Machine Learning Approaches to Musical Genre Classification

Adam McMillan, May 2025

1. Introduction

What defines a *musical genre*? Genre is a label that humans give to music that categorises and describes; based on a fuzzy set of musical properties such as timbral texture, rhythmic structure, pitch content, harmonic progression, and instrumentation, to name a few. Many pieces of music can be challenging to categorise even for humans, with human performance at the task of genre classification sitting around 70% accuracy (Tzanetakis & Cook, 2002). This can be due to composers employing blends of stylistic elements and eclectic sound palettes, as well as due to the simple fact that musical perception involves subjectivity (Tzanetakis & Cook, 2002).

1.1 Dataset description

Of great use to continued research in this area is the Free Music Archive (FMA) (Defferrard et al., 2017), a publicly available dataset, containing 106,574 tracks from 16,341 artists, with metadata provided for each track including genre classification using a hierarchical taxonomy of 161 musical genres. Prior to FMA, research relied on smaller datasets; as such, the arrival of FMA as a data source has provided a new source of momentum for open research into genre classification and other areas of audio signal processing. There are several size options for this dataset; for the purpose of this study, the medium dataset (25,000 tracks of 30s, 16 unbalanced genres 22 GiB) was utilised.

2. Literature Overview

Machine learning approaches to genre classification have existed for over twenty years, and are relevant to recommendation systems, playlist generation, library organisation, and content tagging by (Tzanetakis & Cook, 2002). As music platforms continue to grow in sophistication and user-base, genre classification remains a highly relevant area of research.

An early work by Tzanetakis and Cook (2002) proposed one of the first approaches to genre classification, based on the use of timbral texture, rhythmic content, and pitch-based features extracted from audio samples. They used KNN and Gaussian classification approaches and were able to achieve moderate success in classification accuracy. Subsequent work explored a wider range of audio signal features, such as Mel-frequency cepstral coefficients, chroma features, spectral centroid, and spectral contrast (Bergstra et al., 2006). Classifiers such as support vector machines, random forests, and ensemble methods, among others, have demonstrated strong performance in the traditional feature-engineering approach (Bahuleyan, 2018). More recently, research has explored neural network approaches using audio spectrograms, however these methods are highly computationally intense and require a large amount of data. Thus, there is a continued need to evaluate performant, lightweight approaches to musical genre classification.

3. Methods

3.1 Pre-Processing

Several genres were excluded from the FMA dataset, due to insufficient samples in the medium sized dataset that was used. These were Easy Listening, Old-Time/Historic, Country, Soul-RnB, Spoken, Blues. Samples that did not contain a top-level genre label were pruned from the dataset. The remaining genres were: Classical, Electronic, Experimental, Folk, Hip-Hop, Instrumental, International, Jazz, Pop and Rock. To achieve greater balance between genres, the dataset was pruned such that at most 600 samples remained (chosen randomly) for each genre. All genres except Jazz were downsized to 600; since Jazz was a slightly smaller sample to begin with, it had all ~350 samples retained in the final dataset. The final dataset used for feature extraction and model evaluation contained 5782 samples.

3.2 Feature Construction

Features were constructed using the raw audio data for each 30s audio file, and with the use of *librosa*, a freely available library for audio signal processing in Python (McFee et al., 2015). Table 1 provides a brief description of each feature extracted from the audio using librosa. The choice of features was taken as a unique combination of feature sets utilised in previous research (Bahuleyan, 2018; Bergstra et al., 2006, Tzanetakis, 2002, Velazquez et al., 2024), in order to explore the utility of more novel feature combinations. For almost all features, summary statistics were used (e.g. mean, std) in line with previous research (Bahuleyan, 2018). For example, MFCC was configured with 13 coefficients, as is standard practice (Bahuleyan, 2018); resulting in a two-dimensional result containing an array of values for every 512-byte frame, for each of the 13 coefficients. As such, for MFCC, 13 means and 13 standard deviations were calculated using array data across all frames for each coefficient. In simpler, one-dimensional cases, only one mean and standard deviation needed to be calculated for the entire sample.

Table 1. Features extracted from audio signal

Domain	Feature Name	Derived Statistics	Count	Brief Description
Time Domain	Amplitude	Mean, Std, Skew, Kurtosis	4	Basic statistical moments of the raw audio waveform amplitude
	Zero Crossing Rate	Mean, Std	2	Rate of sign changes in the waveform, indicating frequency content
	RMS Energy	Mean, Std	2	Root mean square energy per frame
	Tempo	Single value	1	Estimated tempo (beats per minute) of the sample
	Tempogram Ratio	Mean, Std	26	Rhythm periodicity features based on tempogram ratios
Frequency Domain	MFCC	Mean, Std	26	Mel-frequency cepstral coefficients capturing spectral envelope
	Spectral Contrast	Mean, Std	14	Differences between spectral peaks and valleys
	Spectral Centroid	Mean, Std	2	Centre of spectral mass, indicating brightness
	Spectral Bandwidth	Mean, Std	2	Spread of the spectrum around the centroid
	Spectral Rolloff	Mean, Std	2	Frequency below which 85% percentage of energy is contained
	Chroma STFT	Mean, Std	24	Chroma features from short-time Fourier transform
	Chroma CQT	Mean, Std	24	Chroma features from constant-Q transform
	Chroma CENS	Mean, Std	24	Chroma Energy Normalised features (more robust to dynamics)
	Tonnetz	Mean, Std	12	Tonal centroid features based on harmonic relationships
Total			165	

3.3 Machine Learning Algorithms

Three machine learning algorithms were selected for evaluation and comparison: logistic regression, support vector machine (SVM), and *k*-nearest neighbours (KNN). Data was separated into stratified train and test sets, with a test set size of 15%. All models were trained and evaluated within a 10-fold cross-validation framework (with 80/20 split) which was stratified to ensure balanced class distribution in within folds. Hyperparameter tuning was performed within cross-validation using sklearn's GridSearchCV, which enables easy testing of all combinations of supplied parameters, in order to identify parameter value combinations that lead to superior model performance.

Logistic regression was evaluated using a one-vs-rest strategy in order to handle multi-class classification. It was nested within a pipeline that included scaling and *k*-best feature selection, with *k* values of 135, 145, 155, and 165 (the latter representing the full feature set). The regularisation hyperparameter *C*, which affects the balance between underfitting and overfitting, was tuned over values 0.1, 1, and 10.

Support vector machine was evaluated using a radial basis function (RBF) kernel, as has been employed in past research (Bahuleyan et al., 2018) and was also nested within a pipeline that included scaling and *k*-best feature selection using the same *k* values as were used in logistic regression. The regularisation parameter *C* was tuned over 1, 5, 10, and 15 to control the trade-off between underfitting and overfitting. Additionally, the kernel coefficient gamma was set to either *scale* or *auto* to in order to manipulate the influence of data points on decision boundary learning.

Finally, *K*-nearest neighbours was evaluated, with varying numbers of neighbours (3, 5, 7, and 9), two weighting approaches (*uniform*, and *distance*) in order to evaluate the informativeness of proximity, and three distance metrics (*Minkowski*, *Euclidean*, and *Manhattan*) in order to determine which would best capture similarity between data points. In contrast to the previous models, feature selection (i.e. *k*-best) was not applied for KNN, as preliminary trials suggested that this led to poorer performance, possibly due to KNN's sensitivity to the structure of the feature space and distance calculations.

4. Results

Figure 1 presents the confusion matrices for each of model, based on their best-performing hyperparameter configurations, as determined by GridSearchCV with macro-averaged F1 score as the selection criterion. Macro-averaged F1 was used as it can help handle imbalanced classes. The confusion matrices provide a visualisation of each models ability to classify genres correctly (with correct classifications appearing on the diagonals), as well as incorrectly (remaining cells).

Table 2 summarizes key evaluation metrics: accuracy, macro-averaged F1 score, precision, and recall, for each model, using predictions made on the held-out test set. The table also includes optimal parameters combinations determined by GridSearchCV. Accuracy represents the overall proportion of correct predictions across all classes. Precision (macro-averaged) reflects the average ability of the classifier to avoid false positives across classes. Recall (macro-averaged) captures the model's ability to correctly identify instances from each genre class (i.e. to minimise false negatives). F1 score (macro-averaged) provides the harmonic mean of precision and recall and as mentioned, is quite useful in evaluating performance on imbalanced datasets.

Among the models tested, SVM achieved the highest F1 score (0.57), and highest accuracy score (0.57), indicating superior classification ability and generalisation of the model. Figure 2 illustrates relative feature importance in the best performing SVM model. In all models, heterogeneity in classification accuracy across genres was apparent; this will later be discussed.

Figure 1. Confusion matrices for each model

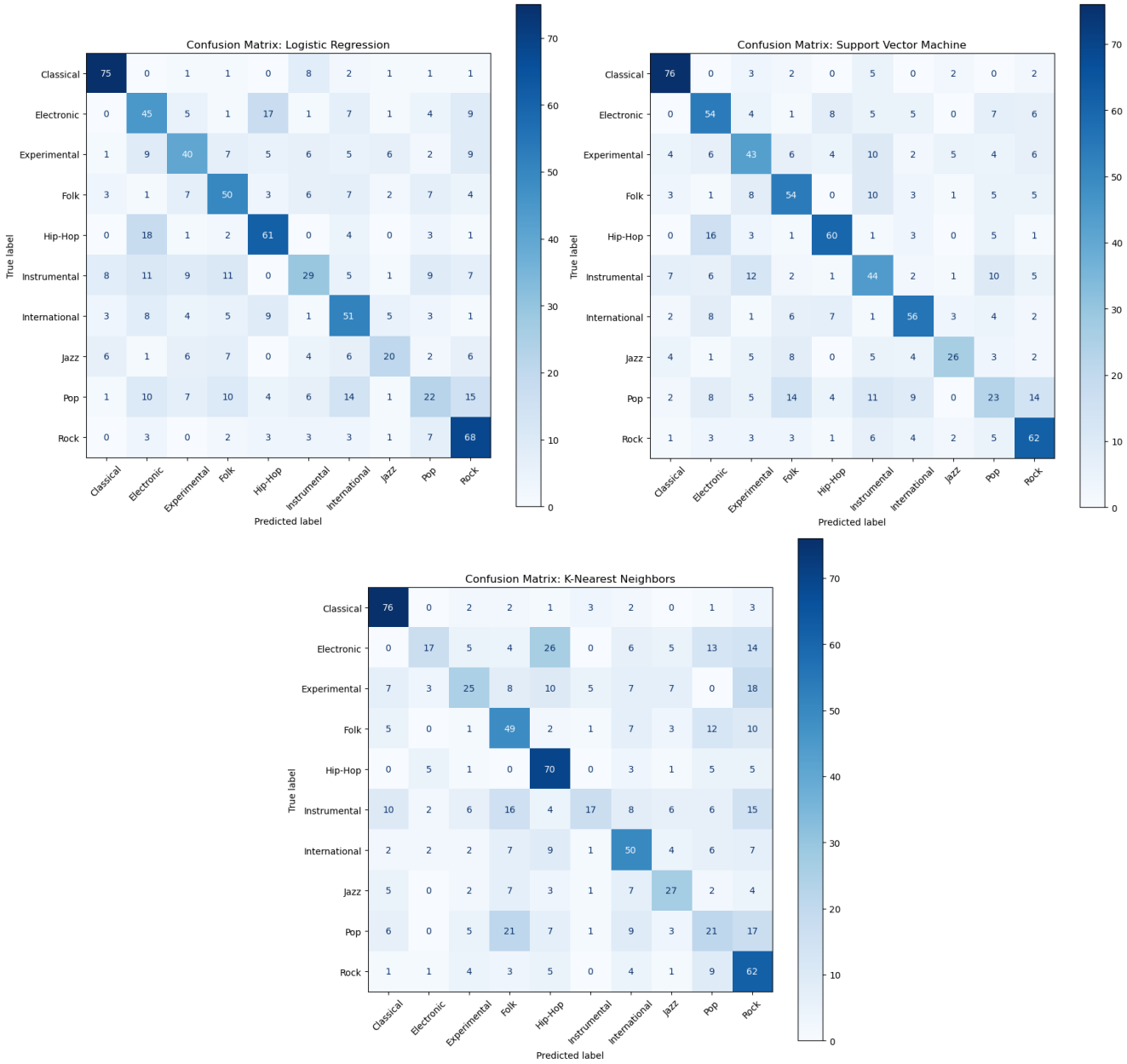

Figure 2. Relative feature importance in best-performing SVM

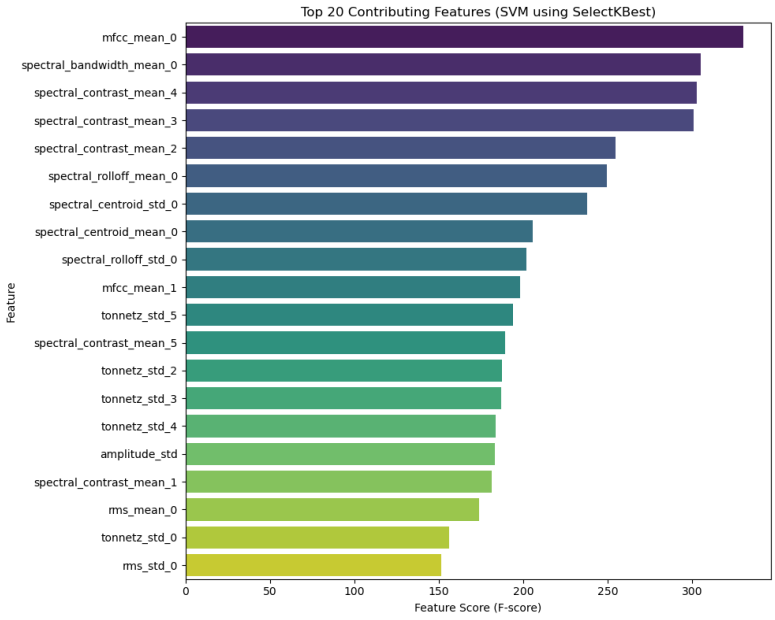

Table 2. Evaluation metrics for each model

Model	Parameters	Accuracy	Precision	Recall	F1-Score
Logistic Regression	*k*=135, *C*=1	0.53	0.52	0.52	0.52
Support Vector Machine	*k*=145, *C*=5, *gamma*=scale, *kernel*=rbf	**0.57**	0.57	0.57	**0.57**
K-Nearest Neighbours	*metric*=manhattan, *n-neighbours*=7, *weights*=distance	0.48	0.49	0.48	0.45

5. Discussion

5.1 Model Comparison

The results demonstrate that although all three models were able to perform genre classification to a reasonable degree, SVM was the best performing model across all metrics, with an F1 score and accuracy score of 0.57. Logistic regression followed somewhat closely behind, and KNN performed the poorest.

SVM likely performed the best as it is well adept in handling high-dimensional data, and due to its ability to find non-linear decision boundaries by using an RBF kernel. Although KNN is also able to find non-linear boundaries, SVM is more robust to overfitting when regularised properly, and KNN suffers from numerous issues such as distortion due to less-informative features, and a lack of clear value behind spatial "closeness" in high-dimensional data. Additionally, KNN may generalise poorly beyond observed data and as a lazy learner does not learn/retain any feature weights. Logistic regression performed slightly worse than SVM, although not by a much; this suggests that some linearly separable relationships exist in the feature space. However, logistic regression is unable to capture more nuanced, non-linear boundaries in genre separation.

5.2 Alternative Approaches

An alternative modelling approach exists in the literature on genre-classification, using spectrograms derived from audio sources as inputs to convolutional neural networks. Essentially, for each audio sample, a spectrogram image can be constructed, which represents frequency and amplitude graphically across time; and these images can be fed into a CNN. This approach was considered here, but due to heavy computational requirements was not pursued.

Naïve Bayes was also considered as a potential modelling approach but was not pursued given the highly inter-dependent nature of the feature space and NB's conflicting assumption of feature independence. Although the features included in the study are based on distinct manipulations of audio signal, they inherently contain significant shared information and thus violate this assumption.

5.2 Genre-wise Performance

Even in the strongest model, SVM, observation of the confusion matrix in Figure 1 highlights particular confusion within the model surrounding Pop and Jazz classification. For example, Pop was commonly misclassified as Rock or Folk. To an extent, some confusion in the model is to be expected, due to the inherently fuzzy nature of musical genres discussed earlier. However, it is also possible that for these genres, additional/alternative features may need to be considered in order to provide greater distinction. A possible way to approach this is to consider practical differences between these genres, such as vocal style and instrumentation, and make use of features that provide some approximation of these characteristics. For example, providing additional versions of each feature with and without voice isolation may enhance classification ability.

6. Conclusion

This study explored machine learning approaches to music genre classification, using three traditional ML techniques. SVM proved to be the superior model of the three examined, likely due to its ability capture complex, non-linear relationships in high-dimensional space, while achieving some protection from overfitting issues through regularisation. Considering that SVM achieved an F1-score of 0.57, and that human ability to classify genre sits around 70% accuracy, there is room for improvement. CNN approaches using audio derived spectrogram images is a promising avenue for further research, as well as ensemble methods that may combine the strength of conventional methods such as SVN with the power of neural network approaches (Bahuleyan, 2018). Musical genre is a fuzzy, ever-evolving, human-dictated construct, and machine learning approaches to genre classification should continue to work with and carefully consider this limitation. The subject of genre classification continues to have real-world utility within recommendation systems, music library organising systems, content tagging, and more.

References

Bahuleyan, H. (2018). *Music genre classification using machine learning techniques*. arXiv. https://doi.org/10.48550/arXiv.1804.01149

Bergstra, J., Casagrande, N., Erhan, D., Eck, D., & et al. (2006). Aggregate features and ADABOOST for music classification. *Machine Learning, 65*(2-3), 473–484. https://doi.org/10.1007/s10994-006-9019-7

Defferrard, M., Benzi, K., Vandergheynst, P., & Bresson, X. (2017). *FMA: A dataset for music analysis*. In *Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR)*. https://arxiv.org/abs/1612.01840

McFee, B., Raffel, C., Liang, D., Ellis, D. P. W., McVicar, M., Battenberg, E., & Nieto, O. (2015). *librosa: Audio and music signal analysis in Python*. In Proceedings of the 14th Python in Science Conference (pp. 18–25).

Tzanetakis, G., & Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing, 10*(5), 293–302. https://doi.org/10.1109/TSA.2002.800560

Velazquez, O., Oropeza, J., & Fuentes-Pineda, G. (2024). *Enhancing music genre classification using Tonnetz and active learning*. Paper presented at MICAI 2024, Puebla, Mexico.