ClaveNet: Generating Afro-Cuban Drum Patterns through Data Augmentation

Daniel Flores García
danialefloresg@gmail.com
Amherst College
Amherst, MA, USA

Hugo Flores García
hugofg@u.northwestern.edu
Northwestern University
Evanston, IL, USA

Matteo Riondato
mriondato@amherst.edu
Amherst College
Amherst, MA, USA

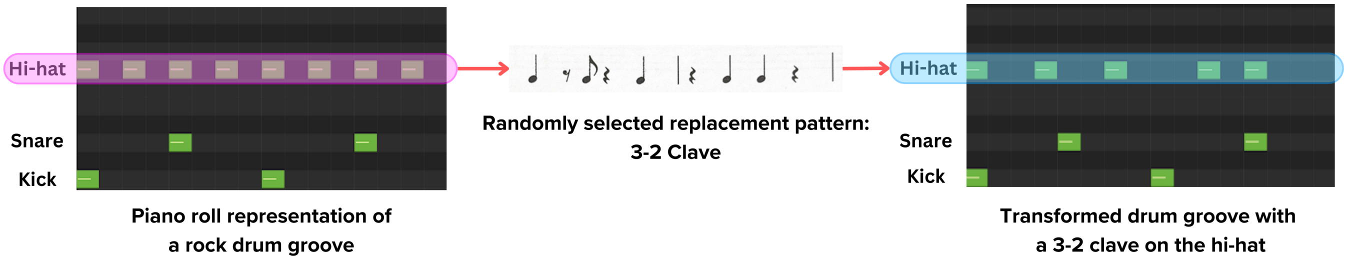
Hi-hat
Snare
Kick
Piano roll representation of a rock drum groove
Randomly selected replacement pattern: 3-2 Clave
Hi-hat
Snare
Kick
Transformed drum groove with a 3-2 clave on the hi-hat

Figure 1: Transformation procedure used in our data augmentation scheme to create new examples in a percussion dataset, which consists of swapping out randomly selected patterns from a MIDI drum file with Afro-Cuban rhythmic seed patterns. Our scheme creates at least one new example per original example in the percussion dataset, and can replace more than one pattern per transformation.

Abstract

We present ClaveNet: a generative MIDI model for Afro-Cuban percussion. We adapt the Monotonic Groove Transformer (MGT) —originally trained on the Groove MIDI Dataset (GMD)— to generate Afro-Cuban-influenced MIDI drum grooves. As Afro-Cuban drum MIDI data is scarce in the GMD and overall, we devise a data augmentation scheme to enrich MIDI percussion datasets with Afro-Cuban-inspired drum grooves by mixing examples with "seed patterns" rudimentary to Afro-Cuban percussion. To validate the effectiveness of our data augmentation algorithm at creating drum grooves infused with Afro-Cuban patterns, we trained MGT models on variants of the Groove MIDI Dataset augmented with our algorithm, and compared them to a baseline model trained on a non-augmented dataset. Our results show that MGT models trained with our augmented datasets are able to generate drum grooves whose rhythmic features are cumulatively closer to those from an evaluation set of real Afro-Cuban examples. We explore the effects of different hyperparameters to our system, discuss individual generated samples of selected models, and assess their faithfulness to Afro-Cuban styles. We hope this project fosters more research on developing music co-creation systems that encompass diverse musical styles outside those found in publicly available datasets.

CCS Concepts

• **Applied computing** → **Sound and music computing**; • **Computing methodologies** → *Neural networks*.

Keywords

Afro-Cuban music, Data augmentation, Generative models;

ACM Reference Format:
Daniel Flores García, Hugo Flores García, and Matteo Riondato. 2024. ClaveNet: Generating Afro-Cuban Drum Patterns through Data Augmentation. In *Audio Mostly 2024 - Explorations in Sonic Cultures (AM '24), September 18–20, 2024, Milan, Italy.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3678299.3678335

1 Introduction

Machine learning (ML) techniques have been a tool in the arsenal of researchers and artists building new instruments and interfaces for musical expression [8, 13, 14] for over a quarter century. In the field of music information retrieval, deep learning models are used for analytical tasks, like automatic music transcription [5, 10, 16]. More recently, deep generative models have become a common technique for generating both music audio [3, 9, 15] and symbolic music [22, 36, 41], with the goal of creating a generation of human-AI co-creative musical instruments [12]. There is a growing body of work focused on creating human-AI interfaces for symbolic music co-creation [17, 26, 27, 36]. These human-AI co-creation systems, like the one we propose, leverage symbolic music generation systems at their core. However, the effectiveness of a generative model is closely tied to the size of its training data [19]. The availability of training data for musical styles varies widely; thus, this availability dictates the feasibility of generating a music signal in a particular style.

AM '24, Milan, Italy

ACM ISBN 979-8-4007-0968-5/24/09
https://doi.org/10.1145/3678299.3678335

Some musical styles and instruments are well represented in large datasets. For instance, publicly-available datasets for jazz [2] and classical [23] piano are sufficiently large to train deep learning models. Consequently, research on music generation hovers around styles for which large datasets exist or can be easily compiled. Styles for which data is scarce are overlooked, even if their inherent musical characteristics are well-suited for deep learning architectures. With the widespread adoption of human-AI co-creation models in musicmaking interfaces, it is important that the multimodality of real musical styles is well-reflected in co-creative generative models, allowing artists across a wide variety of musical traditions to have meaningful interactions with these co-creative systems.

Afro-Cuban music is considered to be a "fundamental music of the New World" [39], as its musical concepts have permeated through classical music, ragtime, tango, jazz, rhythm and blues, country, rock and roll, funk, hiphop, and especially salsa. Its impact on American music has, however, gone largely unrecognized [39]. As such, despite the fact that the basic rhythmic structures intrinsic to the music can be conveyed faithfully with symbolic representations of music, and could be well captured through deep learning methods, there have not been any efforts to generate Afro-Cuban music with deep learning. Furthermore, we are not aware of the existence of any large Afro-Cuban percussion datasets.

This work presents ClaveNet, a generative music system that can convey the rhythmic idiosyncracies of Afro-Cuban music. To the best of our knowledge, this is the first attempt at training a deep learning-based generative model specific to Afro-Cuban music. Specifically, we achieve this goal by devising a data augmentation scheme that leverages domain knowledge of Afro-Cuban rhythms to infuse Afro-Cuban patterns into those of an existing percussion dataset; parameters to this scheme specify the size of the output dataset, the number of patterns infused per new example, and stylistic coherence among infused patterns. Our evaluation shows that, when trained on a dataset with our augmentation scheme, a symbolic percussion generation system is able to generate symbolic drum grooves that more closely resemble an evaluation set of human-performed MIDI Afro-Cuban drums than a baseline model.

2 Background

2.1 Symbolic Drum and Percussion Generation with Deep Learning

Deep learning architectures have been shown to be successful for music generation [6]. The GrooVAE Tap2Drum model by Gillick et al. [18] is a sequence to sequence Variational Autoencoder (VAE) that receives as input a single-voiced "tapped" rhythmic-sequence and outputs a multi-voiced "drum groove". Haki et al. [19] base their Monotonic Groove Transformer (MGT) model on the GrooVAE Tap2Drum model, using similar I/O representation but replacing the VAE architecture with a transformer encoder. Furthermore, they showcase an application of the MGT by incorporating it into a real-time accompaniment system. This system is limited to reinforcing the rhythm of a performance (in other words, it cannot provide contrapuntal rhythmic accompaniment). On the other hand, it is a pitch-agnostic system and thus is able to accompany any instrument that projects rhythmic information and can be reused within larger models for instrument-specific accompaniment generation [19]. We describe the MGT in more detail in section 2.4. McCormack et al. [30] present an alternative real-time drum accompaniment system which incorporates extramusical information via biometric data collection of the instrumentalist.

2.2 Afro-Cuban percussion

Afro-Cuban music is a fusion of a rhythms, melodies, harmonies and instruments stemming from African and European traditions; it is characterized by its complex rhythmic base comprised of syncopated patterns. The fundamental pattern is a two-measure phrase called the *clave*, which comes in two variations/orientations: the 3-2 clave and the 2-3 clave. Other patterns either rhythmically reinforce or syncopate against the clave; these patterns often also come in two variations that match the orientation of the clave; layering these patterns results in the "thick weave" that characterizes the sound of an Afro-Cuban percussion section [29].

2.3 Data Augmentation

Data augmentation techniques have been shown to be effective at several deep learning tasks [20, 35, 37, 40]. In the audio domain, augmentations like pitch-shifting, time-stretching, random filtering and cropping of a waveform have been shown to benefit various analysis tasks [1, 4, 31]. In symbolic music, transformations such as tempo and key changes [43, 45], MIDI excerpt "degradation" by applying note onset shifts, note addition and deletion, pitch-shifting [32], or augmenting the size of a dataset with generated examples [25] have been shown to improve both analytical (i.e., transcription) and generative models for human-AI co-creation interfaces.

2.4 Monotonic Groove Transformer

We chose the MGT [19] as the baseline drum generation model over other real-time drum accompaniment systems such as the one proposed by McCormack et. al [30] because of its reproducibility —as it is based on the transformer and trained on a publicly available dataset— and the fact that, being a Tap2Drum-like model, it generates drum grooves using only rhythmic information.

2.4.1 *Dataset.*

The MGT is trained on the two-bar variant of the Groove MIDI Dataset (GMD) [18], which consists of MIDI recordings of 10 drummers' performances. As Table 1 shows, the GMD contains Afro-Cuban MIDI data, but this style makes up a small proportion of the dataset.

Table 1: Distribution of musical styles across the Groove MIDI Dataset (GMD). Less than 4% of the examples in the GMD are Afro-Cuban.

Total	Rock	Latin	Jazz	Funk	Afrobeat	Afrocuban	Other
21312	31%	18%	11%	11%	5%	4%	20%

2.4.2 *I/O Representation: HVO Sequences.*

The inputs to the MGT are single-voiced "tapped sequences" (a.k.a. *monotonic grooves*), while the outputs are multi-voiced drum grooves, both represented with Hit-Velocities-Offsets (HVO) Sequences: a tensor-based symbolic representation of drum performances that uses a fixed rhythmic grid with a sixteenth note grid resolution. They are comprised of three matrices:

- **Hits** ($H \in \{0,1\}^{t \times v}$): Each entry denotes whether a note (also referred to as a hit) occurs at a particular time step for a given voice.
- **Velocities** ($V \in [0,1]^{t \times v}$): Denotes the normalized velocity of each hit.
- **Offsets** ($O \in [-1,1]^{t \times v}$): Denotes the deviation from the grid for each hit. Used to encode micro-timing information.

where t is the number of timesteps in the rhythmic grid and v is the number of voices (e.g., hi-hat, snare). Since each example is two-bars long, we let $t = 32$. Additionally, we choose a 9-voice set to represent drums, so $v = 9$. A pre-processing step converts each MIDI example in the GMD into an input/target pair of HVO Sequences.

2.4.3 Loss. The loss of the MGT is a sum of three terms: one for predicted hits, one for predicted velocities, and one for predicted offsets. Entries in the predicted velocities and offsets matrices whose locations correspond to values of zero in the target hits matrix are scaled by a penalty factor. For the hits term, binary cross entropy is used. For the velocities and offsets terms, mean squared error is used.

3 Methods

3.1 Drum Data Augmentation with Seed Patterns

We introduce a data augmentation scheme that aims to infuse Afro-Cuban sensibilities into a drum generator model trained on an augmented dataset. For each example in the GMD, the scheme creates new examples by transforming the original example. This transformation consists of randomly swapping out some of the example's voice patterns with a randomly chosen Afro-Cuban "seed pattern". To illustrate this process, suppose we would like to transform a two-bar rock back-beat drum pattern. Now, assume we randomly choose to replace the hi-hat voice. We would then replace the rock hi-hat pattern with a randomly chosen Afro-Cuban hi-hat seed pattern. Suppose this pattern is the clave. Then, our transformed MIDI file would sound like a rock back-beat with the clave on the hi-hat.

3.1.1 Seed Patterns. A seed pattern is a MIDI representation of a rhythmic pattern. The seed patterns used to augment the GMD are drawn from various Afro-Cuban musical traditions and styles as presented in the book *Afro-Cuban Rhythms for Drumset* [28] by Frank Malabe and Bob Weiner. The book's drumset exercises, grouped by Afro-Cuban styles (Son, Mozambique, Conga, Songo, Chachá, Merenge, Guaguancó), served as the seed pattern sources. By extension, each seed pattern is assigned a unique Afro-Cuban style. We selected a subset of these exercises and manually converted them to two-bar MIDI files using *Logic Pro*; when converting, we assigned unaccented MIDI notes a velocity value of 70 and accented notes a value of 100. The exercises in the book do not communicate micro-timing information, so as a workaround, we randomly offset each note by up to 10 ticks. These solutions naïvely include velocity and microtiming information in seed patterns and more work is needed to integrate this information into the augmentation scheme.

We use five voices to distinguish seed patterns in an exercise. In other words, we could theoretically extract up to five seed patterns from a single exercise. These voices are: Hi-hat, Snare, Kick, Toms, and Ride. Although HVO Sequences use separate Low Tom, Mid Tom, and High Tom voices, these were combined into a single voice to avoid splitting traditional patterns such as the "conga guaguancó figure". Additionally, the "Open Hi-hat" and "Crash" voices were omitted since they are not present in any exercise. The subset of exercises were selected such that all rhythmic patterns in the book appear in at least one of the exercises and no exercise's set of seed patterns is a subset of another exercise's set of seed patterns. In addition, we partitioned styles in two subgroups: styles based on the 2-3 clave, and styles based on the 3-2 clave. Table 2 shows the extracted seed pattern count for the 2-3 and 3-2 partitions.

Table 2: Seed pattern counts for Afro-Cuban styles, retrieved from *Afro-Cuban Rhythms for Drumset* [28].

	Songo	Mozambique	Son (2-3)	Conga	**Total**
Kick	8	9	6	6	23
Snare	8	10	9	6	33
Hi-hat	6	3	4	1	14
Tom	6	10	6	2	24
Ride	7	10	15	2	34
Total	35	42	40	9	**134**

(a) 2-3 styles

Son (3-2)	Chachá	Merengue	Guaguancó	**Total**
6	1	2	16	25
9	2	5	14	30
4	3	5	8	20
6	2	1	16	25
15	2	1	3	21
40	10	14	57	**121**

(b) 3-2 styles

3.1.2 Algorithm. To augment a MIDI-drum dataset with our extracted Afro-Cuban seed patterns, we create new examples by transforming each original example in the dataset. We denote the number of transformations —in other words, the number of new examples created per original example— with n_t. The size of an augmented data set is thus given by $|A| = |D|(n_t + 1)$, where A is the augmented dataset, and D is the original dataset.

Given an input example to transform, we first choose between the 2-3 and 3-2 seed pattern partitions uniformly at random. After having chosen between the 2-3 and 3-2 partitions, we can uniformly at random select a *primary* style s for the seed patterns to use to augment the input example. We also allow our algorithm to select any secondary style for each voice outside the primary style with a hyperparameter probability $p_{s'}$. Finally, we replace n_r voices from the original example with seed patterns, where n_r is an input parameter given by the user. Algorithm 1 shows the pseudocode for our data augmentation algorithm.

Algorithm 1: Transform MIDI Drum Example

```
Input: $drumExample$ - The MIDI drum example to transform
$styleSet$ - Set of Afro-Cuban styles
$patternSet$ - Set of candidate seed patterns
$n_r$ - Number of voices to replace
$p_{s'}$ - Probability of choosing a secondary style
Output: Transformed $drumExample$
$s \leftarrow$ style from the $styleSet$ selected uniformly at random
for $i \leftarrow 1\ to\ r$ do
    $useSecondaryStyle \leftarrow$
      True with probability $p_{s'}$, False otherwise
    if $useSecondaryStyles$ then
        $seedPattern \leftarrow$
          selected uniformly at random from $patternSet$ s.t.:
            $seedPattern.style \neq s$
            $seedPattern.voice \notin voicesToExclude$
    else
        $seedPattern \leftarrow$
          selected uniformly at random from $patternSet$ s.t.:
            $seedPattern.style = s$
            $seedPattern.voice \notin voicesToExclude$
    end
    $voicesToExclude.append(seedPattern.voice)$
    Replace $seedPattern.voice$ in $drumExample$ w/ $seedPattern.midi$
end
return $drumExample$
```

4 Experimental Evaluation

To evaluate our data augmentation scheme, we analyze the generated output of a set of models trained on different augmented datasets. We compare these models to a baseline model trained on the original GMD. [1]

4.1 Experiment Design

To create different data augmented datasets, we perform a grid search over the parameter space for our algorithm:

- for the number of transformations n_t, we choose the values $\{1, 2, 3\}$;
- for the number of voice replacements n_r, we choose the values $\{1, 2, 3, 4\}$;
- for the probability of choosing a secondary style $p_{s'}$, we choose the values $\{0.0, 0.5, 1.0\}$.

We train all models using one of the hyperparameter settings that Haki et al. [19] showed to result in a model with highest hit prediction accuracy and lowest test loss. This model is a transformer encoder a model dimension of 128, feed-forward dimension of 16, 4 attention heads, trained with a dropout of 0.16 and batch size of 16. Each model was trained for 50 epochs with a stochastic gradient descent optimizer. A penalty factor of 0.497 was used for velocity and offset losses.

4.2 Evaluation Set Comparison

As a method of performance evaluation for music generation models, we use a scheme proposed by Yang et al. [44]. Our objective is to compare a set of HVO Sequences generated by a data augmented model to an evaluation set of Afro-Cuban drum loops. Specifically, for each evaluation example (represented with an HVO Sequence) in the evaluation set, we reduce it to a monotonic groove which is used to prompt a model, yielding a generated example. Then, we assess the distance between the evaluation example and the generated example as follows. We first extract a set of rhythmic features for both examples. For each feature, we compute an intraset distance array out of the evaluation example's feature values and an interset distance array between the feature values of the generated example and the evaluation example. For both of these distance arrays, we estimate a probability density function (pdf) using a Kernel Density Estimator (KDE) with Scott's method for bin selection.[2] We then compute two distance metrics between the intraset pdf and the interset pdf; namely, KL Divergence and Overlapping Area.

We used two feature sets for evaluation, both defined in Table 3. The first —the complete-feature-set— mostly encodes note onset information, such as syncopation and voice density. This is the same set that Haki et al. used to evaluate their MGT models.[3] The second —the reduced-feature-set— is a subset of the complete-feature-set, where we exclude features that encode velocity and microtiming information, as this information is not essential to representing Afro-Cuban rhythmic structures; although future work would focus on investigating the importance of microtiming information on Afro-Cuban performance.

For our evaluation set, we used ToonTrack's Latin Midi Rhythms Pack [21], which is a set of MIDI drum grooves performed by the drummer Mauricio Herrera. Originally meant for the EZDrummer plugin, we were able to extract four-bar drum MIDI files labeled by style. We filtered out drum fills and non-Afro-Cuban MIDI files. Additionally, we split each MIDI file into two two-bar MIDI files, which were then converted to HVO Sequences. The resulting evaluation set contains 256 examples.

4.3 Ranking Models

To rank models we need to define distance from a model's generated set to the evaluation set. Let $G = g(M, E)$ denote a generated set for a model M that is prompted by the monotonic grooves from an evaluation set E. Let F be the the feature set used to evaluate G. We define the cumulative distance CD between a generated set G and an evaluation set E as:

$$CD(G, E) = \sum_{f \in F} ||\vec{d}(G, E, f) - \vec{t}||, \text{ where}$$

$$\vec{d}(G, E, f) = \begin{bmatrix} \text{KL-D}(G, E, f) \\ \text{OA}(G, E, f) \end{bmatrix}, \vec{i} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

where $\vec{d}(G, E, f)$ is the distance vector for a given feature, $\vec{i}$ is the "target" vector that we would like our distance vector to be close to, KL-D is the KL-divergence, and OA is the overlapping area.

We say that a model "performs better than another" if the cumulative distance from its generated set to the evaluation set is less than the other, i.e., if $CD(g(M, E), E) < CD(g(M', E), E)$. Using this logic, we compute the cumulative distance of all of our candidate models and compare these to the cumulative distance of the baseline model.

[1] Link to codebase: https://github.com/dafg05/ClaveNet-Parent

[2] In our implementation, we used `sklearn.neighbors.KernelDensity` [34]

[3] With the exception of the 'Timing Accuracy' feature, as this measure is undefined for HVO Sequences that do not contain hits at eighth note locations.

Table 3: Evaluation features. Bold features are included in the reduced-feature-set, all features are included in the complete-feature-set. These features as extracted for the generated set and the evaluation set, then they are used to compute the cumulative distance between the two. Adapted from *Real-Time Drum Accompaniment Using Transformer Architecture* [19].

Feature	Description
NoI	Number of instruments [11]
Total Step Density	$\frac{\text{Steps with at least one hit}}{\text{Total number of steps}}$ [11]
Average Voice Density (Low/Mid/Hi)	Step density of either low (Kick), mid (Snare, Toms), or high (Hi-Hat, Ride, Crash) voice groups over total step density [11]
Weak to Strong Ratio	$\frac{\text{number of onsets not on downbeats}}{\text{number of onsets on downbeats}}$ [7]
Polyphonic Sync	Polyphonic syncopation measure [7, 42]
Monophonic Sync (Low/Mid/Hi)	Monophonic syncopation of low, mid, or high voice groups [7]
Syness (Low/Mid/Hi)	$\frac{\text{Monophonic sync in voice group}}{\text{Number of onsets in voice groups}}$ [11]
Combined Sync	Sum of monophonic syncopation for all voices [7]
Complexity	Complexity measure based on mean of density and syncopation [7, 38]
Vel Similarity Score	Velocities of second bar minus velocites of first bar [19]
AC Skewness, Max, Centroid and Harmonacity	Autocorrelation curve attributes of velocity profiles [7, 24, 33]
Swingness	Measures swing weighted by number of swung notes [7]
Laidbackness	Measures laidbackness weighted by number of laidback onsets [7]

Table 4: Cumulative distances of selected models w.r.t. reduced-feature-set and complete-feature-set. The reduced-feature-set is comprised of rhythmic features encoding onset information, while the complete-feature-set also includes features encoding velocity and microtiming information.

Model ID	$CD \downarrow$	n_t	n_r	$p_{s'}$
upbeat-resonance	**0.455**	2	1	0.5
twilight-mountain	0.470	1	1	0.5
morning-cosmos	**0.499**	3	2	0.0
cosmic-plasma	0.569	2	3	0.5
expert-music	0.613	1	1	0.0
dashing-terrain	0.638	2	2	0.5
Baseline	0.652	0	0	0.0
jumping-aardvark	0.728	3	3	0.0
vivid-sky	0.883	2	4	1.0
rosy-violet	0.985	2	4	0.5

(a) Reduced-feature-set

Model ID	$CD \downarrow$	n_t	n_r	$p_{s'}$
twilight-mountain	0.969	1	1	0.5
morning-cosmos	**1.093**	3	2	0.0
Baseline	1.132	0	0	0.0
expert-music	1.173	1	1	0.0
efficient-snow	1.231	2	2	1.0
dashing-terrain	1.288	2	2	0.5
sage-jazz	1.361	1	2	0.5
pretty-dew	1.412	2	3	1.0
expert-bush	1.643	1	3	1.0
rosy-violet	1.896	2	4	0.5

(b) Complete-feature-set

4.4 Results

Thirty models (twenty-nine augmented and the baseline model) were analyzed w.r.t. the reduced-feature-set and the complete-feature-set. Table 4 shows the cumulative distance for the baseline model as well as the cumulative distances of selected data augmented models evaluated w.r.t. each feature set.

Six data-augmented models exhibited a smaller cumulative distance than the baseline model w.r.t to the complete-feature-set, likely due to the way that our data augmentation scheme includes velocity and microtiming information. This observation leads us to the preliminary conclusion that only a small subset of data-augmented models demonstrate more pronounced Afro-Cuban rhythmic characteristics. Our reduced-feature-set analysis yields more promising results. Nineteen out of twenty-nine data-augmented models exhibited a smaller cumulative distance than the baseline model. In addition, our best model's cumulative distance is $\approx 30\%$ smaller than that of the baseline. With these results, we can conclude that our proposed data augmentation scheme can produce models whose outputs have rhythmic features that are decidedly more Afro-Cuban than that of the baseline.

We make the following observations regarding the reduced-feature-set analysis:

- Ten out of the top twelve models have values of $n_t > 1$, a fact that reinforces the notion that bigger datasets lead to better performing-performing models.
- Eight out of the top twelve models have a value of $n_r = 1$. Additionally, the bottom 9 models (which all perform worse than the baseline) have values of $n_r = 3$ or $n_r = 4$. We conjecture that using fewer replacement patterns results in generated examples that strike a balance between representing Afro-Cuban patterns and preserving the "human-recording" quality from the GMD that is also present in the evaluation set.

Figure 2 shows a detailed breakdown of the reduced-feature-set analysis for models upbeat-resonance (the top model) and morning-cosmos (the top model with $n_r > 1$), each model compared to the baseline model. The points in these plots represent feature distance vectors $\vec{d}$. In accordance with our definition of cumulative distance, "better" points minimize kl-divergence and maximize overlapping area; in other words, they are closer to the 'target' vector $\vec{t} = (0, 1)$.

4.5 Discussion of generated samples

Although we were not able to conduct a formal subjective evaluation of our data augmented models, we present our own musical

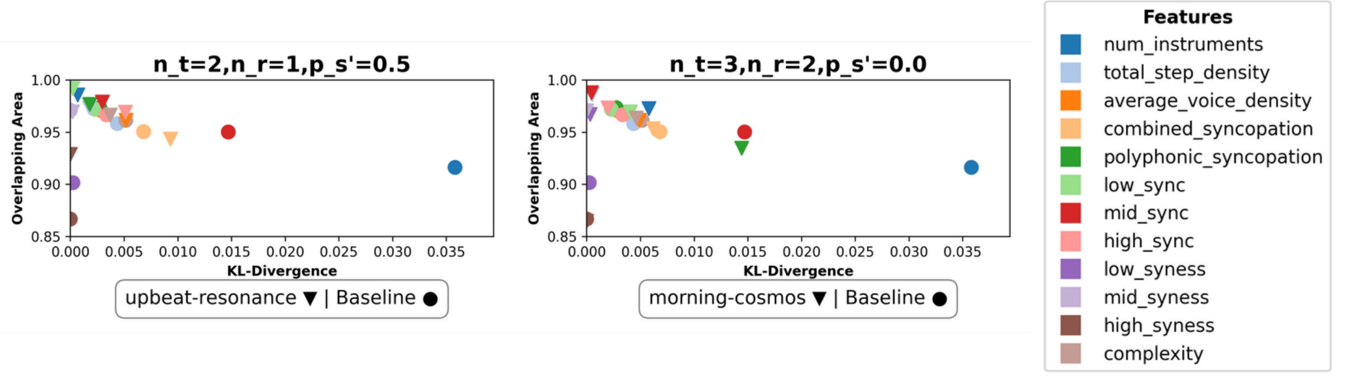

Figure 2: Feature distance vectors (KL-Divergence, Overlapping Area) of the generated set for models upbeat-resonance (left) and morning-cosmos (right) compared to feature distance vectors of the baseline's generated set. The closer a vector is to $(1, 0)$, the closer the generated set's corresponding feature values are to the evaluation set.

assessment of a subset of the generated output of model A (upbeat-resonance), model B (morning-cosmos), and the baseline model. To obtain generated samples from each model, we selected ten random examples from the evaluation set, reduced them to monotonic sequences, and used these to generate ten samples synthesized to `.wav` files[4]. We listened to these samples individually to help us evaluate each model. We encourage the reader to listen to these samples before reading this section.

Based on these samples, we consider Model B to be the most characteristically Afro-Cuban, followed by A, and then the baseline model. All models exhibit Afro-Cuban elements in at least some of their samples, which is surprising for the baseline model. Still, the augmented models are better at accurately representing Afro-Cuban rhythms. For example, many samples across all models exhibit a songo snare pattern. However, songo-snare-samples from the baseline model feature non-snare patterns more idiomatic to Brazilian samba (such as the four-on-the-floor kick pattern). Conversely, songo-snare-samples from Model A also include palito patterns idiomatic to guaguancó, while songo-snare-samples from Model B feature non-snare patterns (such as the kick and tom patterns) that reinforce the songo feel. In general, the baseline model gravitates more towards rock and samba, although it does introduce Afro-Cuban-like patterns occassionally.

Some samples generated by the augmented models exhibit unconventional drumset arrangement. Some Model A samples exhibit layered, dense Afro-Cuban rhythmic patterns that seem to lack polyrhythmic cohesion, which might be due to the absence of the clave. Other samples are essentially unplayable on a standard drumset by a single drummer, as they feature multiple simultaneous patterns voiced in a way that requires more than two hands to perform. We conjecture that unconventionally arranged samples are an artifact of the pattern replacement procedure from the data augmentation scheme. Regardless, such samples are both aesthetically and practically valuable, as there exist numerous widely-adopted techniques that enable the performance of these samples, such as MIDI sequencing and arranging to multiple percussionists.

5 Conclusions

To initiate discourse around deep-learning generation of Afro-Cuban percussion, we devised a data augmentation scheme that instills Afro-Cuban rhythmic patterns onto a percussion dataset. We found that models trained on augmented datasets generate drum grooves that exhibit more pronounced Afro-Cuban rhythmic ideas than those generated by a non-augmented baseline model. Thus, we believe that the development of these models are an important step towards faithful representation of Afro-Cuban music in generative music co-creation systems.

For future work, we'd like to conduct a formal subjective study of the augmented models' output; for an adequate study, it is crucial that it involves multiple musicians that have significant experience performing and/or studying Afro-Cuban music. Additionally, we would like to investigate enhancements to our data augmentation scheme that represent rhythmic patterns with meaningful velocity and micro-timing information. Finally, a more flexible rhythmic grid than that encoded into HVO Sequences is necessary to generate triplet-based polyrhythms and 6/8-based patterns idiomatic to many Afro-Cuban styles.

Acknowledgments

We are grateful to Camuel Hart for his comments on each of the generated samples; his observations regarding each model's faithfulness to Afro-Cuban percussion and to conventional drum arrangement techniques were key to constructing our discussion section. We thank Jason Robinson for suggesting *Afro-Cuban Rhythms for Drumset* as the source of seed patterns used in our data augmentation scheme. We are thankful to Will Rosenbaum for his feedback on the presentation of the evaluation results.

This work is based on Daniel Flores García's honors thesis at Amherst College. Part of this work is supported by the National Science Foundation grants IIS-2006765 (https://www.nsf.gov/awardsearch/showAward?AWD_ID=2006765), and CAREER-2238693 (https://www.nsf.gov/awardsearch/showAward?AWD_ID=2238693).

[4] Link to generated samples: https://dafg05.github.io/ClaveNet-Samples/

References

[1] Olusola O. Abayomi-Alli, Robertas Damaševičius, Atika Qazi, Mariam Adedoyin-Olowe, and Sanjay Misra. 2022. Data Augmentation and Deep Learning Methods in Sound Classification: A Systematic Review. *Electronics* 11, 22 (2022). doi:10.3390/electronics11223795

[2] Tosiron Adegbija. 2023. Jazznet: A Dataset of Fundamental Piano Patterns for Music Audio Machine Learning Research. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 1–5. doi:10.1109/ICASSP49357.2023.10096620

[3] Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, and Christian Frank. 2023. MusicLM: Generating Music From Text. arXiv:2301.11325 [cs.SD]

[4] Rafael L. Aguiar, Yandre M.G. Costa, and Carlos N. Silla. 2018. Exploring Data Augmentation to Improve Music Genre Classification with ConvNets. In *2018 International Joint Conference on Neural Networks (IJCNN).* 1–8. doi:10.1109/IJCNN.2018.8489166

[5] Rachel M. Bittner, Juan José Bosch, David Rubinstein, Gabriel Meseguer-Brocal, and Sebastian Ewert. 2022. A Lightweight Instrument-Agnostic Model for Polyphonic Note Transcription and Multipitch Estimation. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 781–785. doi:10.1109/ICASSP43922.2022.9746549

[6] Jean-Pierre Briot, Gaëtan Hadjeres, and François Pachet. 2017. Deep Learning Techniques for Music Generation - A Survey. *CoRR* abs/1709.01620 (2017). arXiv:1709.01620 http://arxiv.org/abs/1709.01620

[7] Fred Bruford, Olivier Lartillot, SKoT McDonald, and Mark Sandler. 2020. Multidimensional similarity modelling of complex drum loops using the GrooveToolbox. *ISMIR* (2020).

[8] Baptiste Caramiaux and Atau Tanaka. 2013. Machine Learning of Musical Gestures. In *Proceedings of the International Conference on New Interfaces for Musical Expression*. Graduate School of Culture Technology, KAIST, Daejeon, Republic of Korea, 513–518. doi:10.5281/zenodo.1178490

[9] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2023. Simple and Controllable Music Generation. In *Thirty-seventh Conference on Neural Information Processing Systems*.

[10] Frank Cwitkowitz, Kin Wai Cheuk, Woosung Choi, Marco A. Martínez-Ramírez, Keisuke Toyama, Wei-Hsiang Liao, and Yuki Mitsufuji. 2024. Timbre-Trap: A Low-Resource Framework for Instrument-Agnostic Music Transcription. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 1291–1295. doi:10.1109/ICASSP48485.2024.10446141

[11] Sergi Jordà Daniel Gómez-Marín and Perfecto Herrera. 2020. Drum rhythm spaces: From polyphonic similarity to generative maps. *Journal of New Music Research* 49, 5 (2020), 438–456. doi:10.1080/09298215.2020.1806887 arXiv:https://doi.org/10.1080/09298215.2020.1806887

[12] Philippe Esling and Ninon Devis. 2020. Creativity in the era of artificial intelligence. *CoRR* abs/2008.05959 (2020). arXiv:2008.05959 https://arxiv.org/abs/2008.05959

[13] Rebecca Fiebrink and Laetitia Sonami. 2020. Reflections on Eight Years of Instrument Creation with Machine Learning. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, Romain Michon and Franziska Schroeder (Eds.). Birmingham City University, Birmingham, UK, 237–242. doi:10.5281/zenodo.4813334

[14] Rebecca A. Fiebrink and Baptiste Caramiaux. 2018. 181The Machine Learning Algorithm as Creative Musical Tool. In *The Oxford Handbook of Algorithmic Music.* Oxford University Press. doi:10.1093/oxfordhb/9780190226992.013.23 arXiv:https://academic.oup.com/book/0/chapter/214418708/chapter-ag-pdf/44588862/book_28278_section_214418708.ag.pdf

[15] Hugo Flores Garcia, Prem Seetharaman, Rithesh Kumar, and Bryan Pardo. 2023. VampNet: Music Generation via Masked Acoustic Token Modeling. ismir:2307.04686 [cs.SD]

[16] Josh Gardner, Ian Simon, Ethan Manilow, Curtis Hawthorne, and Jesse Engel. 2021. MT3: Multi-task multitrack music transcription. *arXiv preprint arXiv:2111.03017* (2021).

[17] Jon Gillick and David Bamman. 2021. What to Play and How to Play it: Guiding Generative Music Models with Multiple Demonstrations. In *Proceedings of the International Conference on New Interfaces for Musical Expression.* Shanghai, China, Article 6. doi:10.21428/92fbeb44.06e2d5f4

[18] Jon Gillick, Adam Roberts, Jesse H. Engel, Douglas Eck, and David Bamman. 2019. Learning to Groove with Inverse Sequence Transformations. *CoRR* abs/1905.06118 (2019). arXiv:1905.06118 http://arxiv.org/abs/1905.06118

[19] Behzad Haki, Marina Nieto, Teresa Pelinski, and Sergi Jordà. 2022. Real-time Drum Accompaniment Using Transformer Architecture. In *Proceedings of the 3rd Conference on AI Music Creativity (AIMC 2022)* (2022-09-13/2022-09-15). AI Music Creativity, [s.l.], 10. doi:10.5281/zenodo.7088343 Online.

[20] Alon Halevy, Peter Norvig, and Fernando Pereira. 2009. The Unreasonable Effectiveness of Data. *IEEE Intelligent Systems* 24, 2 (2009), 8–12. doi:10.1109/MIS.2009.36

[21] Herrera, Mauricio. Accessed 2024. Latin Rhythms MIDI. https://www.toontrack.com/product/latin-rhythms-midi/. Accessed on April 10, 2024.

[22] Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Curtis Hawthorne, Andrew M. Dai, Matthew D. Hoffman, and Douglas Eck. 2018. An Improved Relative Self-Attention Mechanism for Transformer with Application to Music Generation. *CoRR* abs/1809.04281 (2018). arXiv:1809.04281 http://arxiv.org/abs/1809.04281

[23] Qiuqiang Kong, Bochen Li, Jitong Chen, and Yuxuan Wang. 2022. GiantMIDI-Piano: A large-scale MIDI dataset for classical piano music. arXiv:2010.07061 [cs.IR]

[24] Olivier Lartillot, Tuomas Eerola, Petri Toiviainen, and Jose Fornari. 2008. Multi-Feature Modeling of Pulse Clarity: Design, Validation and Optimization.. In *ISMIR*. 521–526.

[25] Alisa Liu, Alexander Fang, Gaëtan Hadjeres, Prem Seetharaman, and Bryan Pardo. 2020. Incorporating Music Knowledge in Continual Dataset Augmentation for Music Generation. *CoRR* abs/2006.13331 (2020). arXiv:2006.13331 https://arxiv.org/abs/2006.13331

[26] Ryan Louie, Andy Coenen, Cheng Zhi Huang, Michael Terry, and Carrie J. Cai. 2020. Novice-AI Music Co-Creation via AI-Steering Tools for Deep Generative Models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Honolulu</city>, <state>HI</state>, <country>USA</country>, </conf-loc>) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. doi:10.1145/3313831.3376739

[27] Ryan Louie, Jesse Engel, and Cheng-Zhi Anna Huang. 2022. Expressive Communication: Evaluating Developments in Generative Models and Steering Interfaces for Music Creation. In *27th International Conference on Intelligent User Interfaces* (Helsinki, Finland) *(IUI '22)*. Association for Computing Machinery, New York, NY, USA, 405–417. doi:10.1145/3490099.3511159

[28] Frank Malabe and Bob Weiner. 1983. *Afro-Cuban Rhythms for Drumset* (1st ed.). Manhattan Music.

[29] Rebeca Mauleón. 1993. *Salsa Guidebook: For Piano and Ensemble* (1st ed.). Sher Music Co.

[30] Jon McCormack, Toby Gifford, Patrick Hutchings, Maria Teresa Llano Rodriguez, Matthew Yee-King, and Mark d'Inverno. 2019. In a Silent Way: Communication Between AI and Improvising Musicians Beyond Sound. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–11. doi:10.1145/3290605.3300268

[31] Brian McFee, Eric J Humphrey, and Juan Pablo Bello. 2015. A software framework for musical data augmentation.. In *ISMIR*, Vol. 2015. Citeseer, 248–254.

[32] Andrew McLeod, James Owers, and Kazuyoshi Yoshii. 2020. The MIDI Degradation Toolkit: Symbolic Music Augmentation and Correction. *CoRR* abs/2010.00059 (2020). arXiv:2010.00059 https://arxiv.org/abs/2010.00059

[33] Maria Panteli, Bruno Rocha, Niels Bogaards, and Aline Honingh. 2017. A model for rhythm and timbre similarity in electronic dance music. *Musicae Scientiae* 21, 3 (2017), 338–361.

[34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[35] Luis Perez and Jason Wang. 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. *CoRR* abs/1712.04621 (2017). arXiv:1712.04621 http://arxiv.org/abs/1712.04621

[36] Victor Shepardson, Jack Armitage, and Thor Magnusson. 2022. Notochord: a Flexible Probabilistic Model for Embodied MIDI Performance. (2022). doi:10.5281/ZENODO.7088404

[37] Connor Shorten and Taghi M. Khoshgoftaar. 2019. A Survey on Image Data Augmentation for Deep Learning. *J Big Data* 6, 60 (2019). doi:10.1186/s40537-019-0197-0

[38] Georgios Sioros and Carlos Guedes. 2011. Complexity driven recombination of midi loops. *ISMIR* (2011).

[39] Ned Sublette. 2007. *Cuba and its music: From the first drums to the mambo.* Chicago Review Press.

[40] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. 2017. Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. *CoRR* abs/1707.02968 (2017). arXiv:1707.02968 http://arxiv.org/abs/1707.02968

[41] John Thickstun, David Hall, Chris Donahue, and Percy Liang. 2023. Anticipatory music transformer. *arXiv preprint arXiv:2306.08620* (2023).

[42] Maria AG Witek, Eric F Clarke, Mikkel Wallentin, Morten L Kringelbach, and Peter Vuust. 2014. Syncopation, body-movement and pleasure in groove music. *PloS one* 9, 4 (2014), e94446.

[43] Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang. 2017. MidiNet: A Convolutional Generative Adversarial Network for Symbolic-domain Music Generation. arXiv:1703.10847 [cs.SD]

[44] Li-Chia Yang and Alexander Lerch. 2020. On the Evaluation of Generative Models in Music. *Neural Computing and Applications* 32, 9 (2020), 4773–4784. doi:10.1007/s00521-018-3849-7

[45] Sangeon Yong, Changhyun Kim, Jiwon Kim, and S Telecom. 2019. Data augmentation and model optimization for piano transcription.