Practical Data Science Assignment 2

s3895776

Guo An

s3895776@student.rmit.edu.au

Affiliations: RMIT University.

Date of Report: 21st of May

I certify that this is all my own original work. If I took any parts from elsewhere, then they were non-essential parts of the assignment, and they are clearly attributed in my submission. I will show I agree to this honor code by typing "Yes": Yes

TOC

Contents

Executive summary

The goal of this report was to compare two supervised models and their classification on the mood of a song given its attributes. Two different supervised models, decision tree classifier and k nearest neighbours' classifier, were fitted to a 400 sample Turkish music dataset. Both models were evaluated to be better than each other for different target classes. Generalisation was observed in both models as the model performed similarly on testing and validation. Of the two models, decision tree classification was recommended, as the small number of samples required thorough tuning.

Introduction

Music is a popular entertainment venue in many first-world countries. Record labels and music vendors gain from dissecting and using core components of music to appeal to a target demographic. Practical examples of music classification emerge in areas with vested interest, such as recommendation systems [(1)]. Recommendation algorithms may recommend music based on its features, such as the feeling of the song. If the class is missing, we can perform a prediction on what type the music belongs to by extracting and modelling its features. A functional model would be able to automate this process.

In this report, two different classifiers were modelled and evaluated on their classification rate of the mood of a song. The usefulness of each model in a music classification context was extrapolated.

Methodology

Data retrieval

Data retrieved from Turkish music emotion dataset [2].

These are the attributes of each observation in this dataset:

- Are classified by the feeling the songs give: happy, sad, angry, relax.
- Are verbal or non-verbal music.
- Are from different genres of Turkish music.
- Are 30 second samples of the original song.

There are a total of 100 music pieces for each class, with 400 observations in total.

Information about the dataset is addressed in data retrieval as well.

Preparation

Design & steps of pre-processing.

General cleaning of the dataset included:

- Processing missing values
- Sanity checks

Post exploration cleaning included:

- Log transformation
- Outlier clipping in distribution.

Exploration

All features were continuous except the target feature.

For continuous features, scatter plots and scatter matrix can be used to find the relationship between variables. Scatter plots were separated by the target variable to determine the degree of separation each class had in one relationship.

Histogram graphs the distribution of the variable.

Boxplot was used to plot the range of values and outliers. The plot can also be divided by categorical features, which was used to determine the features correlation to the target variable.

Modelling

A decision tree classifier and k nearest neighbours' classifier was used to classify the data with its target values.

The dataset was split 60-20-20 into training, validation, and test data respectively. Due to the small size of the dataset, hold-out validation was considered over k-fold validation.

Evaluation was performed using a confusion matrix, precision, recall and f1 score.

- The confusion matrix is a visualisation of the predictions compared to the actual result. This is useful for observing which class is predicted well by comparing the samples present in the diagonal to the number of samples present in that class.
- Precision is a measure of how accurate the prediction is.
- Recall is measure of how accurately the class is predicted.
- F1 score averages precision and recall.

These measures determine how well the classifier predicts the target feature.

Evaluation was performed on validation data to return results for tuning the classifier. In the decision tree classifier, the tree was used to tune the classifier further.

Then, the model is evaluated on test data to observe how well it generalises.

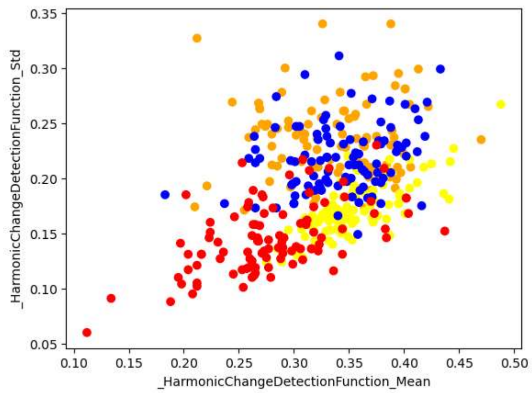
Feature selection was used for both classifiers.

For k nearest neighbours, hill climbing was used. Its scoring was determined with the internal function (KNeighborsClassifier.score), which returns the mean accuracy of the classifier on test data.

Feature selection is done implicitly by the decision tree classifier. The classifier chooses the best sub-tree to generate based on the resulting gini values of the split.

Results

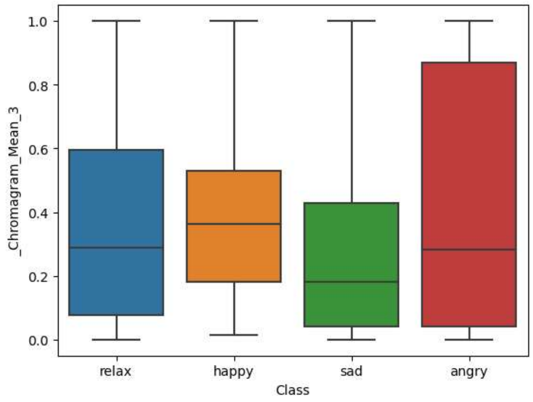
In the EDA, features and feature relationships were found to be significant for identifying the angry class.



```
['orange = relax', 'yellow = happy', 'blue = sad', 'red = angry']
```

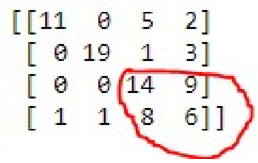
This scatter plot displays that for the angry class, observations tend to have low mean and low std in terms of its harmonic change. This separates the angry class from the other classes with a higher mean and std.

```
<Axes: xlabel='Class', ylabel='_Chromagram_Mean_3'>
```



This boxplot indicates that for this feature, the closer the value is to one the higher the chance that the observation is an angry class.

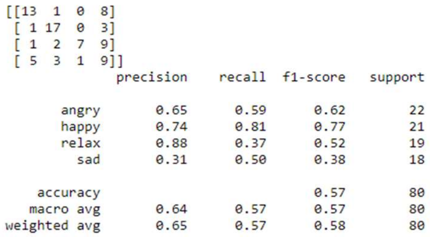
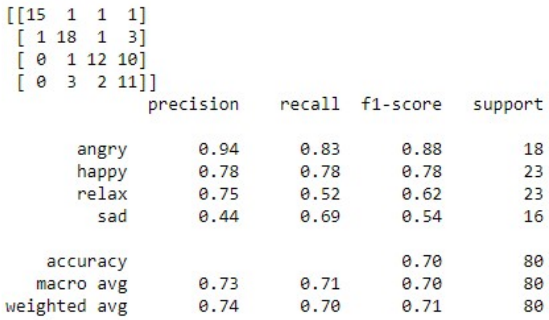
Both classifiers identified relaxing songs as sad songs frequently, and vice versa. This resulted in poor performance in both models for these classes.



```
[[11  0  5  2]
 [ 0 19  1  3]
 [ 0  0 14  9]
 [ 1  1  8  6]]
```

Sample from the Decision tree classifier confusion matrix predicting test data.

K-neighbours classifier

Validation evaluation	Test evaluation
	

```
[[13  1  0  8]
 [ 1 17  0  3]
 [ 1  2  7  9]
 [ 5  3  1  9]]
              precision    recall  f1-score   support

       angry       0.65      0.59      0.62        22
       happy       0.74      0.81      0.77        21
       relax       0.88      0.37      0.52        19
         sad       0.31      0.50      0.38        18

    accuracy                           0.57        80
   macro avg       0.64      0.57      0.57        80
weighted avg       0.65      0.57      0.58        80
```

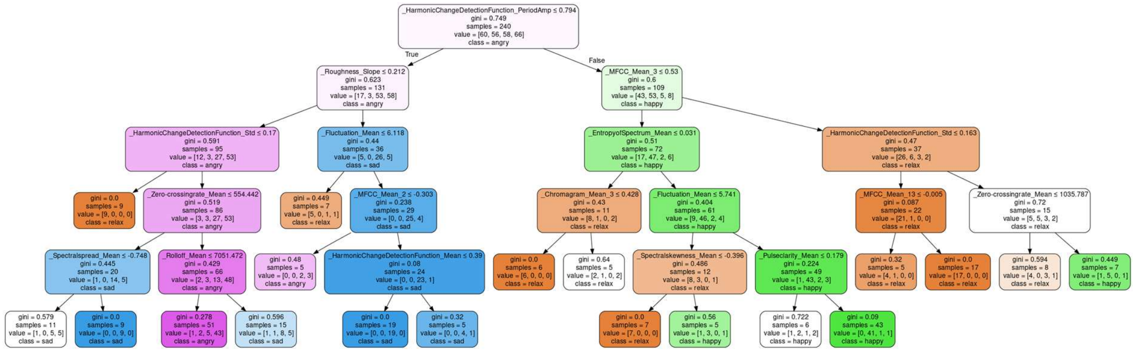
```
[[15  1  1  1]
 [ 1 18  1  3]
 [ 0  1 12 10]
 [ 0  3  2 11]]
              precision    recall  f1-score   support

       angry       0.94      0.83      0.88        18
       happy       0.78      0.78      0.78        23
       relax       0.75      0.52      0.62        23
         sad       0.44      0.69      0.54        16

    accuracy                           0.70        80
   macro avg       0.73      0.71      0.70        80
weighted avg       0.74      0.70      0.71        80
```

Model can generalise, performing better on test data than validation in terms of average score and f1 score.

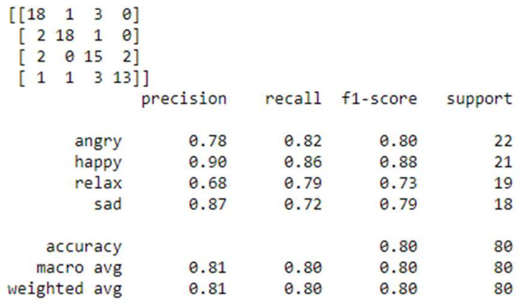
Relax has a higher precision than sad while sad has a higher recall than relax. What we observe here is that because the classifier frequently predicts the relax class as the sad class, the predictions of sad class are inaccurate and the rate which relax is called correctly is also inaccurate.



Decision tree generated from final model.

After a depth of two, all subtrees struggled to achieve homogeneity. This is indicated by the small number of splits made per decision node. For example, in the left subtree, the sad class was split 3 times with a small number of samples separated each time. Each leaf is impure with a gini value at least above 0.3. This indicates a need for data exploration to find or process features that classify each class accurately.

Validation evaluation

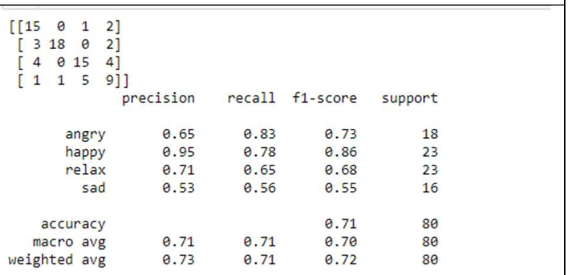


```
[[18  1  3  0]
 [ 2 18  1  0]
 [ 2  0 15  2]
 [ 1  1  3 13]]
              precision    recall  f1-score   support

       angry       0.78      0.82      0.80        22
       happy       0.90      0.86      0.88        21
       relax       0.68      0.79      0.73        19
         sad       0.87      0.72      0.79        18

    accuracy                           0.80        80
   macro avg       0.81      0.80      0.80        80
weighted avg       0.81      0.80      0.80        80
```

Test evaluation



```
[[15  0  1  2]
 [ 3 18  0  2]
 [ 4  0 15  4]
 [ 1  1  5  9]]
              precision    recall  f1-score   support

       angry       0.65      0.83      0.73        18
       happy       0.95      0.78      0.86        23
       relax       0.71      0.65      0.68        23
         sad       0.53      0.56      0.55        16

    accuracy                           0.71        80
   macro avg       0.71      0.71      0.70        80
weighted avg       0.73      0.71      0.72        80
```

Model loses accuracy in terms of f1 score and average score. The model overfitted to training data.

For k nearest neighbours' model, F1 score is higher for class angry and sad than the decision tree model on test data. For class happy and relax, it is lower.

Discussion

The classifiers were not suitable for classification of multiple classes, but a few specific ones.

With relaxing and sad songs, both models performed poorly. With happy and angry songs, the models performed well.

We observe in the EDA that relaxing and sad songs tend to have the same values in features. As a result, more time was spent tuning these classes that were not separated easily.

The confusion matrix for both classifiers show that the classifiers mislabelled the relaxing class as the sad class and the sad class as the relaxing class often. The relatively low f1 score both classes possess compared to angry and sad classes indicate poor accuracy as well.

In K neighbours' classifier, the weights were set to distance to weigh closer observations more heavily. n_neighbours was set to 8. These parameters were considered due to the small number of observations. P=1 or Manhattan distance seemed to work best with classifying the data point correctly, which evaluates all features equally.

In the decision tree classifier, minimum samples split is set at 10. Leaves below the split size were deemed uninformative or misleading. Because of the small sample size, we are expecting further splitting to overfit more. Similar observations were made for splits at a depth of 4 or 5. Leaves with a large sample size were mostly homogenous, while the leaves with a smaller size tended to be less homogenous. Splitting on a small sample size leads to overfitting, thus the tree is cut short.

Model recommendation

Neither of the models provided better accuracy, however the decision tree classifier required more parameters to tune and overfitted.

For the Turkish music dataset with many continuous variables, the decision tree would have difficulty separating target classes based on linear relationships. The decision tree can only decide boundaries for continuous variables, meaning it can separate data points vertically or horizontally, but not diagonally. The k nearest neighbour classifier is more accurate in selecting the class that is closest to the point that is classified, regardless of boundary. I therefore recommend for the Turkish music dataset that the k nearest neighbour classifier is selected over the decision tree due to the problem space being reliant on discrete features.

Conclusion

It was concluded that k nearest neighbours was more suited to classifying continuous features over the decision tree. The models had similar accuracy on average, although both had different performance in different classes.

Because features identified certain classes accurately, in future research we could identify and discuss these features on the separation of each label and apply the results to other music classification models.

Due to the models performing better on separate classes it may indicate that both models were good at identifying classes in different features, which could be expanded upon in independent study of the two models in a music classification context.

References

Lecture notes for Practical Data Science provided by Prof. Yongli Ren RMIT, March 2023

Music Classification: Beyond Supervised Learning, Towards Real-world Applications (1)

https://music-classification.github.io/tutorial/part1_intro/what-is-music-classification.html

Published by Minz Won, Janne Spijkervet, Keunwoo Choi, 2021

Accessed 21st May 2023

Turkish Music Emotion Dataset (2):
https://archive.ics.uci.edu/ml/datasets/Turkish+Music+Emotion+Dataset

Published by Mehmet Bilal Er

Accessed 17th April 2023

Additional citation: Bilal Er, M., & Aydilek, I. B. (2019). Music emotion recognition by using chroma spectrogram and deep visual features. Journal of Computational Intelligent Systems, 12(2), 1622â€"1634. International Journal of Computational Intelligence Systems,
DOI: [https://www.atlantis-press.com/journals/ijcis/125927469]