Musical Deep Learning: Stylistic Melodic Generation with Complexity Based Similarity

Benjamin D. Smith

Abstract

The wide-ranging impact of deep learning models implies significant application in music analysis, retrieval, and generation. Initial findings from musical application of a conditional restricted Boltzmann machine (CRBM) show promise towards informing creative computation. Taking advantage of the CRBM's ability to model temporal dependencies full reconstructions of pieces are achievable given a few starting seed notes. The generation of new material using figuration from the training corpus requires restrictions on the size and memory space of the CRBM, forcing associative rather than perfect recall. Musical analysis and information complexity measures show the musical encoding to be the primary determinant of the nature of the generated results.

Introduction

Deep learning models have recently, and dramatically, improved the state-of-the-art in a wide array of computational domains, making major advances in solving problems that have frustrated artificial intelligence researchers for years (LeCun, Bengio, and Hinton 2015). Of particular benefit and interest is the data driven, representation learning ability of these models, better mimicking human learning, and eliminating the need for hand engineered feature formulation. This capability is enabling machines to process raw data directly, automatically deriving features in order to discover high-dimensional structures, without relying on human conceptualizations (which can be both highly time consuming to formulate and limited in scope)(Humphrey, Bello, and LeCun 2012). This pure learning approach is highly promising in informing generative, creative musical models that can bypass the intrinsic limits of historical theories and access musical information in an unmediated fashion.

The ways in which music is formalized and described are primary determining factors in how musical artifacts are understood and evaluated. Generative computational systems and models rely entirely on the conceptual musical framework with which they are presented. Formulations of 'scale,' 'key,' and 'diatonic' underlie and frame most of contemporary popular music and historical western art music. Yet, most of western music theory was created post-facto to explain compositional practices already in common use (such as the 'rules' of voice leading and counter-point). As such these formulations are primarily descriptive, and their use as proscriptive rules for generative systems will limit the output to a subset of our human musical conceptualization (Lerdahl and Jackendoff 1983).

Creative systems, which seek to model or mimic some aspect of human creative behavior, often require an extensive conceptual model programmed by hand to provide an adequate context for analytic or generative processes. Meaningful output (i.e. music) is often predicated on access to the human-derived frameworks, such as classical music theory. However these systems will be limited by the accuracy and depth of the framework provided. Given that significant refinements and improvements are periodically made to modern music theory, based on both musical and psychological research, it follows that the current state-of-the-art is incomplete and our understanding fallible. MIDI is biased towards specific musical expression (such as 12 tone equal tempered scales and keyboard input), yet is commonly employed as the starting point for artificial intelligence work on representational music data.

The conditional restricted Boltzmann machine (CRBM) (Taylor and Hinton 2009) employs a temporally dependent network model and brings the promise of data-driven learning with the ability to encode time reliant data. Exploring a range of potential feature sets with different configurations of the CRBM results in a variety of generated musical output from exact replications (of the training corpus) to highly dissimilar content. Comparing the information complexity of the generated material confirms musical analysis observations showing the dependency of the results on the feature encoding.

Deep Learning and Music

Investment in deep learning in music and music information is on the rise and is reaching many application areas (Humphrey, Bello, and LeCun 2012). Music audio classification is making use of deep belief networks and deep convolutional nets (Hamel and Eck 2010; Lee et al. 2009; Van den Oord, Dieleman, and Schrauwen 2013), leveraging

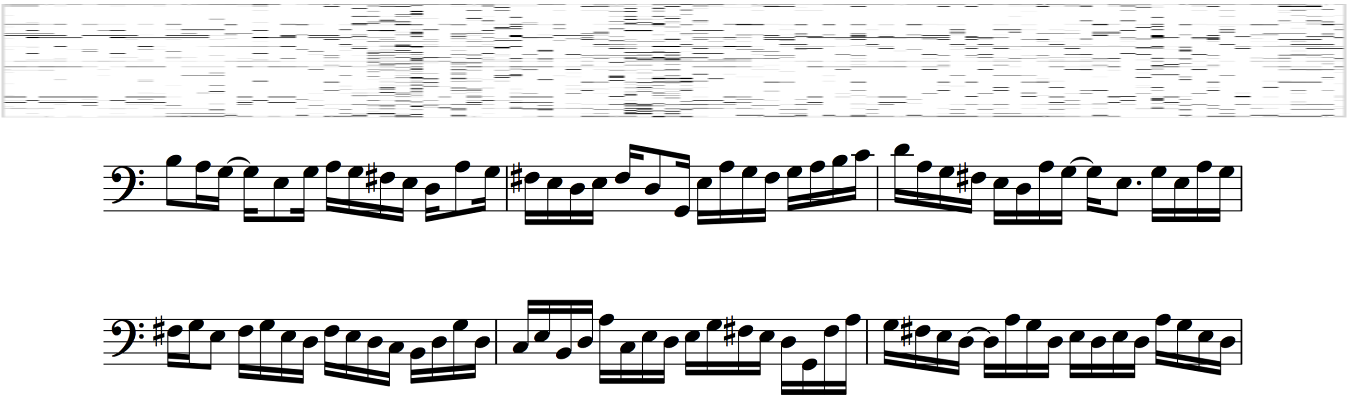


Figure 1: Probabilities of each feature being "on" in the visible units of a trained CRBM. Shown is a 200 unit model (plotted over time), trained on J.S. Bach's *Suite for Solo Violoncello, BWV 1007*, and the resulting music generation.

work in speech processing for music feature identification and classification. Dynamic Bayesian networks in layered models have been applied to creative, generative systems, improving characterization of long-term (form level) dependencies (Smith and Garnett 2012; Smith 2013).

Several formulations of RBMs have been used for polyphonic music analysis and generation, as a component in polyphonic transcription (Boulanger-Lewandowski, Bengio, and Vincent 2012; Vohra, Goel, and Sahoo 2015). While these models have proven statistically successful at predicting chord sequences from a training corpus the generative output is limited. Musical structure, longer-term dependencies, and musical quality is not evidenced. In these cases the source MIDI files are converted into 88-bit feature vectors, where each bit represents a key on a typical piano keyboard (so called 'one-hot' encoding). This "piano roll" representation works well with classic, binary RBMs. However the ability for the model to abstract 'transposition,' melodic, or motivic relationships will depend on the corpus (i.e. octave equivalency is based on patterns appearing 12 steps apart). Starting from a symbolic representation of music (i.e. MIDI notes) brings many assumptions about how humans hear music, yet the encoding will influence the deep learning model in a fundamental way.

Conditional RBMs have proven successful at modeling human motion and styles of gesture (Taylor and Hinton 2009), learning and characterizing continuous temporal dependencies, and this indicates the potential to model musical structures. After training on a corpus of musical pieces, with a full feature set, the CRBM is able to reconstruct a piece given only a few starting notes (i.e. a query by melody task). This is excessively accurate and presents the problem of how to extend this deep learning model into a fuzzier, association retaining and generating model. Effectively, the specificity of the model must be reduced to encourage the abstraction of broader features, while still retaining characteristic surface level figurations (motifs and melodic patterns). The RBM model presents many options, including restricting its memory space (the number of units) and restricting its temporal linkages (number of time steps or 'order'), which can cause it to make generalizations rather than exact recall (see fig. 1).

CRBM

The generic RBM models static frames of data, with a layer of binary visible units v and a layer of hidden units h (Smolensky 1986). Undirected connections between layers, and the absence of connections within a layer, comprise this "energy-based" model, with the energy function $E(v,h)$ defined as:

$$E(v,h) = -b'_v v - b'_h h - h'Wv \tag{1}$$

where the visible and hidden units are connected with weights W and b_v, b_h denote layer biases. Observing the visible units makes the hidden units conditionally independent, resulting in easy inference (for full details on the RBM model see (Taylor, Hinton, and Roweis 2007; Smolensky 1986)])

Modeling temporal dependencies is accomplished by treating each time slice as fixed inputs for the next (Taylor, Hinton, and Roweis 2007). These additional directed connections (fig. 2), from the past visible units to the current visible and hidden configuration, define the Conditional RBM. The number of past time steps linked is referred to as the "order" of the model (an order of 3 is used for most of this study). Alternating Gibbs sampling is used to generate new material starting from several frames of initialization data to prime the model.

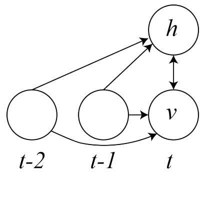


Figure 2: Conditional RBM architecture (three previous time steps are shown, $order = 3$).

Melodic Complexity

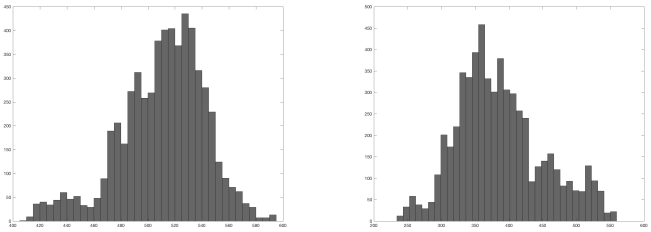
The possibilities of the CRBM are explored through a generative imitation task, using the CRBM to produce melodies that exhibit similarities to a training corpus (Pasquier et al. 2016). The production of works that sound similar to, but do not directly copy the originals, is a positive result (while output that is entirely dissimilar or an exact duplicate is a failure). Truly successful models are expected to incorporate both surface figurations (motifs, melodic patterns) as well as harmonic and structural elements. Similarity is measured by complexity distance and through traditional musical analysis. The six solo cello suites by J.S. Bach (BWV 1007-1012) are used as the training set, selected for their extensive analysis literature, relatively monophonic nature, and rhythmic simplicity. This training set comprises 36 pieces and 30222 notes.

Table 1: Complexity measures of 1000 note sequences from J.S. Bach and CRBM output.

	Entropy		LZW		Zip	
Features	M	s	M	s	M	s
All 1s	0	0	45	0	49	0
Bach	**2.41E+07**	**2.45E+06**	**509.533**	**31.817**	**387.026**	**64.413**
Pitch	2.01E+07	6.2E+05	465.01	8.92	499.207	8.281
Pitch+P.C.	2.26E+07	1.397E+06	471.314	11.588	494.053	12.564
Pitch+Int.	2.29E+07	1.89E+05	473.519	16.06	494.38	15.73
Pitch+P.C.+Int.	5.12E+06	1.94E+06	215.142	55.131	234.418	63.063
Pitch+Deriv.	2.36E+07	2.25E+05	381.696	15.217	406.899	17.974
Intervals	1.21E+07	1.14E+07	594.297	30.437	636.558	40.785
Int+Deriv	1.408+E07	8.12E+06	643.323	31.829	718.298	43.122
Int+P.C.	8.63E+07	1.73E+07	624.338	30.307	679.545	36.933
Raw Pitch	1.95E+07	7.25E+04	247.71	7.615	285.441	9.205
6 Pitch Deriv.	2.04E+07	3.99E+05	374.933	46.142	**376.38**	39.407
12 Pitch Deriv.	2.34E+07	5.18E+05	573.606	7.225	595.661	6.115
24 Pitch Deriv.	2.11E+07	2.42E+05	443.264	11.346	486.374	10.457
FFT 4	1.9E+07	7.45E+04	356.228	5.44	404.247	6.948
FFT 8	**2.39E+07**	1.18E+05	**491.359**	5.886	524.417	5.519
FFT 16	2.33E+07	3.23E+05	542.213	8.384	554.706	6.219
FFT 32	2.11E+07	4.64E+06	628.9	116.134	612.722	116.467
FFT 128	5.34E+12	1.05E+13	565.123	36.25	574.569	43.026
Random Numbers	4.88E+07	1.48E+06	972.11	4.44	934.218	1.224

The complexity of the training corpus is characterized through measurements of Shannon's Entropy, Lempel-Ziv (LZW) compression (i.e. run-length encoding), and Zip compression. These have been applied to music as measurements of information density, and shown to capture comparable characteristics of music (Li and Sleep 2005; Shmulevich and Povel 2000). The LZW and Zip compression of 1000 note sequences drawn from Bach show relatively normal distributions (fig. 3), and have a fair degree of correlation (Zip data is not shown further, unless it deviates from this correlation in specific cases). The Shannon's entropy measure is non-linear and appears to consist of distinct sub-populations, but with a relatively narrow distribution (fig. 4). The correlation between LZW and Shannon is highly non-linear and presents a complex relationship (fig. 5), portraying centers of complexity (corresponding with different pieces from the corpus, connected by samples that span pieces).

The complexity measures of 1000 note generated sequences tend towards normal distributions with tighter deviations (compared to the training set, see table 1 and figs. 5 & 6). A variety of different features and musical encodings are explored, below.



(a) LZW ($M = 509.53, \sigma = 31.82$)

(b) Zip ($M = 387.03, \sigma = 64.42$)

Figure 3: Compression of 1000 note segments from J.S. Bach's solo cello suites. (correlation $r = 0.73, p < 0.001$).

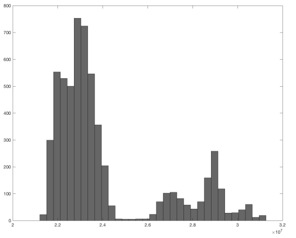


Figure 4: Shannon's entropy of 1000 note segments of Bach's solo cello suites.

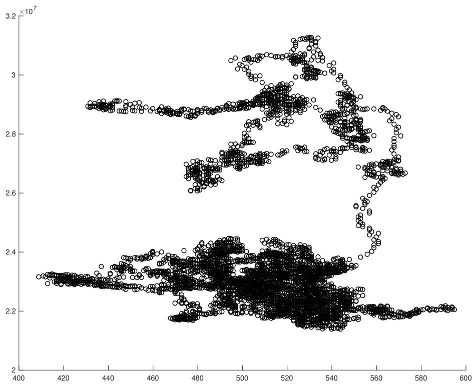
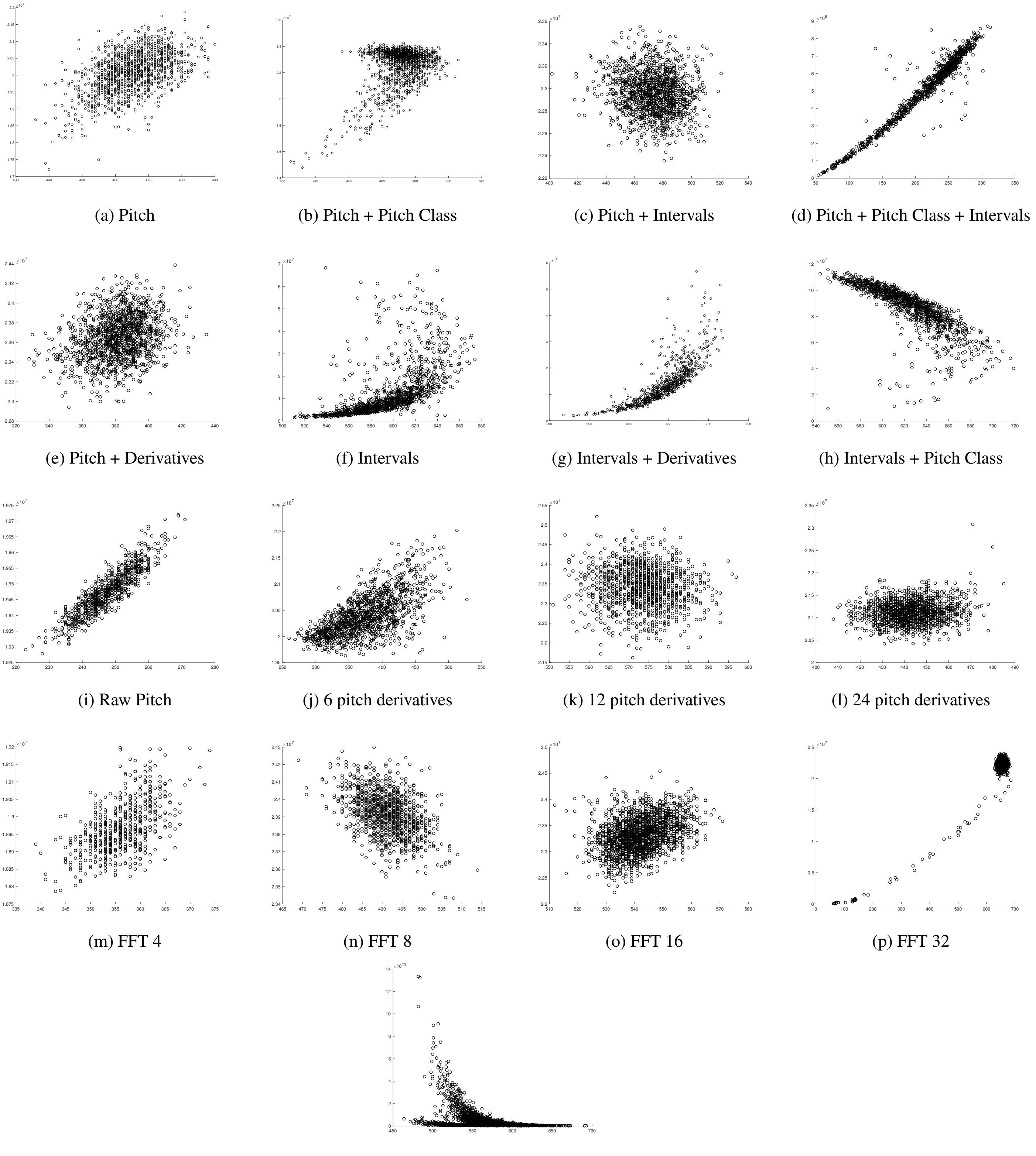


Figure 5: Shannon's entropy vs. LZW compression of 1000 note segments of Bach's cello suites.



(a) Pitch

(b) Pitch + Pitch Class

(c) Pitch + Intervals

(d) Pitch + Pitch Class + Intervals

(e) Pitch + Derivatives

(f) Intervals

(g) Intervals + Derivatives

(h) Intervals + Pitch Class

(i) Raw Pitch

(j) 6 pitch derivatives

(k) 12 pitch derivatives

(l) 24 pitch derivatives

(m) FFT 4

(n) FFT 8

(o) FFT 16

(p) FFT 32

(q) FFT 128

Figure 6: Entropy vs. compression of 1000 note sequences of generated material with different feature sets. All CRBMs with $order = 3, N_v = N_h = 100$.

Encodings

All of the included musical examples were randomly selected from generated material.

Pitch

Since the CRBM boasts minimal data pre-processing requirements (other than normalization) the naïve approach is to feed melodic pitch data to the network for training (i.e. 1 feature representing the MIDI pitch value is used).

The results from the naïve approach are lack luster, succeeding in producing music that is not entirely random (it wanders around in small steps, up to major 3rds), but is highly chromatic and non stylistic to the original (fig. 8). However it does exhibit direction with a regular rise and fall every 2 beats in the first 2 bars, then extending this to a whole bar in the second system. While the intermediary notes appear incidental, distinct major and minor chords and scales are evident (commencing on G, A-flat on beats 3 & 4, E-flat minor and major implied in bars 2 and 3, and a strong D-flat major in bar 4). The complexity measures indicate that it is more compressible, due to the rather narrow wandering behavior.

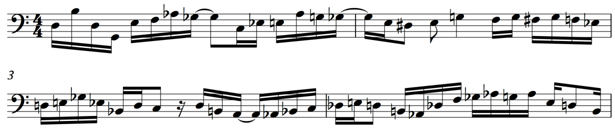


Figure 7: CRBM generated music, using raw pitch data ($order = 3, N_v = N_h = 30$).

Treating the pitch p as a position (say, along a fixed length string) and adding features F of the derivatives of that position (velocity, acceleration, etc. pre-calculated from the training data) greatly increases the similarity to the corpus:

$$F = [p, \frac{dp}{dt}, \frac{d^2p}{dt^2}, \frac{d^3p}{dt^3}, ... \frac{d^np}{dt^n}] \tag{2}$$

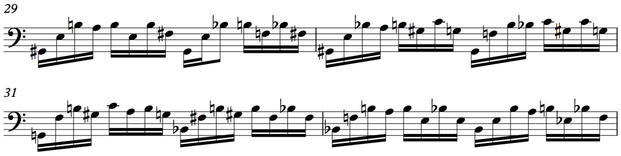


Figure 8: Generated from pitch data with derivatives, $n = 12$ ($order = 3, N_v = N_h = 100$)

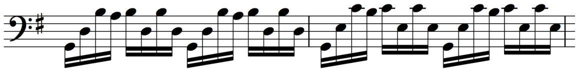


Figure 9: Opening of BWV 1007, Prelude for Solo Violoncello by J.S. Bach.

While the iconic opening of the G-major Prelude (BWV 1007, fig. 10) can be recognized underneath the chromaticism (fig. 9, pitches have been rounded to the closest half step for notation), it appears that the RBM has retained a fuzzier memory and representation of the original. In a promising fashion, the generated material alternates between several different types of melodic patterning (not shown here), returning to the opening arpeggiated pattern every 16 bars or so. This indicates it has been able to entrain melodic features and short-term dependencies (melodic/motion behaviors), and reproduces them in a continuous, coherent fashion. The complexity measures for this encoding are higher than pitch alone, however now they indicate too much unique material without enough structural repetition.

It apparently treats small intervallic changes (in fig. 9) lightly, yet this sets up harmonic expectations that are not trivial variations. The harmonic sequence in figure 9, starting with an E major, implying a flat-5, moving through a G7 to various altered B-flat connotations, is not true to Bach. The CRBM has been unable to derive appropriate harmonic implications or relationships from the linear MIDI pitch data.

Piano Roll

Here features F encode the position of each key on a piano (continuous from depressed 1 to up 0) at each time frame t (every 250ms, or sixteenth-note at 120 b.p.m.). This essentially captures the same information as a piano roll such as is used in mechanical Player Pianos. However, keys do not spring back instantaneously so interpolation after each key press i over several frames is used (employing a short-term memory spatial encoding of pitch where the length of the memory is determined by decay factor d, fig. 11) (Smith and Garnett 2012):

$$F_t = dF_{t-1} \tag{3}$$

$$F_t i = 1 \tag{4}$$

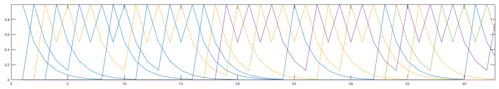


Figure 10: Encoding of 'key presses' over time.

The harmonic improvement is immediately apparent (fig. 12), commencing with a strong A7 scale and arpeggio, resolving to D major in the second bar, eventually hinting at a D7 in bar 4. This is due to the effective inversion of the data representation, disallowing the CRBM from sliding between keys, or playing any that are absent in the training data (i.e. out of the cello's range).

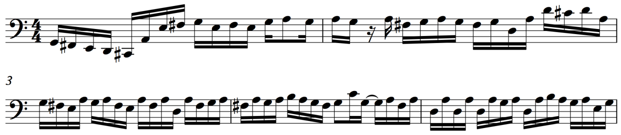


Figure 11: Music from CRBM trained on 88-key 'piano-roll' ($order = 3, N_v = 100, N_h = 100$).

It is apparent from examination of extensive results at this point that the model has no inclination of octave equivalency, but this can be provided by adding features for pitch class (i.e. $pc = p \ mod \ 12$). The immediate impact of this addition appears to be subtle, and insufficient evidence is currently available to document the effects, however it appears to produce no deleterious change and the results are similar to the 88-key model (see fig. 13). We can see more rhythmic patterning and the repetition of fragments to form longer phrases (up to a bar and a half long).

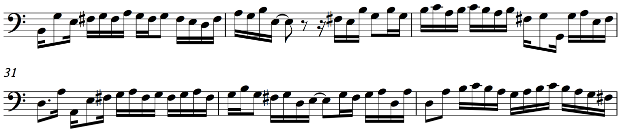


Figure 12: Music from 88-key 'piano-roll' with 12 pitch class features ($order = 3, N_v = N_h = 100$).

Intervals

Functional harmonic music, such as the works of Bach, is heavily reliant on the specificity of intervals (distance between successive notes). While the frequency distances between notes are linearly related, in harmonic dimensions the distances are much more complex (G is closely related to D, frequency ratio of $2 : 3$, but G:G# is harmonically distant, with a frequency ratio of $1 : \sqrt[12]{2}$). Hence it is common to frame intervals in terms of distinct 'classes,' rather than continuous distances. Considering each interval (minor 2nd, major 2nd, minor 3rd, etc., signed) a distinct token and encoding these as with pitches before provides results like figure 14.

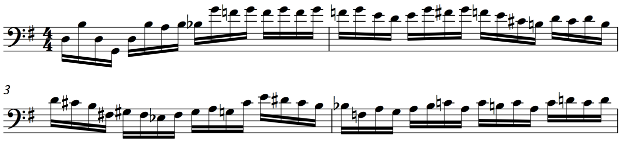


Figure 13: Music from CRBM trained on interval tokens ($order = 3, N_v = N_h = 100$).

The chromatic wandering seen here again reveal that intervals alone are not sufficient to guide the generation, and can cause the generation to move out of range of audible pitch (not shown). Combining intervals with the pitch derivatives and pitch classes further increases the complexity of the output (see table 1).

Fourier Transform

The closest feature set, in entropy measure (above), results from computing the Discrete Fourier Transform (DFT) of windowed note sequences. This is calculated for every sequence of length n (the conventional 'window' size, with a hop of 1 note) in the training set (DFT at time $t = m$ is computed over notes $[t_{m-n}, t_m]$) and the real and imaginary outputs are concatenated to comprise the feature vector. The inverse DFT is performed on the generated output to create the musical results. However, there appears to be a threshold above which the impact of the DFT encoding is diminished. The best results have been achieved with a length of $n = 8$.

However the musical example (fig. 15) does not compare favorably with the training corpus. Excessive chromatic movement is seen, and the complexity match appears to be achieved through more unique pitches, rather than harmonic movement or musical structure.

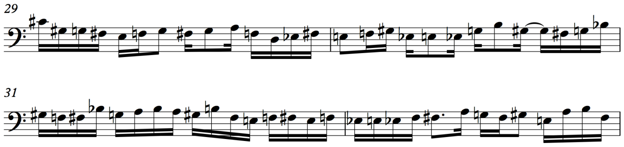


Figure 14: Music from CRBM trained on FFT encoding (window size=8,$order = 3, N_v = N_h = 100$).

CRBM Order

The order of the CRBM determines the depth of temporal dependencies the model is able to encode. An order of 2 only looks at two note pairs (in the visible layer) while higher orders look at more previous notes for each time frame. An order of 5 and greater (with sufficient hidden units) causes the CRBM, after being fed the initialization frames (notes), to regenerate the whole matching piece from the training corpus. The initialization notes can be selected from anywhere in the corpus and the CRBM will correctly proceed from there. This is impressive behavior, however it goes beyond stylistic mimicry to straight copying (a promising result for query and retrieval, but less valuable for generative tasks with variation).

We have seen many examples of lower order generation thus far. Increasing the order, as anticipated, increases the local temporal dependencies, i.e. motifs become longer (fig. 16). In some cases this causes the CRBM to repeat the same 8-16 note pattern in an endless loop (the beginnings of this behavior are seen in fig. 16).

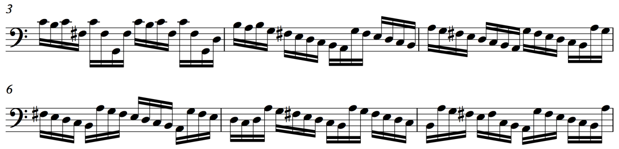
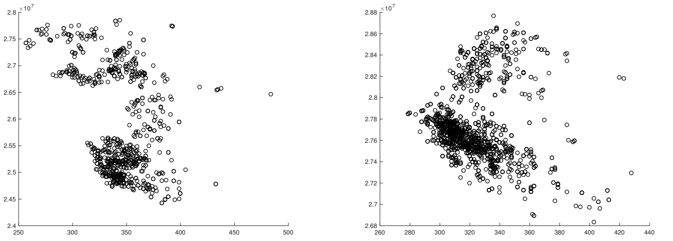


Figure 15: Higher order CRBM generation ($order = 7, N_v = 100, N_h = 100$).

Yet this is musically more reminiscent of the original corpus and the musical processes that are theoretically understand to be at work within Bach's music (i.e. repetition and sequence of short melodic fragments, descending scales in this instance). The complexity measures show that at this point the CRBM is capable of producing the non-normal variety of outputs seen in the training set (fig. 17), but with higher entropy and compressibility.



(a) $order = 7, N_v = N_h = 500$ (b) $order = 9, N_v = N_h = 300$)

Figure 16: Compression of 1000 note segments from CRBM output trained on all proposed features.

Finally (fig. 18, and fig. 1), we combine all the musical features into one encoding and allow the CRBM to associate amongst all of them (the complete feature vector is 88 'keys,' 12 pitch classes, 40 signed intervals, 12 pitch and pitch derivatives, and 16 FFT values). The order and number of units per layer is restricted in order to force reductive associative learning in the model ($N_v, N_h > 300$ and $order \geq 5$ tends towards perfect recall).

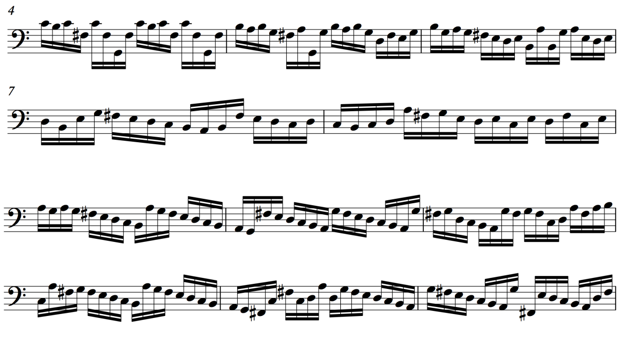


Figure 17: Two sections generated with all proposed features ($order = 3, N_v = 400, N_h = 20$).

The result, compared with figure 16, is more varied and appears to show more direction. The upper half contains appropriate harmonic resolution from the D7 to G in the second bar and a close quote of the Bach Prelude opening (fig. 10). From there it proceeds into stylistically appropriate scalar and broken-chord material, implying movement to the sub-mediant and dominant. In the lower half we see descending patterning analogous to figure 16 but with more variation in length (descending to F#2) and repetition of sub-motifs (the G-F#-G movement in the 3rd bar).

Once a sufficient musical encoding is in place the primary challenge for the CRBM is preventing overly accurate recall. Reducing the categorization ability of the network layers and limiting the temporal dependency length appears to accomplish this effectively.

Discussion

In the process of selecting feature sets and CRBM tuning parameters several additional observations were made:

- Any feature set that is based primarily on intervallic distances causes the CRBM to chromatically drift, producing a-stylistic melodic movement. A level of approximation appears in the CRBM that causes the intervals to shrink or expand just enough to result in steps and leaps that are not seen in the training corpus. This further points out the non-linear nature of musical harmonic space, misrepresented by linear encodings of musical movement.
- Frequently, regardless of encoding, the CRBM finds a cycle within the musical sequence that causes it to loop continuously in the generated output. It may start with more variety but if a loop has been entrained within the 1000 note output it appears to converge on the repetitive cycle (which results in music with far less entropy and lower compression values).
- Too much training data for the number of units in the network cause the training process to diverge, rather than training properly. A thinning method for the training data (similar to downsampling pixel data prior to RBM training) would help alleviate these situations.

Future Work

The CRBM shows promise as a memory/recall component in a creative generative system. It has the ability to abstract musical patterns and temporal dependencies and reproduce them in new combinations. However the two layer model does not appear to be deep enough to move beyond simple stylistic imitation. Yet the CRBM can easily be extended into multilayer models, allowing the deeper layers to further abstract longer-term temporal structures and dependencies. This model may also profit from incorporation in a multi-agent system wherein other learning agents can reinforce or critique the CRBM's output and guide it towards novel generation.

Given the multitude of deep learning applications currently being researched it is possible that other representations of music may be usefully analyzed and reproduced. For example, image and computer vision processing deep learning models could analyze notated scores. These models are highly advanced in object and hand writing identification and it seems reasonable that they could easily be extended to learn the elements of standard musical notation and write new scores visually. The possibilities of direct audio analysis processing remain to be explored as well.

References

Boulanger-Lewandowski, N.; Bengio, Y.; and Vincent, P. 2012. Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. In *Proceedings of the 29th International Conference on Machine Learning (ICML- 12)*, 1159–1166.

Hamel, P., and Eck, D. 2010. Learning features from music audio with deep belief networks. In *Proc. ISMIR*, volume 10, 339–344. Utrecht, The Netherlands.

Humphrey, E. J.; Bello, J. P.; and LeCun, Y. 2012. Moving beyond feature design: Deep architectures and automatic feature learning in music informatics. In *Proc. ISMIR*, 403–408.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *Nature* 521(7553):436–444.

Lee, H.; Pham, P.; Largman, Y.; and Ng, A. Y. 2009. Unsupervised feature learning for audio classification using convolutional deep belief networks. In *Advances in neural information processing systems*, 1096–1104.

Lerdahl, R., and Jackendoff, R. 1983. *A generative theory of tonal music*. MIT Press.

Li, M., and Sleep, R. 2005. Genre classification via an LZ78-based string kernel. In *Proc. ISMIR*, 252–259.

Pasquier, P.; Eigenfeldt, A.; Bown, O.; and Dubnov, S. 2016. An introduction to musical metacreation. *Computers in Entertainment (CIE)* 14(2).

Shmulevich, I., and Povel, D.-J. 2000. Measures of temporal pattern complexity. *Journal of New Music Research* 29(1):61–69.

Smith, B., and Garnett, G. 2012. Improvising musical structure with hierarchical neural nets. In *Proc. AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*.

Smith, B. D. 2013. Tracking creative musical structure: The hunt for the intrinsically motivated generative agent. In *Proc. AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*.

Smolensky, P. 1986. Information processing in dynamical systems: Foundations of harmony theory. Technical Report CU-CS-321-86, DTIC Document.

Taylor, G. W., and Hinton, G. E. 2009. Factored conditional restricted boltzmann machines for modeling motion style. In *Proceedings of the 26th annual international conference on machine learning*, 1025–1032. ACM.

Taylor, G. W.; Hinton, G. E.; and Roweis, S. T. 2007. Modeling human motion using binary latent variables. *Advances in neural information processing systems* 19:1345.

Van den Oord, A.; Dieleman, S.; and Schrauwen, B. 2013. Deep content-based music recommendation. In *Advances in neural information processing systems*, 2643–2651.

Vohra, R.; Goel, K.; and Sahoo, J. 2015. Modeling temporal dependencies in data using a DBN-LSTM. In *Proc. IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 1–4. IEEE.