods: logistic regression and decision trees.

Logistic Regression is a statistical machine learning algorithm predicting the probability of a target variable by fitting a logistic function to the input features. This optimizes the parameters of the logistic function by minimizing the cost function, which measures the difference between the predicted probabilities and the actual class labels.

Decision Tree is a machine learning algorithm to predict the class of an input based on its features. This algorithm partitions the input space into increasingly smaller regions based on the values of the input features by selecting the features and test conditions that best separate the training data into the different classes. This process is repeated recursively to create a tree-like structure until a final decision is made.

Currently, he is working on the logistic regression model.

For the model, Khoi considered the learning rate $\alpha = 0.01$, and the model will train for 1000 iterations. Moreover, he implemented the sigmoid, logistic loss, and gradient functions from scratch. After one run, here are his results:

Precision: 94.56%

Recall: 49.18%

Accuracy: 96.83%

Along with this ROC curve

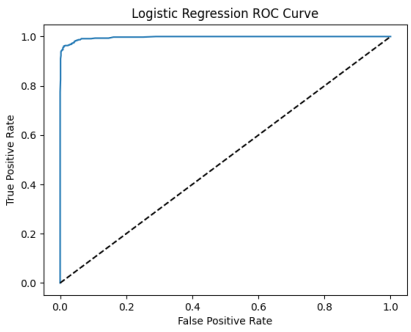

Figure 1. ROC Curve for Logistic Regression

3.3. Airi's Progress

For this project, Airi is implementing two methods: naive bayes and random forest.

Naive Bayes is a group of probabilistic machine learning algorithms based on applying Bayes' theorem with the assumption of independence between features. This assumes that the presence or absence of one feature does not affect the presence or absence of any other feature within a class, and calculates the probabilities of each class label given the values of the features from the input. The class with the highest probability is then selected as the predicted class label for the given input.

Random Forest is an ensemble learning method that combines multiple decision trees to improve the performance and reduce overfitting. Each decision tree is constructed by randomly selecting a set of features and samples from the training set. This randomization helps to make a model less sensitive to noise and outliers in the data, and outperforms other algorithms.

For the random forest model, Airi used the Gini index as the criterion for tree splitting with 20 decision trees, and the Naive Bayes model was implemented based on the Bayes' theorem. Also, to evaluate the performance of the models, precision, accuracy, and recall were calculated and both of the models achieved high scores as follows

— Random Forest:

Precision: 97.41%

Recall: 49.29%

Accuracy: 97.91%

— Naive Bayes:

Precision: 97.78%

Recall: 49.04%

Accuracy: 97.68%

4. Difficulties Encountered

The ROC curves for the Naive Bayes, Decision Trees, and Random Forest are all straight lines, which indicates the models cannot identify any relevant patterns or features, so either the data processing or the models need to be re-evaluated.

Moreover, Naive Bayes performance is much better than Logistic Regression and Random Forest; so all of these models will need to be re-evaluated.

5. Future Work

Khoi will continue refining his logistic regression model, as well as work on the decision tree model. The decision tree model will be useful for random forest, as well.

To enhance the performance of the random forest and Naive Bayes models, Airi will continue re-evaluating them and obtain ROC curves. Moreover, she will expand the random forest model by increasing the number of trees and maximum depth for each tree. In addition, she will experiment with two other criteria, namely entropy and chi-squared, to assess their impact on the model's performance.

Finally, as mentioned in the previous section, the Logistic Regression, Naive Bayes, and Random Forest models will need to be redone, to ensure accurate performance.