LAB 1 REPORT

Music Genre Classification

Manuel Castillo Obregón

Universitat Pompeu Fabra

Machine Learning for Sound and Music

Google Colab Notebook

GitHub Repository

1. INTRODUCTION

This report aims to discuss the different traditional machine learning algorithms applied to audio-based genre classification. Two tasks were conducted using different feature sets derived from music excerpts. Task 1 uses the GenreAll dataset, which consists of Essentia descriptors, whereas Task 2 evaluates two additional datasets containing 30 second aggregated features and 3 second frame level features.

All models were evaluated using 10-fold cross-validation. In addition to classification accuracy: computational time, peak memory consumption, and estimated CO2 emissions were measured using the CodeCarbon library, following principles of sustainable machine learning.

2. DATASETS

Three different feature representations were used in this lab:

1. GenreAll.csv: a high-dimensional feature set based on Essentia descriptors for full-length clips.
2. features_30_sec.csv: aggregated features computed over 30-second excerpts of audio.
3. features_3_sec.csv: short-term features computed on 3-second segments, resulting in a much larger dataset.

3. METHODS

3.1 Algorithms

Following the instructions, the following machine learning algorithms were selected and were tested using Python and Scikit-learn:

1. Logistic Regression: Linear baseline model, effective in high-dimensional spaces and it's also relatively robust to overfitting.
2. k-Nearest Neighbors: Non-parametric method based on distance metrics, which can capture local structure in the feature space.
3. Decision Tree: Interpretable non-linear model that recursively partitions the feature space.
4. Random Forest: Ensemble of decision trees, suitable for modeling complex non-linear relationships and improving stability.
5. Multi-Layer Perceptron: Feed-forward neural network for flexible non-linear decision boundaries.

3.2 Evaluation protocol

For all the experiments a 10-fold cross validation was performed. Stratification helps to preserve class distribution in each fold, which is useful for genre classification.

When necessary, features were standardized with StandardScaler and hyperparmeters were chosen using relatively reasonable defaults. The main objective in both tasks is to perform a comparative analysis rather than tuning.

To evaluate the computational footprint of each model, the following was measured:

1. Wall-clock runtime of the cross-validation procedure.
2. Peak RAM usage during model evaluation.
3. Estimated CO_2 emissions (kg CO2eq) using CodeCarbon.

4. TASK 1: RESULTS AND DISCUSSION

Task 1 uses the GenreAll dataset with Essentia descriptors as input features for genre classification.

4.1. Numerical results

Table 1 reports the cross-validated accuracy, runtime, memory consumption, and estimated CO2 emissions for each model.

Model	Accuracy (mean ± std)	Time (s)	Peak Memory (MB)	CO_2 (kg)
Logistic Regression	0.8298 ± 0.0430	0.64	13.59	0.000000
k-NN	0.7189 ± 0.0489	0.19	5.43	0.000000
Decision Tree	0.6374 ± 0.0356	1.37	4.35	0.000001
Random Forest	0.8055 ± 0.0483	9.66	4.46	0.000005
MLP	0.8308 ± 0.0392	7.29	5.78	0.000004

Table 1

4.2. Discussion

The best performing model (in terms of accuracy) was the MLP (0.8308), closely followed by Logistic Regression (0.8298). This suggests that the GenreAll descriptors are linearly separable, making Logistic Regression a very strong baseline despite its simplicity. Random Forest also performed well (0.8055), confirming that ensemble tree-based methods can capture non-linear relationships in the feature space.

In contrast, the Decision Tree and k-nn models achieved lower accuracy. The Decision Tree is particularly prone to overfitting in high dimensional settings, which likely shows its weaker generalization performance. k-NN, although “efficient”, can struggle in high dimensional spaces due to its dimensionality.

Regarding computational cost, Logistic Regression offers the best trade-off between accuracy and efficiency, with relatively low runtime and modest memory requirements.

Overall, Logistic Regression is a strong choice when both performance and efficiency are considered.

5. TASK 2: RESULTS AND DISCUSSION

In Task 2 the evaluation was repeated on two alternative feature sets: 30 second aggregated features and 3 second frame level features. These representations differ in temporal granularity and dataset size, which has a strong impact on both classification performance and resource usage.

5.1. Task 2a: 30 second features

Table 2 summarizes the performance of the models on the 30 second feature representation.

Model	Accuracy (mean ± std)	Time (s)	Peak Memory (MB)	CO_2 (kg)
Logistic Regression	0.7210 ± 0.0399	0.41	4.17	0.000000
k-NN	0.6790 ± 0.0421	0.13	1.37	0.000000
Random Forest	0.7170 ± 0.0415	16.21	1.12	0.000009

Table 2

5.2. Task 2b: 3 second features

Table 3 summarizes the performance of the models on the 3 second feature representation.

Model	**Accuracy (mean ± std)**	**Time (s)**	**Peak Memory (MB)**	**CO_2 (kg)**
Logistic Regression	0.7240 ± 0.0175	1.94	50.25	0.000001
k-NN	0.8932 ± 0.0084	0.29	13.07	0.000000
Random Forest	N/A	93.03	7.83	0.000049

Table 3

The results for Task 2 highlight the importance of time granularity. On the 30 second features, all models achieve moderate accuracy (around 0.70 - 0.72). Logistic Regression and Random Forest perform similarly, while k-nn stays slightly behind.

On the 3 second features performance improves substantially, particularly for k-nn, which reaches an accuracy of 0.8932 (the highest across all experiments). The short segments increase the number of training samples and seem to preserve local temporal structure, which benefits approaches like k-nn. Logistic Regression also benefits from the richer dataset but cannot match k-nn on this representation.

Random Forest, while potentially powerful, become extremely expensive (computational-wise) on the 3 second dataset, with a runtime of over 90 seconds and the highest estimated CO2 emissions. This illustrates poor scalability of tree based ensembles with respect to dataset size, especially when cross-validation is applied.

6. GENERAL CONCLUSSIONS

In both tasks, the following conclusions can be drawn:

1. Logistic Regression offers an excellent balance between accuracy and computational efficiency. This is shown with the GenreAll dataset.
2. The MLP achieves the best accuracy in Task 1, suggesting non linear relationships exist in the feature space.
3. K-nn is the strongest performer on the 3 second segmented features. It benefits from the bigger amount number of samples and local structure.
4. Random Forest often performs well but scales poorly in terms of runtime and energy consumption on large datasets.

Overall, the choice of model and feature representation in a real case scenario will depend on the available computational budget and the objective. For resource efficient classification, Logistic Regression is recommended, while k-nn on short segments is preferable when maximizing accuracy is the main objective.