CMLS Homework 1 - Audio Event Classification

E. Castelli, E. Intagliata, A. Rizzitiello, G. Zanocco

April 2021

1 GitHub Repository

https://github.com/ElisaCastelli/CMLS-HW1-Group11.git

2 Introduction

The goal of the first CMLS homework which has been assigned to our group is to build a classifier that could predict the audio event recorded in an audio excerpt. As specified in the guidelines, all the experiments have been carried out using the *UrbanSound8k*[1] sampled sounds collection. This database includes 8732 labeled audio excerpts of urban sounds organized in ten different categories, drawn from the urban sound taxonomy discussed in this paper [1]. As suggested by the authors, we proceeded to work using the *10-fold cross-validation* technique to evaluate our predictive model.

3 Data analysis

The samples of the dataset have been recorded from 10 classes of different sound sources: air conditioner, car horn, children playing, dog bark, drilling, engine idling, gun shot, jackhammer, siren and street music. The first thing we have done is a preprocessing analysis of the data to understand which were the best characteristics to distinguish them and to check if the dataset is well balanced.

We noticed that the classes “gun shot” and “car horn” have a few samples less than the other ones and that the great part of audio files has a duration of 4.0 seconds. Despite these small differences between audio and classes, as suggested from the authors of the dataset, we have decided to not modify the dataset and also to not do any reshuffling because the files have already been pre-sorted in a proper way to help the comparison.

In order to manage easily the different audios, we have read the .csv file included in the dataset building for each row an object Sound and collect then in a list we have called soundList. In this way, we have retrieved all the information contained in the file and stored them into the attributes of Sound objects.

[1] UrbanSound8k dataset available at https://urbansounddataset.weebly.com/urbansound8k.html

In addition to the attribute already stored in the .csv file, we have chosen to add an array attribute in the object Sound to store the features we have computed for each sound during the reading.

We have tried different approaches but we have decided to choose as features the following ones:
- MFCC mean and standard deviation (25 coefficients each), as in Figure[1]
- Chroma features mean and standard deviation (12 coefficients each), as in Figure[2]
- Spectral contrast mean and standard deviation (4 coefficients each), as in Figure[3]

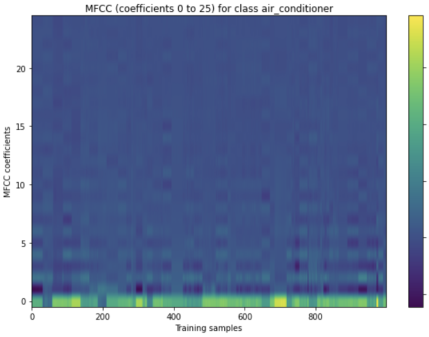

Figure 1: MFCC coefficients for air conditioner's files

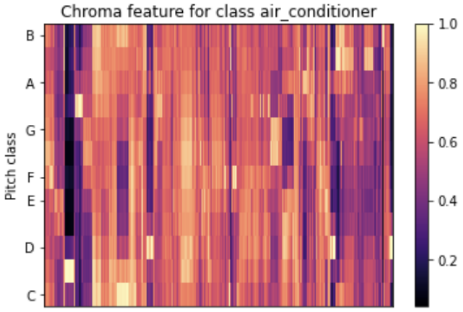

Figure 2: Chroma coefficients for air conditioner's files

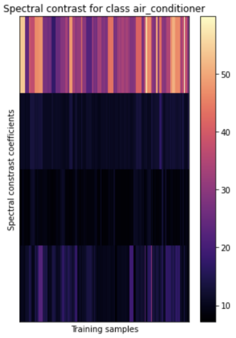

Figure 3: Spectral contrast coefficients for air conditioner's files

In this way we have built a feature vector of 82 coefficients, Figure[4], to capture different qualities about the audio over time, without exceeding in expensive computation.

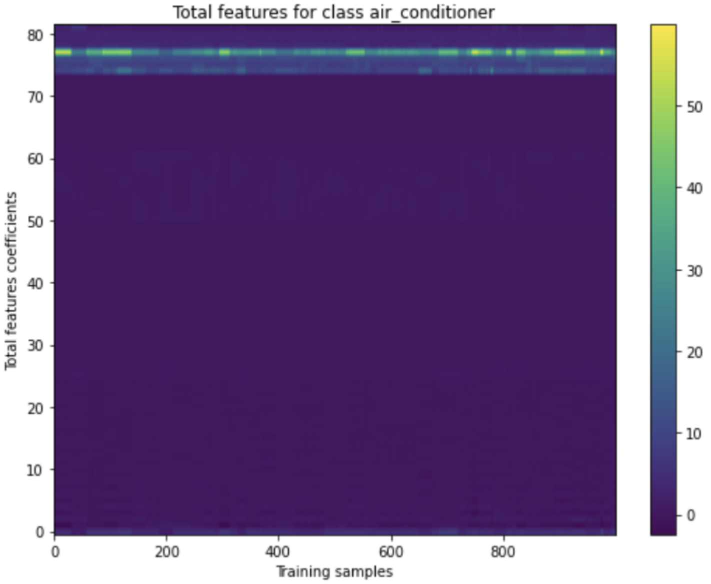

Figure 4: Total features coefficients for air conditioner's files

As suggested by the author of the dataset, we performed the 10-fold cross validation to do the training, test steps and the classification. Only at the end we have taken the ten different results and we have made an average on them.

4 Classification

Once the desired features were computed, we could proceed with the classification. We decided to use SVM (Support Vector Machine) as model for the classifier and RBF (radial basis function) as kernel function.

As for the training and testing steps, we proceeded as follows. At each iteration of the cross-validation algorithm we initialized a new model, then trained it on supervised data from 9 of the 10 predefined folds and finally tested it by predicting the labels of the samples contained in the remaining fold. The results of each prediction were stored for the evaluation part while the model was discarded.

The actual implementation of the SVM classifier is the one provided by the sklearn library. According to its documentation, this classifier is able to perform multiclass labeling using a one-vs-one scheme. This means that for every sample in the test set there is a set of multiple classifiers, one for each pair of possible classes which the sample can belong to. At the end of the prediction, the sample is assigned to the class that has received the most votes.

5 Results

In order to evaluate the performances of the algorithm, we evaluated the classification accuracy and the confusion matrix. Dealing with a multiclass classification problem, we used the *sklearn* function `accuracy_score` to compute the accuracy at each iteration. It returns the fraction of correctly classified samples:

$$[0.630, 0.646, 0.6086486486486486, 0.606, 0.703, 0.579, 0.663, 0.700, 0.680, 0.708] \qquad (1)$$

This gives us an average accuracy of about **65.2 %** which is slightly below the results obtained in [1].

The confusion matrix in Figure[5] is obtained by summing the confusion matrices obtained at each iteration.

We can see the performance between classes is in line with [1], too. In particular both classifiers, even relating on a different feature space, confuse pairs of classeswith similar timbres. For example: air conditioners and idling engines, jackhammers and drills, children and street music.

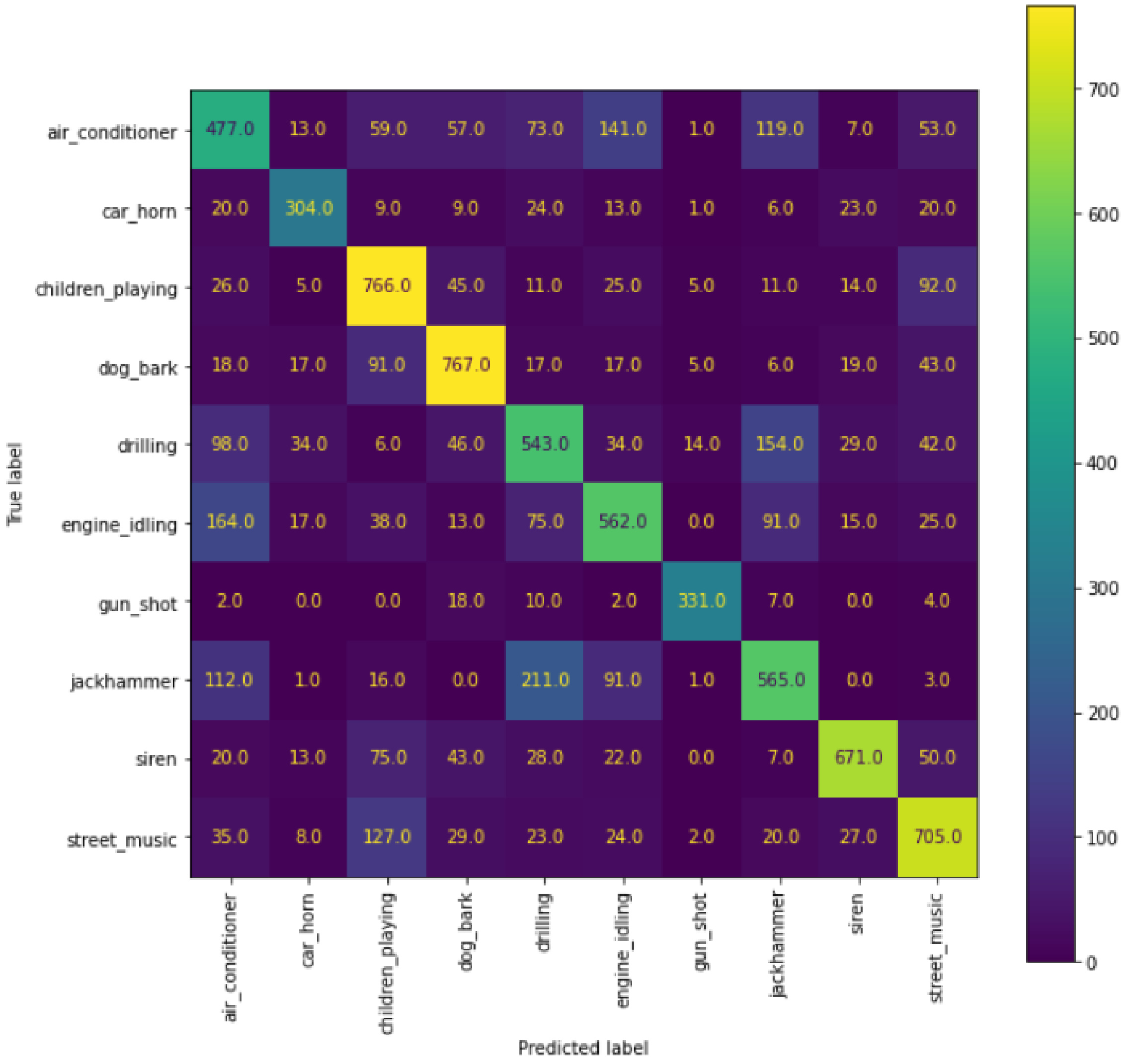

Figure 5: Confusion Matrix

References

[1] Justin Salamon, Christopher Jacoby, Juan Pablo Bello. A Dataset and Taxonomy for Urban Sound Research. *Proceedings of the 22nd ACM international conference on Multimedia. November 2014, pages 1041–1044*. Cited at page 1 3.