ANALYSIS BY CLASSIFICATION: A COMPARATIVE STUDY OF ANNOTATED AND ALGORITHMICALLY EXTRACTED PATTERNS IN SYMBOLIC MUSIC DATA

Iris Yuping Ren
Utrecht University
y.ren@uu.nl

Anja Volk
Utrecht University
a.volk@uu.nl

Wouter Swierstra
Utrecht University
w.s.swierstra@uu.nl

Remco C. Veltkamp
Utrecht University
r.c.veltkamp@uu.nl

ABSTRACT

Musical patterns are salient passages that repeatedly appear in music. Such passages are vital for compression, classification and prediction tasks in MIR, and algorithms employing different techniques have been proposed to find musical patterns automatically. Human-annotated patterns have been collected and used to evaluate pattern discovery algorithms, e.g., in the Discovery of Repeated Themes & Sections MIREX task. However, state-of-the-art algorithms are not yet able to reproduce human-annotated patterns. To understand what gives rise to the discrepancy between algorithmically extracted patterns and human-annotated patterns, we use jSymbolic2 to extract features from patterns, visualise the feature space using PCA and perform a comparative analysis using classification techniques. We show that it is possible to classify algorithmically extracted patterns, human-annotated patterns and randomly sampled passages. This implies: (a) Algorithmically extracted patterns possess different properties than human-annotated patterns (b) Algorithmically extracted patterns have different structures than randomly sampled passages (c) Human-annotated patterns contain more information than randomly sampled passages despite subjectivity involved in the annotation process. We further discover that rhythmic features are of high importance in the classification process, which should influence future research on automatic pattern discovery.

1. INTRODUCTION

Patterns occur in many dimensions of life: we constantly look for patterns to classify and predict based on our experience [40]. In music, composers employ patterns to induce structures to their music [14]; listeners look for patterns while they listen attentively [16, 19]; performers learn patterns to better memorise, perform and improvise [39]; musicologists use patterns as evidence for categorisation and theorisation [1, 23]. In this paper, we work mainly with repeated patterns which characterise and categorise folk songs.

Because of the many potential applications of musical patterns, algorithms that can automatically identify patterns are useful in many contexts. Automatic pattern discovery is an active research area in which many different methods have been developed, such as string-based approaches [5, 8, 17, 21, 22, 32], geometric approaches [4, 7, 29, 41], data mining approaches [6, 36], and machine learning approaches [34, 46].

One open question is how one should evaluate the quality of algorithmically extracted patterns. One common approach is to compare the extracted patterns with human-annotated patterns [2, 11, 15]. However, because of the aforementioned versatile application possibilities and diverse definitions of musical patterns, we face several challenges using human-annotated patterns to evaluate the algorithms. First, there is a lack of human-annotated pattern datasets in general [37]. Second, subjectivity and irreducible human errors could be introduced in the annotation process [27]. Third, it is not straightforward to see what metrics one should compute to compare the human-annotated patterns with automatically extracted patterns.

Previous research has addressed these challenges to a certain extent. Historically, algorithms have been tested on unassociated datasets with disparate metrics [15]. One attempt to standardise the evaluation of algorithms is the MIREX Discovery of Repeated Themes & Sections task initiated in 2014. In the task, a pattern is defined as a set of time-pitch pairs that occurs at least twice in a piece of music and the JKU-PDD dataset was introduced [11]. According to the evaluation metrics in this task, the state-of-the-art algorithms perform acceptably well in precision, recall, and F1-scores, although they cannot reproduce the human-annotated patterns yet. Another pattern annotation dataset which has been used for evaluating the algorithms is the MTC-ANN Dutch Folk Song dataset [43]: human-annotations have been compared with algorithmically extracted patterns by their performance in a classification task [2] showing the annotated patterns perform better. Furthermore, a large disagreement between annotated and computationally extracted patterns has been shown in both the JKU-PDD and MTC-ANN dataset in [37].

The aim of this paper is to identify and analyse the discrepancy between human annotations and algorithmically extracted patterns. To achieve this goal, we extract characteristic features from human-annotated and automatically extracted patterns, and conduct a comparative study on the

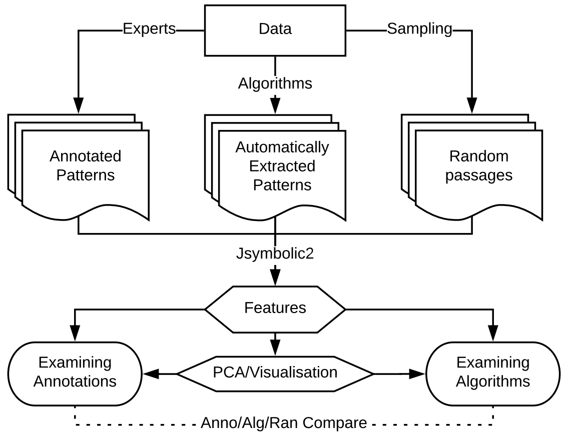

Figure 1. Pipeline of our experiments. Given the music data, experts annotate patterns, algorithms extract patterns, and we randomly sample passages in the corpus. Tasks are shown in rounded boxes. Diamond boxes are transformed data/features. Section 2 gives a detailed description.

pattern features using classification methods. To establish a baseline, we randomly sample passages that have the same lengths as human annotations. By performing a ternary classification task amongst the human-annotated patterns, algorithmically extracted patterns and random passages, we provide evidence that they are separable by classifiers. Despite taking musical patterns out of context and only considering the local structures annotated by humans and extracted by algorithms, the result of the experiment shows preliminary implications for the future design and evaluation of pattern discovery algorithms.

Contribution Using the monophonic MTC-ANN Dutch Folk Song dataset [43], our main contributions are: (a) By calculating features of human-annotated, automatically extracted and sampled passages, we summarise and visualise the distributions of patterns in the feature space using Principal Component Analysis (PCA) (b) Our classifiers successfully discriminate between human-annotated patterns and algorithmically extracted patterns above random chance level, which enables us to analyse what characterises the differences between human-annotated patterns, automatically extracted patterns and random passages (c) Based on the analysis of features and classification results, we propose several ways to improve pattern discovery algorithms.

Figure 1 is the pipeline of our experiments to be detailed in the next section. Abbreviations such as **Anno** (Annotations), **Ran** (Random passages), and **Alg** (Algorithmically extracted patterns) are used in tables and figures.

2. DATA PREPARATION

We use the MTC-ANN Dutch Folk Song dataset [43], which contains an exceptionally large number of annotated patterns and is therefore suitable for a classification experiment. In this section, we examine groups of patterns, random passages, and their features in this dataset.

Algorithm	#Pattern	#Occurrences	Incl.
(Annotation)	153	1657	✓
SIAR	893	5576	✓
SIAP	250	3650	✓
SIAF1	822	5308	✓
VM	182	25679	✓
VM2	159	4658	✓
SC	126	355	✓
SCFP	200	724	✓
PatMinr	105663	182306	✗
ME	3339951	5651956	✗
MDGP	3543940	5457210	✗
COSIATEC	61499	99501	✗

Table 1. Algorithms and the count of extracted patterns. Abbreviation correspondence and details are given in Section 2.1. The counts of PatMinr, ME, MDGP, and COSIATEC are larger by serveral magnitude because we include a parameter sweep, while other algorithms use a parameter setting preset by authors of the algorithms. A comprehensive investigation into parameter settings of algorithms is not conducted in this paper.

2.1 Pattern groups in MTC-ANN

Annotated patterns During the making of MTC-ANN, three experts have been asked to annotate the prominent patterns in each song which best classify the song into one of 26 tune families. *Tune family* is a concept in ethnomusicology that groups together tunes sharing the same ancestor in the process of oral transmission [9]. The dataset consists of 360 Dutch folk songs with 1657 annotated pattern occurrences. In an annotation study on what influences human judgements when categorising melodies belonging to the same tune family, repeated patterns turned out to play the most important role [45]. It is, therefore, reasonable to use repeated pattern discovery algorithms on this dataset.

Patterns from algorithms Table 1 shows the number of extracted patterns from state-of-the-art musical pattern discovery algorithms that have been used and compared in previous research [2, 37]. The count numbers for PatMinr [22], MotivesExtractor (ME) [32], MDGP [8], and COSIATEC [26] include different parameter settings of the algorithm and are therefore several magnitudes larger than other entries. We do not include these patterns because a comprehensive parameter search of the algorithms would be out of the scope of this paper. For the same reason, although algorithms such as SIATECCompress - TLP (SIAP), SIATECCompress - TLF1 (SIAF1), SIATECCompress - TLR (SIAR) are not optimised for MTC-ANN, a parameter search is not conducted.

We use the seven pattern discovery algorithms and extract the patterns from the MTC-ANN dataset using the same setup as in [2, 37]. The extracted patterns from each algorithm form a subgroup under the umbrella of the extracted pattern group. The seven algorithms were submitted to the MIREX task during 2014-2017: SIATECCompress - TLP (**SIAP**), SIATECCompress - TLF1 (**SIAF1**), SIATECCompress - TLR (**SIAR**) [28], **VM** & **VM2** [44], SYMCHM (**SC**) [35], and SIARCT-CFP (**SIACFP**) [7].

Sampling random passages We compare annotated and extracted patterns with randomly sampled passages as a baseline in order to potentially support or refuse the significance of musical patterns. In more detail, taking the annotated patterns from MTC-ANN, random passages are sampled with the following procedures: for each annotated pattern, we find the corresponding song where the annotation appears. We then find a random starting point and take an excerpt of the same length as the pattern to construct a candidate excerpt. Finally, we repeat the sampling procedures five times to prevent accidental results.

2.2 Compute features

Much work of research exists on how to design and compute musical features. As we are concerned with repeated patterns, and there are many possibilities as to what features make a pattern repetitive [42], we hence adopt a standardised feature extraction process as described below.

Feature Calculation We calculate features from the patterns by using a common feature extraction tool: the jSymbolic2 toolbox in the jMIR toolset [25]. jSymbolic2 takes MIDI files as input and computes 155 musically meaningful features in six categories: texture, rhythm, dynamics, pitch, melody and chords. After computing all the features for all the patterns, we have a feature vector of 155 dimensions associated with each pattern. Another well-known feature extraction package, the FANTASTIC toolbox [30] is not used because it cannot process input of short length, which excludes valuable annotated patterns from contributing to subsequent classification tasks.

Feature Selection We perform a feature selection step and retain 63 features by first eliminating the features which are constant across all patterns, such as Vibrato Prevalence, Average Range of Glissandos, and so forth. Next, we eliminate the features which are not relevant to the music content of time and pitch, such as the dynamics features and artefacts introduced by MIDI conversion.

PCA and Visualisation PCA is known to be a practical preprocessing step and visualisation tool for classification problems. PCA produces linear combinations of features which maximise variances in a given dataset and are suitable for visualising differences in data.

In Figure 2, we plot different groups and subgroups of patterns in a two-dimensional [1] PCA embedding of the feature space. We make four cross-group comparisons to show typical cases of how musical patterns distribute in the feature space spanned by the first two components of the PCA decomposition. The visualisation is generated by using the annotated patterns as training data to obtain the PCA embedding, then project random passages and patterns from different algorithms onto this PCA embedding space.

From the four snapshots we take from the musical pattern PCA feature space as shown in Figure 2, we make several observations: (1) Annotated patterns and random passages have an extensive area of overlap, which makes it impossible to find a linear classifier using the first two principal components of the annotated patterns, which in turn makes it nontrivial to differentiate the two groups of patterns as shown in the upper left figure. (2) SIAR patterns exhibit very different distribution from the annotated patterns and random passages as shown in the top right subfigure. Notice the annotated patterns concentrate at the top left corner. In this case, it is relatively easy to separate the long-tail area of the extracted patterns from the annotation area. By applying this observation and designing a filtering process, it could substantially improve the performance of the SIAR algorithm on MTC-ANN. (3) The overlap between the annotated patterns and extracted patterns is small in the bottom left figure. A linear classifier can be devised to separate the two groups of data using the first two principal dimensions of the annotated patterns. The extracted patterns of the SC algorithm have different features than the annotated patterns. (4) In the bottom right figure, we show all the heterogeneous patterns as extracted by algorithms, annotated by humans or randomly sampled in the same PCA embedding. Patterns extracted by algorithms of the same family, namely SIATECCompress - TLP (SIAP), SIATECCompress - TLF1 (SIAF1), SIATECCompress - TLR (SIAR), and SIACFP tend to share the same long-tail property, and therefore their performance on MTC-ANN can be improved by an extra filtering step as described above.

In summary, setting out from the visual examination and our observations above, it is promising to apply classification techniques to discriminate the features of different groups of patterns. We commence on the classification task and conduct a comparative analysis using the classification results in the next section.

3. METHOD CONFIGURATION

In this section, we introduce the classifiers and evaluation metrics we use for the classification task.

3.1 Classification

Supervised classification methods have been used extensively in MIR tasks such as genre classification and classifying geographically different corpora. In addition, comparative analyses using classification methods have been performed in many areas of research [10, 33]. To the best of our knowledge, using supervised classification for conducting comparative analyses have not been used with symbolic musical patterns. In this paper, we use supervised classification methods to differentiate human-annotated, algorithmically extracted and randomly sampled passages in MTC-ANN. By putting patterns into groups (the group of algorithmically extracted patterns, the group of annotations, and the group of random passages) and observing whether there are systematic differences on the group level, we gain a different perspective than using the metrics based on individual patterns, such as the precision, recall, and F1-score used in MIREX.

To prevent the results to be classifier-specific, we use a mixture of simple and more sophisticated, linear and nonlinear classifiers to perform the ternary classification task. We also use standard machine learning techniques to train and test classifiers: first, scaling and centering preprocessing steps are performed on all the features and PCA input;

[1] More visualisations can be found at `https://goo.gl/qmyxdh`

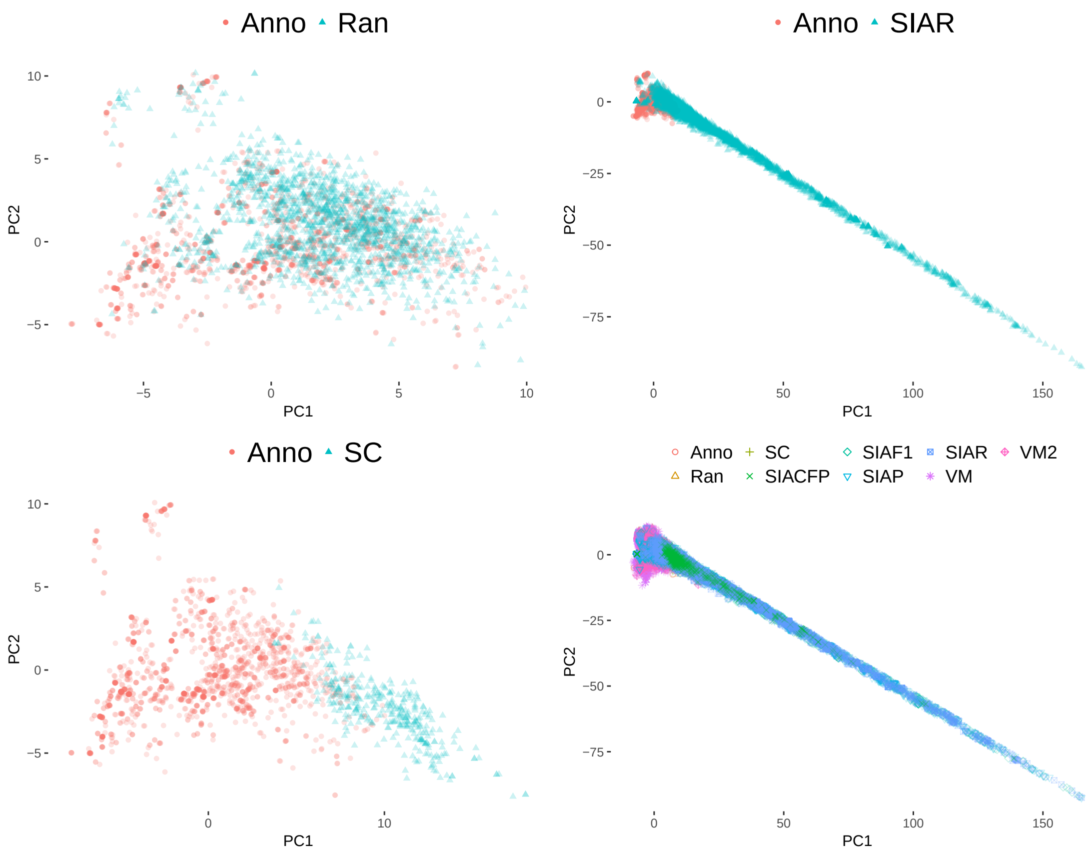

Figure 2. Visualisation of different groups of patterns using the space spanned by the first two principal components of the annotated pattern features. The legend denoting the colour correspondence with algorithms/annotated patterns/random passages is on the top of each subfigure. Notice that the scopes of figures in the left column are subregions of figures on the right. Notes on each subfigure: (1) Upper left: Random passages and annotated patterns. The overlap between the two groups is large, and it is nontrivial to separate them in this two-dimensional PCA embedding. (2) Upper right: SIAR patterns and the annotated patterns. SIAR patterns exhibit a long-tail behaviour which is not shared by the annotated patterns. (3) Bottom left: SC patterns and the annotated patterns. The overlap of the data points is small, which makes it easier to separate the two groups in this embedding. (4) Bottom right: Random passages, annotated patterns and patterns from all algorithms. We see some of the algorithmically extracted patterns are very different from the annotated patterns, and the algorithms belonging to the same family exhibit the same long-tail behaviour.

additionally, to avoid overfitting, for all experiments, we use a 10-fold cross-validation 3-times repetition scheme. The PCA projection and parameter search of each classifier are performed separately on each fold. The six statistical classifiers we use are:

GBM [13] (Gradient Boosting Machine) produces a prediction model consisting of an ensemble of decision trees. The parameters we search through are the learning rate, the complexity of trees, the minimum number of samples to commence splitting and the number of iterations.

LVQ [18] (Linear Vector Quantisation) applies a winner-takes-all Hebbian learning-based approach. We search through two parameters in this classifier: the codebook size and the number of prototypes.

LDA [38] (Linear Discriminant Analysis) produces a linear classifier which finds a linear combination of features that best separates different classes in datasets. This classifier does not contain parameters.

NB [31] (Naive Bayes) computes the conditional a-posterior probabilities of a categorical class variable given independent predictor variables using the Bayes rule. Three parameters are tuned for this classifier: the Laplace smoothing, kernel bandwidth and distribution type.

RF [3] (Random Forest) operates by constructing a multitude of decision tree. The parameter we consider is the number of variables per level.

SVM [12] (Support Vector Machine) calculates a map from data to a new representation so that the data points of the separate categories are divided by a gap that is as wide as possible. We use the radial basis function kernel and consider two parameters: the smoothing factor and the weight of training examples.

The experiments have been performed using R. The task takes about 2 hours on an i7 CPU with a maximum memory usage of 2Gb. For reproducibility, the data and code to replicate the experiments can be downloaded[2].

[2] https://github.com/irisyupingren/patdisISMIR2018

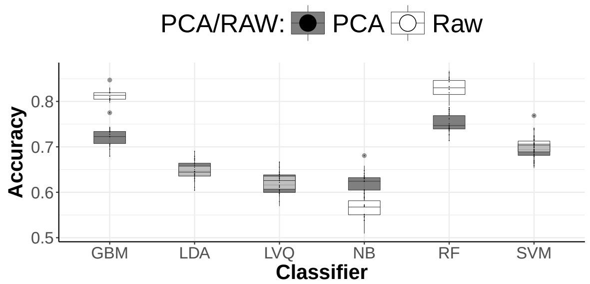

Figure 3. Accuracy values of classifiers in thirty experiments (10-fold cross-validation repeated three times) using six classifiers with jSymbolic2 features and features after PCA decomposition.

Other schemes with different parameters and with a new test set split were used, too, but are omitted because they give similar results to our analysis.

3.2 Evaluation

We mainly use accuracy and its variance as a measure of the performance of classifiers. To further interpret the results of the classification task, we compute confusion matrices and feature importance measures. Ten other metrics for each classifier are provided for further inspection [3]. In the next section, we report the most relevant results.

4. RESULTS AND DISCUSSION

In this section, we first report the model metrics of classifiers. By comparing the metrics, we identify Random Forest as the best classifier. Then we interpret the performance of the Random Forest classifier using the confusion matrix. Last, we examine important features in our best model.

4.1 Model metrics

In Figure 3, we show the accuracy and variance of different classifiers using two groups of features: the raw features and features after PCA decomposition. The baseline accuracy is $\frac{1}{\#group} \sim 33\%$ because the number of patterns in each group is the same $= 1657$, as ensured by uniform sampling.

We see that all the classifiers give a result higher than the baseline accuracy. PCA improves the performance of the classifier NB; for three classifiers, LVQ, SVM and LDA, using PCA or raw input does not make a significant difference on the performance; the performance of other classifiers is worse when using the PCA input. PCA has different influences on the performance of classifiers because there are different internal feature transformation mechanisms in each classifier. Overall, the random forest classifier gives the best results with the raw feature input and the parameter $\#variables = 32$.

The high accuracy and the fact that we can construct a classifier to differentiate the three groups of data imply that: first, algorithmically extracted patterns possess different properties than human-annotated patterns, which suggests an extra consideration to features of patterns when trying to discover patterns automatically; second, algorithmically extracted patterns have different structures than random passages, which means the extracted patterns cannot be replaced by sampled passages and could be more useful than sampled passages for various applications that employ musical patterns; last, human-annotated patterns contain more information than randomness despite subjectivity involved in the annotation process, which is in agreement with the carefully designed annotation acquiring process [43] and the previous findings that the annotations are useful for classifying tune families [2].

[3] https://goo.gl/ezuTCT

Original → Classified ↓	Alg	Ran	Anno
Alg	1595(±7.4)	17.2(±4.6)	24.8(±8.4)
Ran	8.3(±2.7)	1597(±2.8)	5.0(±2.2)
Anno	54.1(±9.9)	42.6(±2.7)	1627(±10.0)

Table 2. Confusion matrix results from the ternary classification experiment using the Random Forest classifier: mean and variance (in parenthesis) of ten experiments. The row names indicate the patterns are classified into the group of this name by the classifier; the column names indicate the patterns are orginally from the group of this name. Three groups of data are classified with high accuracies and significant p-values $\ll 0.05$.

4.2 Confusion Matrix

In Table 2, we give the confusion matrix results calculated from the classifier which has the best classification results: Random Forest. We perform the repeated cross-validation experiment ten times and take the average and variance of the resulting ten confusion matrices. The results show us on the individual patterns level how different groups of data are separable to one another. The sum of each column is roughly 1657, which is the group size of our data. The row sums do not have this constraint because we do not put restrictions on the group size as determined by the classifier. To read the table, for example, the number 24.8 in the right top corner of the table is the mean number of patterns classified as algorithmically extracted patterns but are actually annotations.

We see the classifier can differentiate the three groups with few misclassified instances. Although it would indicate a good performance of the algorithms if the count in the confusion matrix is larger in the algorithm pattern group and the annotation pattern group, we come to the conclusion that the algorithmically extracted patterns, annotated patterns and random passages all possess their own traits and are not similar enough for the classifier to fail. This is in accordance with previous research that the extracted patterns are not yet indistinguishable from the human annotations [2, 37]. On the positive side, we establish that neither annotated patterns nor extracted patterns are as meaningless as random data.

4.3 Feature Importance

In Figure 4, we show the individual importance value of the features in the classification process by using the Boruta algorithm [20]. The Boruta algorithm randomly duplicates

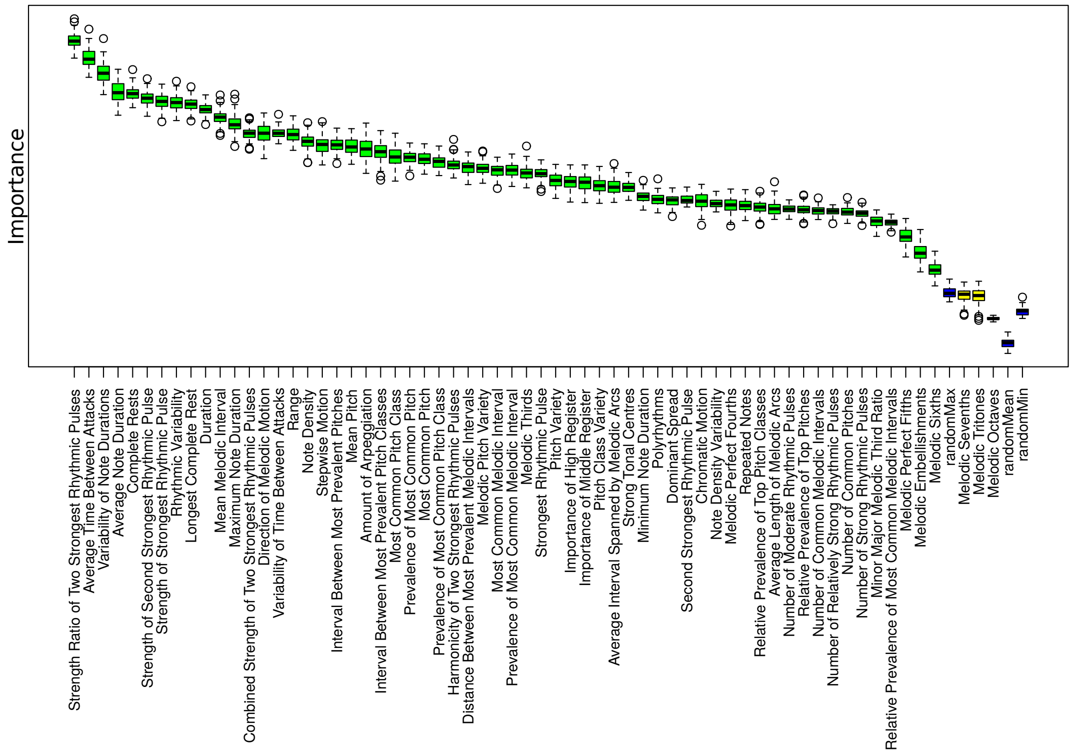

Figure 4. Feature importance in classifying annotated patterns, extracted patterns and randomly sampled passages using a random forest classifier. The boxplot shows the mean and variance (interquartile ranges) of the feature importance value [20]. The features are ranked by their importance. We omit the y-axis label because the absolute importance values are not relevant for our analysis. The colour green indicates features that are more important than the random features and are therefore confirmed to be important; blue entries show the performance of the random features; red and yellow indicate unimportant and tentative features respectively.

and shuffles the values in the original features. The algorithm then employs the random features together with the original features in classification tasks. During the classification process, the algorithm calculates and compares the Mean Decrease Impurity importance value [24].

Although we have 23 rhythmic features out of 63 features in total, all top ten most important features are rhythmic features. This suggests that these rhythmic features are relatively more important than other features in constructing the random forest classifier. The prominent features give hints on potential improvements to current existing pattern discovery algorithms. String-based and data mining algorithms translate pitch and duration pairs into a list of symbols and do not take into account metric structures imposed by musical punctuations such as bar lines and measures. Other known algorithms also seldom explicitly consider metric features in patterns. The feature importance values send the message that, in designing and evaluating pattern discovery algorithms, at least for the MTC-ANN dataset, we should take metric structures into considerations as well as the repetitions and pitch related features in the patterns.

In addition, the importance of other jSymbolic2 features is confirmed with the exception of three features which performed worse or at the same level as random features, as shown in Figure 4. For example, the Melodic Octaves feature is confirmed to be unimportant and the Melodic Sevenths and Melodic Tritones feature are marked to be a tentative attribute. They are unessential features because such intervals rarely happen in the MTC-ANN dataset.

5. CONCLUSIONS AND FUTURE WORK

We visualised and successfully classified human-annotated patterns, algorithmically extracted patterns and random passages in MTC-ANN. An analysis of the classification results suggests that the automatically extracted patterns are not yet indistinguishable from the human-annotated patterns, and both extracted and annotated patterns show different traits than randomly sampled passages. Using classification methods for comparative analysis of pattern groups provides a new perspective on examining the output of pattern discovery algorithms than the comparison of individual patterns in the MIREX task. In this way, we discover that rhythmic features play an important role in distinguishing the groups of patterns in MTC-ANN.

Future research needs to consider different contexts of patterns, such as within a melody, within a tune family and within the corpus, in order to investigate the influence of the context on what establishes a musical pattern. Expanding our research to other datasets once pattern annotations become available will allow us to verify whether the importance of rhythmic features is specific to MTC-ANN.

6. REFERENCES

[1] Kofi Agawu. *Music as discourse: Semiotic adventures in romantic music*. Oxford University Press, 2014.

[2] Peter Boot, Anja Volk, and W. Bas de Haas. Evaluating the role of repeated patterns in folk song classification and compression. *Journal of New Music Research*, 45(3):223–238, 2016.

[3] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[4] Chantal Buteau and Guerino Mazzola. Motivic analysis according to Rudolph Reti: Formalization by a topological model. *Journal of Mathematics and Music*, 2(3):117–134, 2008.

[5] Emilios Cambouropoulos. Musical parallelism and melodic segmentation. *Music Perception*, 23(3):249–268, 2006.

[6] Tsung-Ping Chen and Li Su. Discovery of repeated themes and sections with pattern clustering. *Music Information Retrieval Evaluation eXchange (MIREX 2017)*, 2017.

[7] Tom Collins, Andreas Arzt, Sebastian Flossmann, and Gerhard Widmer. SIARCT-CFP: Improving precision and the discovery of inexact musical patterns in point-set representations. *Proceedings of the 14th International Society for Music Information Retrieval Conference*, pages 549–554, 2013.

[8] Darrell Conklin. Discovery of distinctive patterns in music. *Intelligent Data Analysis*, 14(5):547–554, 2010.

[9] James R. Cowdery. A fresh look at the concept of tune family. *Ethnomusicology*, 28(3):495–504, 1984.

[10] Luciano da Fontoura Costa and Roberto Marcondes Cesar Jr. *Shape classification and analysis: theory and practice*. CRC Press, Inc., 2009.

[11] Music Information Retrieval Evaluation eXchange (MIREX) 2013. Discovery of Repeated Themes & Sections. `http://www.musicir.org/mirex/wiki/2013`.

[12] Vojtech Franc and Václav Hlavác. Multi-class support vector machine. In *Proceedings of the 16th International Conference on Pattern Recognition*, volume 2, pages 236–239. IEEE, 2002.

[13] Jerome H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[14] Jia-Lien Hsu, Arbee LP Chen, and C-C Liu. Efficient repeating pattern finding in music databases. In *Proceedings of the 7th International Conference on Information and Knowledge Management*, pages 281–288. ACM, 1998.

[15] Berit Janssen, W. Bas de Haas, Anja Volk, and Peter van Kranenburg. Finding repeated patterns in music: State of knowledge, challenges, perspectives. *Proceedings of the 10th International Symposium on Computer Music Modeling and Retrieval*, pages 277–297, 2013.

[16] Mari Riess Jones. Dynamic pattern structure in music: Recent theory and research. *Perception & psychophysics*, 41(6):621–634, 1987.

[17] Ian Knopke and Frauke Jürgensen. A system for identifying common melodic phrases in the masses of Palestrina. *Journal of New Music Research*, 38(2):171–181, 2009.

[18] Teuvo Kohonen. Improved versions of learning vector quantization. In *Proceedings of the 3rd International Joint Conference on Neural Networks*, pages 545–550. IEEE, 1990.

[19] Gerhard Kubik. Pattern perception and recognition in african music. *The performing arts: music and dance*, pages 221–49, 1979.

[20] Miron B. Kursa and Witold R. Rudnicki. Feature selection with the boruta package. *Journal of Statistical Software*, 36(11):1–13, 2010.

[21] Olivier Lartillot. Multi-dimensional motivic pattern extraction founded on adaptive redundancy filtering. *Journal of New Music Research*, 34(4):375–393, 2005.

[22] Olivier Lartillot. PatMinr: In-depth motivic analysis of symbolic monophonic sequences. *Music Information Retrieval Evaluation eXchange (MIREX 2014)*, 2014.

[23] Fred Lerdahl and Ray S. Jackendoff. *A generative theory of tonal music*. MIT press, 1985.

[24] Gilles Louppe, Louis Wehenkel, Antonio Sutera, and Pierre Geurts. Understanding variable importances in forests of randomized trees. In *Advances in neural information processing systems*, pages 431–439, 2013.

[25] Cory McKay. *Automatic Music Classification with jMIR*. PhD thesis, McGill University, 2010.

[26] David Meredith. COSIATEC and SIATECCompress: Pattern discovery by geometric compression. *Music Information Retrieval Evaluation Exchange (MIREX 2013)*, 2013.

[27] David Meredith. Music analysis and point-set compression. *Journal of New Music Research*, 44(3):245–270, 2015.

[28] David Meredith. Using SIATECCompress to discover repeated themes and sections in polyphonic music. *Music Information Retrieval Evaluation Exchange (MIREX 2016)*, 2016.

[29] David Meredith, Kjell Lemström, and Geraint A. Wiggins. Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music. *Journal of New Music Research*, 31(4):321–345, 2002.

[30] Daniel Müllensiefen. Fantastic: Feature analysis technology accessing statistics (in a corpus): Technical report. *Goldsmiths University of London*, 2009.

[31] Andrew Y Ng and Michael I Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Proceedings of the 15th Advances in Neural Information Processing Systems*, pages 841–848, 2002.

[32] Oriol Nieto and Morwaread M. Farbood. Identifying polyphonic patterns from audio recordings using music segmentation techniques. *Proceedings of the 15th International Society for Music Information Retrieval Conference*, pages 411–416, 2014.

[33] Sandeep Patel and A James Barkovich. Analysis and classification of cerebellar malformations. *American Journal of Neuroradiology*, 23(7):1074–1087, 2002.

[34] Matevž Pesek, Aleš Leonardis, and Matija Marolt. Symchman unsupervised approach for pattern discovery in symbolic music with a compositional hierarchical model. *Applied Sciences*, 7(11):1135, 2017.

[35] Matevz Pesek, Urša Medvešek, Aleš Leonardis, and Matija Marolt. Symchm: a compositional hierarchical model for pattern discovery in symbolic music representations. *11th Annual Music Information Retrieval eXchange (MIREX15). Malaga*, pages 1–3, 2015.

[36] Iris Yuping Ren. Closed patterns in folk music and other genres. *Proceedings of the 6th International Workshop on Folk Music Analysis*, pages 56–58, 2016.

[37] Iris Yuping Ren, Hendrik Vincent Koops, Anja Volk, and Wouter Swierstra. In search of the consensus among musical pattern discovery algorithms. *Proceedings of the 18th International Society for Music Information Retrieval*, pages 671–680, 2017.

[38] Brian D Ripley. *Pattern recognition and neural networks*. Cambridge university press, 2007.

[39] Grace Rubin-Rabson. Studies in the psychology of memorizing piano music: A comparison of the whole and the part approach. *Journal of Educational Psychology*, 31(6):460, 1940.

[40] Herbert A. Simon and Richard K. Sumner. Machine models of music. chapter Pattern in Music, pages 83–110. MIT Press, Cambridge, MA, USA, 1992.

[41] Wai Man Szeto and Man Hon Wong. A graph-theoretical approach for pattern matching in post-tonal music analysis. *Journal of New Music Research*, 35(4):307–321, 2006.

[42] Jan Van Balen. *Audio description and corpus analysis of popular music*. PhD thesis, Utrecht University, 2016.

[43] Peter van Kranenburg, Berit Janssen, and Anja Volk. The Meertens Tune Collections: The Annotated Corpus (MTC-ANN) versions 1.1 and 2.0.1. *Meertens Online Reports*, 2016(1), 2016.

[44] Gissel Velarde and David Meredith. A wavelet-based approach to the discovery of themes and sections in monophonic melodies. *Music Information Retrieval Evaluation Exchange (MIREX 2014)*, 2014.

[45] Anja Volk and Peter Van Kranenburg. Melodic similarity among folk songs: An annotation study on similarity-based categorization in music. *Musicæ Scientiæ*, 16(3):317–339, 2012.

[46] Cheng-i Wang, Jennifer Hsu, and Shlomo Dubnov. Music pattern discovery with variable markov oracle: A unified approach to symbolic and audio representations. In *Proceedings of the 16th International Society for Music Information Retrieval Conference*, pages 176–182, 2015.