COMP300027 assignment 1

Niha Waseem
1212439

Roshni Siva
1209191

April 2023

1 Task 1

1.1 Question 1

We call the function evaluate() to calculate the accuracy and generate a classification report and confusion matrix for instances in the test data set.

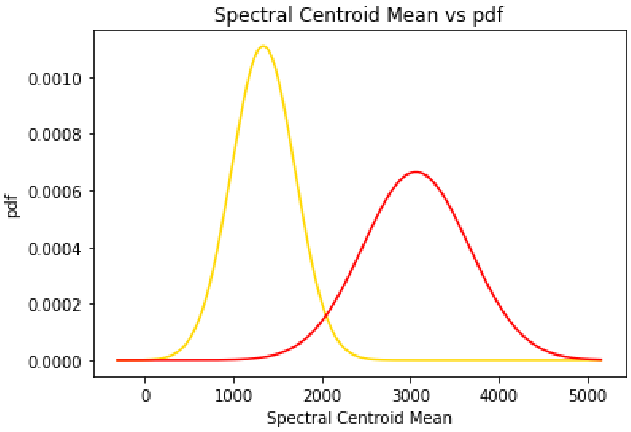
```
Accuracy: 0.9767441860465116.
Classification report:
              precision    recall  f1-score   support

   classical       0.95      1.00      0.98        20
         pop       1.00      0.96      0.98        23

    accuracy                           0.98        43
   macro avg       0.98      0.98      0.98        43
weighted avg       0.98      0.98      0.98        43

Confusion matrix:
[[20  0]
 [ 1 22]]
```

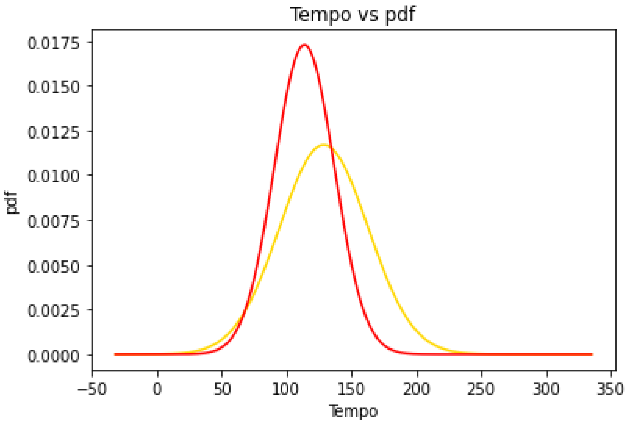
We observe an accuracy of 0.976 and similar values of macro and weighted average due to the fact that this data-set is well-balanced.

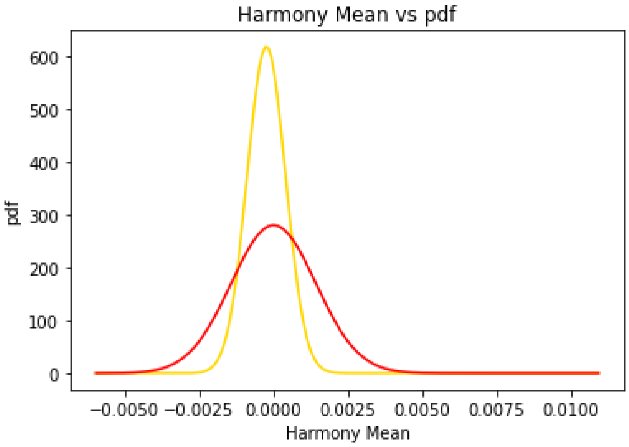
1.2 Question 2

The following plots show the pdf distribution for spectral centroid mean, harmony and tempo. The x-axis shows the range of the class and the y-axis reflects the pdf P(X—Class = pop) and P(X—Class=classical). The yellow line shows the distribution of classical and the red shows the distribution of pop music.

To classify classical and pop, we would use the spectral centroid mean due to the significant difference between the mean for classical and pop music. From the graph, we can see spectral centroid means have the least overlap between them. This may reduce false negatives which affect the recall and sensitivity of the model.







2 Task 2

2.1 Question 3

We are comparing the accuracy of the full model to the 0R baseline and one-attribute baseline. With the full model, we get the accuracy to be 0.49; refer to the classification report below:

```
Accuracy: 0.49.
Classification report:
              precision    recall  f1-score   support

       blues       0.50      0.21      0.30        19
   classical       0.89      0.85      0.87        20
     country       0.39      0.69      0.50        16
       disco       0.43      0.41      0.42        22
      hiphop       0.46      0.29      0.35        21
        jazz       0.50      0.33      0.40        18
       metal       0.37      0.90      0.52        20
         pop       0.80      0.70      0.74        23
      reggae       0.53      0.57      0.55        14
        rock       0.20      0.11      0.14        27

    accuracy                           0.49       200
   macro avg       0.51      0.51      0.48       200
weighted avg       0.50      0.49      0.47       200
```

For the 0R model, we used the Dummy Classifier with the most frequent strategy. To ensure accuracy over repetitions, we ran the model 30 times and obtain the following result:

```
Average accuracy over 30 runs is: 0.07.
```

We verify this by checking the most frequent label in the training data set and confirming it to be **reggae**. Since the number of instances of reggae in the test dataset = 14 (as seen in the classification report above) and 14/200 = 0.07, it is clear that the model is predicting the class for every instance to be reggae. For the one-attribute model, we loop over all attributes (using one at a time) and find the accuracy each time to be 0.07 for all attributes. The output is shown below (shortened for clarity):

```
Accuracy for chroma_stft_mean is: 0.07.
Accuracy for chroma_stft_var is: 0.07.
Accuracy for rms_mean is: 0.07.
Accuracy for rms_var is: 0.07.
```

Once again, this due to **reggae** being the most frequent class in the training dataset, giving it a high prior probability which we believe is controlling the posterior probability when each of the attributes are used.

Therefore, we report that our model's accuracy of 0.49 is better than the 0R and one-attribute baselines in all cases for this test data set. However, as the training data set gets more imbalanced, 0R and one-attribute baseline may give higher accuracies.

2.2 Question 4

We used sklearn's KFold() to set up cross-fold validation splits and test different training set sizes. Considering runtime constraints, we chose k-values of 5, 10, 20, and 50. Additionally, we added a parameter to iterate over the cross-validation function multiple times to observe average results for accuracy. To compare our model's performance using cross-fold validation against a normal implementation using the provided training and testing datasets, we refer to the classification report:

```
Average accuracy: 0.49.
Classification report:
              precision    recall  f1-score   support

       blues       0.50      0.21      0.30        19
   classical       0.89      0.85      0.87        20
     country       0.39      0.69      0.50        16
       disco       0.43      0.41      0.42        22
      hiphop       0.46      0.29      0.35        21
        jazz       0.50      0.33      0.40        18
       metal       0.37      0.90      0.52        20
         pop       0.80      0.70      0.74        23
      reggae       0.53      0.57      0.55        14
        rock       0.20      0.11      0.14        27

    accuracy                           0.49       200
   macro avg       0.51      0.51      0.48       200
weighted avg       0.50      0.49      0.47       200
```

Accuracy was chosen as a sufficient evaluation metric for the model's performance as this is a multiclass classification problem and precision and recall appear to vary across classes and hence would not give meaningful evidence of improvement. Additionally, as the training data set is well balanced over the 10 classes, accuracy is a reliable metric for performance.

Our function takes the dataset and the list of k-values and returns mean accuracies for each k-value. Below are some sample output:

```
Number of runs: 1
Average accuracy for k = 5: 0.5370000000000001
Average accuracy for k = 10: 0.53
Average accuracy for k = 20: 0.53
Average accuracy for k = 50: 0.524
```

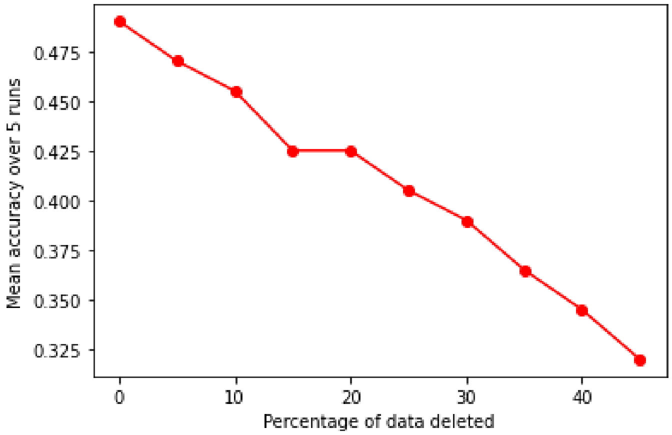
To verify our results, we increase the number of runs:

```
Number of runs: 3
 Average accuracy for k = 5: 0.525
 Average accuracy for k = 10: 0.527
 Average accuracy for k = 20: 0.527
 Average accuracy for k = 50: 0.53
```

Since the accuracy values remain more or less constant, we conclude that the model's accuracy is higher using cross-fold validation than our normal accuracy, which was 0.49. Our results imply that an ideal value of k (number of folds) may be 5 as it would give similar accuracy as a higher number of folds with less runtime. In order to cross-check the fact that the accuracy value remains constant as number of folds increases, we ran sklearn's GaussianNB() function on our dataset and got similar results.

2.3 Question 6

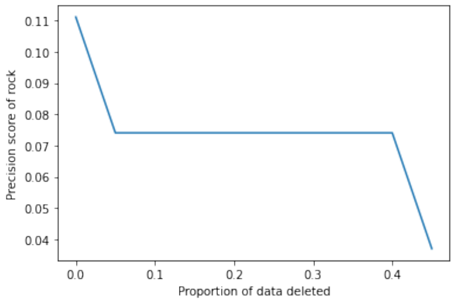
To control the percentage of attributes deleted from the test dataset, we used the sample() method with a parameter that specifies the proportion of attributes to randomly select and set to NaN. In our predict() function, we set a condition such that missing attributes are skipped. Our results show that our model is quite robust to missing data, with the results graphically represented as follows.

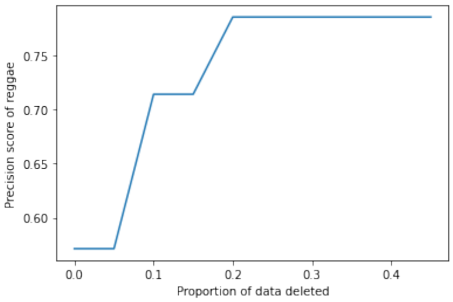


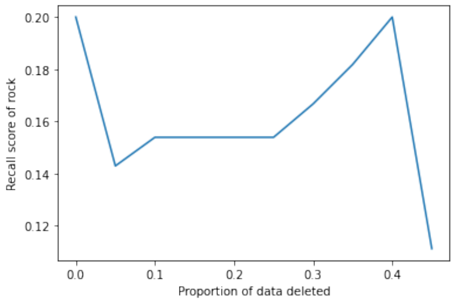
To explore further on implications this may have on specific classes, we took a look at precision and for the most and least frequent classes in the test data set - rock and reggae respectively.

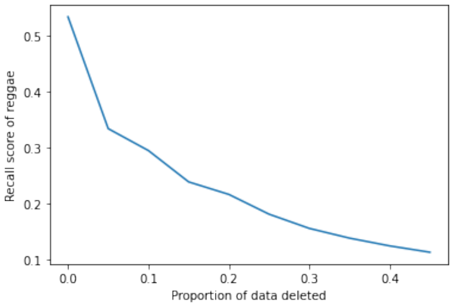
For precision, we observe that the precision for rock decreases as the percentage of data deleted increases, but that of reggae increases. Meanwhile, the decrease in recall for reggae is a constant decrease while that of rock is steep and only happens at the end, with a higher percentage of data being deleted.

Therefore, we conclude that as the number of missing attributes in the test data increases, classes that are more frequent in the test data may be more likely to be detected than classes that are less frequent.


Precision score of rock
Proportion of data deleted


Precision score of reggae
Proportion of data deleted


Recall score of rock
Proportion of data deleted


Recall score of reggae
Proportion of data deleted