Summary Music Genre Classification using ML Techniques 2018

- Relevance of the task : useful for iTunes, Spotify…
- Deep learning models are becoming popular : main advantage is that they do not need hand-crafted features
- Dataset : Google's *Audio Set*
- 7 Genres : Pop, Rock, Hip Hop, Techno, Blues, Vocal, Reggae with between 3k and 8k samples in each → total 40k samples, 34 GB of data
- Pre-emphasis filter is used to reduce noise
- Data is represented using a spectrogram : Time in x-axis, Frequency in y-axis, Color indicating the intensity. Obtained by STFT (Short Time Fourier Transform). It gives a visual representation of the signal → image classification algorithms can be used
- Uses VGG-16 (winner of the ImageNet Challenge 2014)
- Preventing overfitting : L2-regularization term in the loss function, dropout of certain neurons (neurons are randomly discarded on certain iterations of the training, to prevent the algorithm from relying too heavily on a small subset of neurons)
- Dataset split between training (10%), validation (5%), test (5%)
- GPU used for processing
- Algorithms used : Convolutional Neural Networks, Baseline Feed-forward Neural Networks, and classical ML algs using hand-crafted features
- Hand-crafted features :
  Central moments (first 4 moments of the amplitude of the signal), Zero Crossing Rate, RMS Energy (calculated per frame, then the average and sigma are taken), BPM, MFCC, Chroma (pitch), Spectral Centroid, Spectral BW, Spectral Contrast, Spectral Roll-off

- **Classifiers** : Logistic Regression (one-vs-rest), Random Forest, (Extreme Gradient) Boosting, SVMs (one-vs-rest)
- **Evaluation Metrics :** Accuracy, F-Score, AROC
- **Best ML classifiers :** SVMs and Boosting
- **Most important features :** MFCCs and Spectral Contrast
- When retaining **only a few features** (the best ones), the results are only slightly lower than with all the features → similar conclusion as the article "Comparative study on content-based…"
- Experiment with either ONLY time-domain features or ONLY frequency-domain features → **frequency-based features always provide better results**, and adding time-based features only **very slightly** improves the model
- Generating **confusion matrices** (matrix representing when a song of genre *i* was labeled as genre *j*) is useful to know where the model has the hardest time
- **Ensemble learning** (combining classifiers) can improve performance of individual algorithms when the classifiers take different inputs (here, CNN takes images as input whereas Boosting takes hand-crafted features, and their combination achieves better scores)
- According to the author, a good improvement to make is finding a good preprocessing of the data to reduce noise