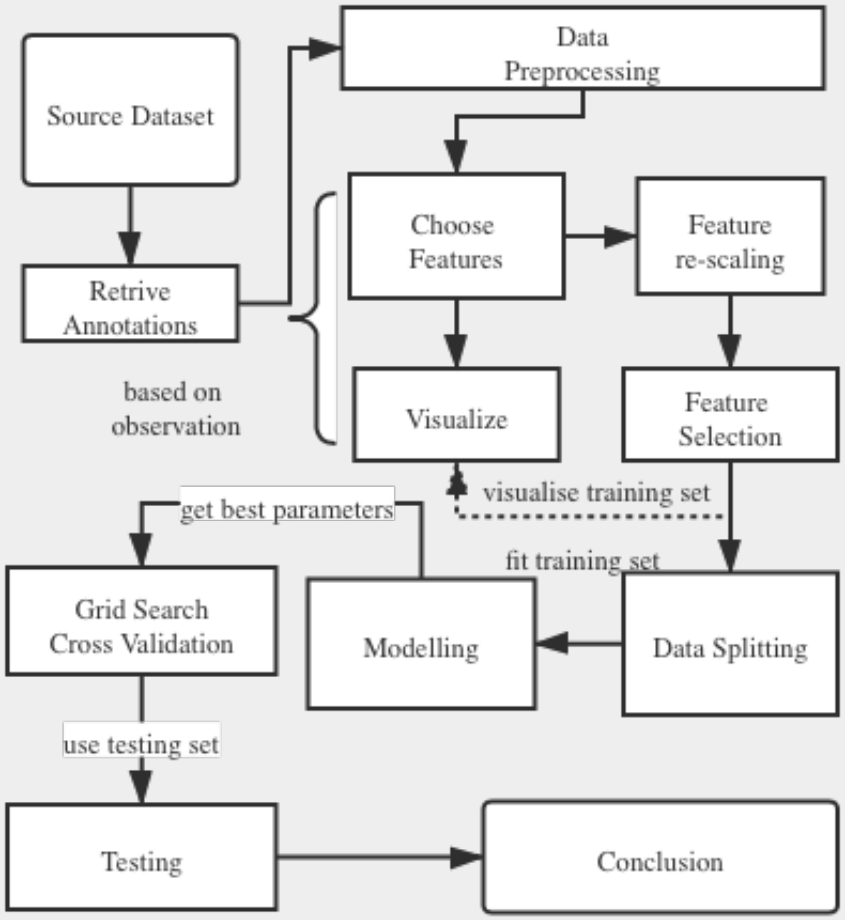
Group5-CMLS-HW1

Music Genre Classification

{ 10482867 10521088
10539533 10702368
10751919 } @mail.polimi.it

Project Workflow



Based on a subset of 400 audio pieces from GTZAN dataset

- We build Model1: **KNN** Classifier, and Model2: **SVC** Classifier
- Able to predict unseen audios among **Blues, Metal, Hip-hop and Reggae**

Preprocessing

- retrieve annotations
- check original dataset
- split the dataset into training/test (80% vs. 20%)

Observe the Dataset

- have a look at the data
- check the quality

Genre	Rec./Exact Repetition	Unbalance	Version	Mislabelling
Blues	–	only 9 artists	–	–
Hip-hop	6	–	2	4
Metal	–	–	3	13
Reggae	3	30 % from one artist	2	1

Table 1: A Summary of the known errors in the dataset.

The Features

- 34 columns of feature
- 400 audio tracks
- 80 % for training
- 20 % for testing
- rescaling, selection, saving

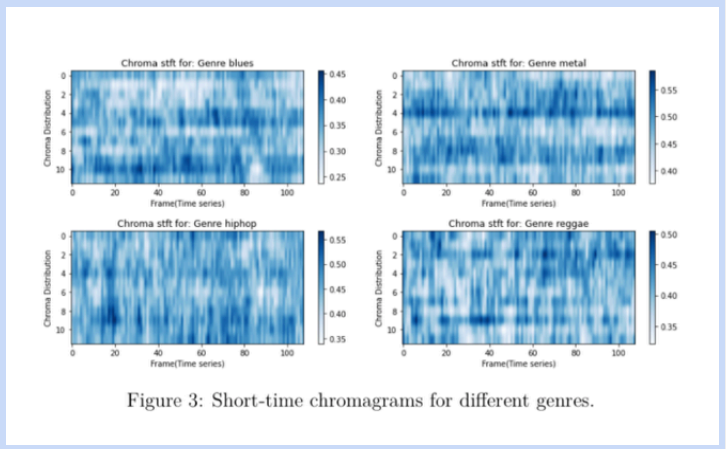
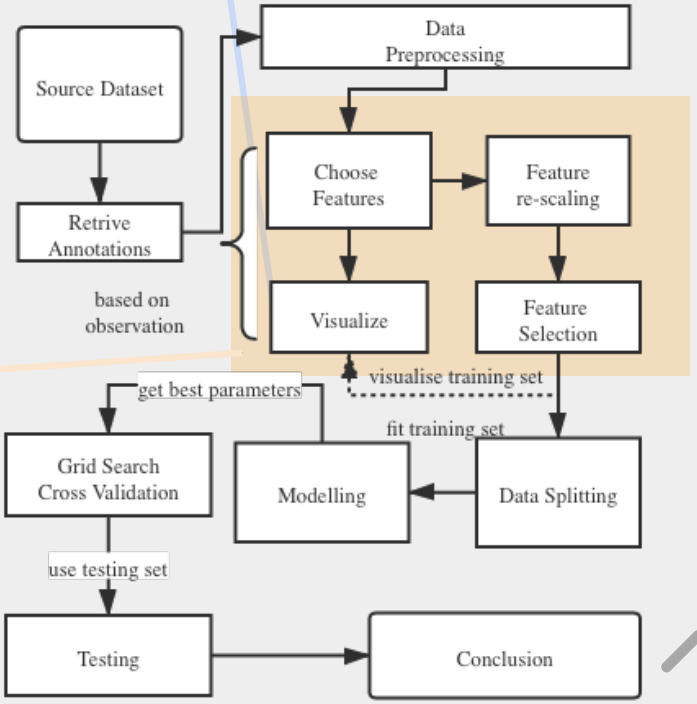


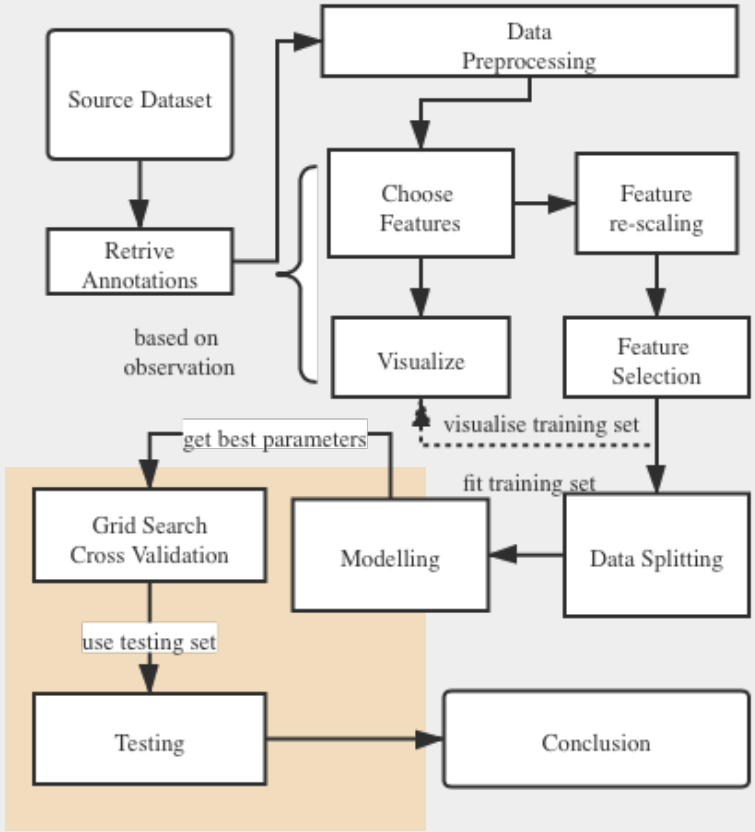
Figure 3: Short-time chromagrams for different genres.

	mfcc1	mfcc2	mfcc3	mfcc4	mfcc5	mfcc6	mfcc7	mfcc8	mfcc9
0	-209.826150	124.463215	10.342456	47.010909	6.508127	15.410516	-2.005106	6.706564	-21.557632
1	-107.170265	88.488187	-4.169413	55.477715	-8.013872	38.415162	-6.410026	28.572667	-19.821608
2	-89.354890	93.800390	-14.560049	54.388645	-33.502993	19.423905	-19.761012	13.031584	-13.861511
3	-238.951198	108.705427	34.145849	30.211149	-0.502583	27.292958	-21.399484	14.171864	-24.512312
4	-346.969336	139.273043	24.147860	22.209932	18.703974	11.669042	9.033813	6.945858	-2.019918
...	...	...	...	...	...	...	...	...	...
395	-103.136673	67.205032	2.212647	28.504494	4.172197	11.874952	14.244765	14.292828	3.261752
396	-132.977565	117.143566	-2.825532	22.537389	25.921838	0.278448	13.535487	3.408554	4.533075
397	-164.594570	142.707450	-41.489226	41.226193	16.171186	-5.588592	7.633069	5.620200	0.810726
398	-169.175089	110.450371	-8.545039	43.888601	0.316973	26.636772	-14.361243	13.035166	-11.153996
399	-138.122043	101.482168	-35.343779	36.139295	-9.510810	14.766630	-0.240599	11.524783	-2.105261

400 rows × 34 columns



Two Classifiers: KNN & SVC



- The process to use the classifiers: pick the right parameters --> build the classifiers -->fit with training data set
- We use K-fold (k = 10) Cross Validation and Grid Search method to optimise the parameters for those two classifiers

Classifier	parameters	best score
KNN	p:1, k:5	0.8375
SVC	'C':10, 'kernel':'rbf', 'gamma':0.001	0.797(+/-0.126)

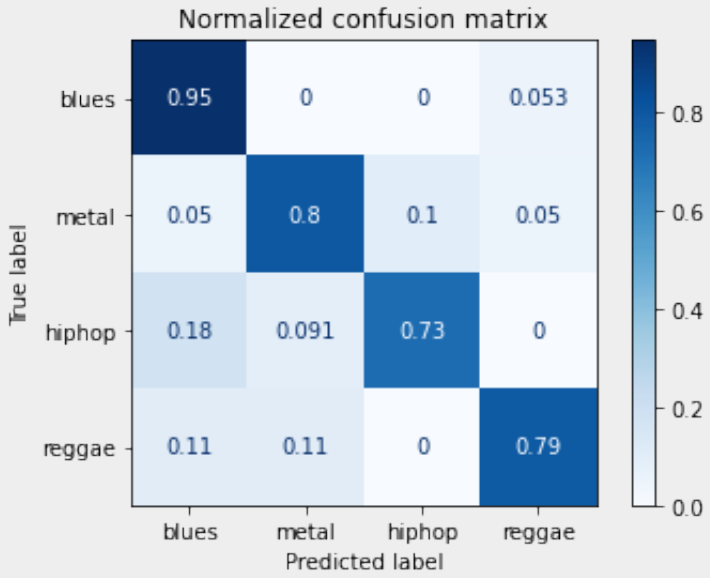
Table 2: Grid Search to find best parameters

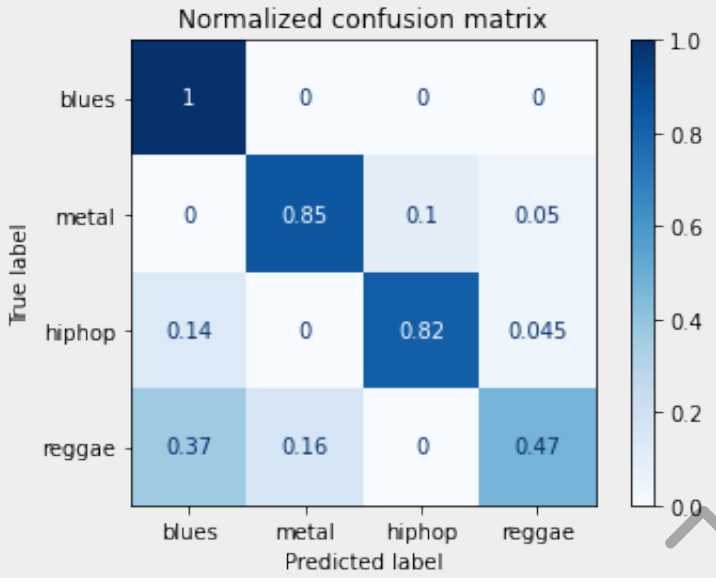
Results & Conclusions

- Here are two confusion matrix to represent the final testing result, separately for KNN and SVC
- Each row in the matrix is the real genre, while each column is the predicted genre category classified by the model we build

Table 3: Final Results

Classifier	Accuracy	F1 Score
KNN	0.8125	0.812
SVC	0.785	0.79





Plot 4. KNN & SVC results of Confusion Matrix

Reference

- [1]**Data Sets. GTZAN Genre Collection.**url:http : / / marsyas.info/downloads/datasets.html.
- [2]**LibROSA Documentation.**url:https://librosa.github.io/librosa/.
- [3]**NumPy.**url:https://numpy.org/.
- [4]**Pandas documentation.**url:https://pandas.pydata.org/pandas-docs/stable/index.html.
- [5]**Scikit-learn.**url:https://scikit-learn.org/stable/.
- [6] Bob L. Sturm. **"The GTZAN dataset: Itscontents, its faults, their effects on eval-uation, and its future use"**. In:CoRRabs/1306.1461 (2013). arXiv:1306.1461.url:http://arxiv.org/abs/1306.1461.
- [7] **G. Tzanetakis and P. Cook. "Musicalgenre classification of audio signals".** In:IEEE Transactions on Speech and AudioProcessing10.5 (2002), pp. 293–302
- [8] Final Detailed Report, **Music Genre Classification**, https://yilin10.github.io/MusicalGenreClassification/final.pdf
- [9] Final Google Colab IPython Notebook, **Music Genre Classification**, https://github.com/yilin10/MusicalGenreClassification/blob/master/Assignment1.ipynb