Journal Pre-proof

IDyOMpy: A new Python-based model for statistical analysis of musical expectations

Guilhem Marion, Fei Gao, Benjamin P. Gold, Giovanni M. Di Liberto, Shihab Shamma

PII: S0165-0270(24)00292-9
DOI: https://doi.org/10.1016/j.jneumeth.2024.110347
Reference: NSM 110347

To appear in: *Journal of Neuroscience Methods*

Received date : 29 May 2024
Revised date : 20 November 2024
Accepted date : 12 December 2024

Please cite this article as: G. Marion, F. Gao, B.P. Gold et al., IDyOMpy: A new Python-based model for statistical analysis of musical expectations. *Journal of Neuroscience Methods* (2024), doi: https://doi.org/10.1016/j.jneumeth.2024.110347.

Highlights

IDyOMpy: A New Python-Based Model for Statistical Analysis of Musical Expectations

Guilhem Marion, Fei Gao, Benjamin P. Gold, Giovanni M. Di Liberto, Shihab Shamma

- We present IDyOMpy: a new Python-based Model based on the IDyOM, a widely used statistical model of music;
- IDyOMpy outperforms the previous Lisp model in terms of generalization error and correlation with behavioral data;
- IDyOMpy makes statistical modeling of music more accessible;
- IDyOMpy offers new features such as prediction of musical rests and training monitoring. Also, its modular design facilitates future modifications.

IDyOMpy: A New Python-Based Model for Statistical Analysis of Musical Expectations

Guilhem Marion[a,b,c], Fei Gao[d], Benjamin P. Gold[e], Giovanni M. Di Liberto[f], Shihab Shamma[a,c]

[a] *Laboratoire des Systèmes Perceptifs, Département d'Étude Cognitive, École Normale Supérieure, PSL, Paris, France*
[b] *Department of Psychology, New York University, New York City, USA*
[c] *Institute for Systems Research, Electrical and Computer Engineering, University of Maryland, College Park, USA*
[d] *Department of Computational Linguistics, University of Zurich, Zurich, Switzerland*
[e] *Neuroimaging & Brain Dynamics Lab, Vanderbilt University, Nashville, USA*
[f] *School of Computer Science and Statistics, Trinity College, The University of Dublin; ADAPT Centre; Trinity College Institute of Neuroscience, Dublin, Ireland*

Abstract

Background: IDyOM (Information Dynamics of Music) is the statistical model of music the most used in the community of neuroscience of music. It has been shown to allow for significant correlations with EEG (Marion, 2021), ECoG (Di Liberto, 2020) and fMRI (Cheung, 2019) recordings of human music listening. The language used for IDyOM -Lisp- is not very familiar to the neuroscience community and makes this model hard to use and more importantly to modify.

New method: IDyOMpy is a new Python re-implementation and extension of IDyOM. This new model allows for computing the information content and entropy for each melody note after training on a corpus of melodies. In addition to those features, two new features are presented: probability estimation of silences and enculturation modeling.

Results: We first describe the mathematical details of the implementation. We extensively compare the two models and show that they generate very similar outputs. We also support the validity of IDyOMpy by using its output to replicate previous EEG and behavioral results that relied on the original Lisp version (Gold, 2019; Di Liberto, 2020; Marion, 2021). Finally, it reproduced the computation of cultural distances between two different datasets as described in previous studies (i.e. Pearce, 2018).

Comparison with existing methods and Conclusions : Our model replicates the previous behaviors of IDyOM in a modern and easy-to-use language -Python. In addition, more features are presented. We deeply think this new version will be of great use to the community of neuroscience of music.

Keywords: IDyOM, Statistical Model of Music, Music Cognition, Expectations
PACS: 0000, 1111
2000 MSC: 0000, 1111

1. Introduction

During the 1950s, the music theorist Leonard Meyer advanced the idea that musical predictions were at the core of music perception[1]. The development of the Predictive Processing[2, 3] has since further elaborated this idea and provided computational formulations for its implementation[4, 5, 6]. This framework revolves around the notion that the brain learns a model of the world that is continuously used to predict sensory inputs. Perception, therefore, occurs at the encounter between sensory inputs and their predictions[7] generating a *prediction error* that is exploited to update the model[8]. This theory rests on two main hypotheses[9]: (1) The Statistical Learning Hypothesis which states that the brain needs to learn and update an internal model of the environment's regularities; (2) The Probabilistic Prediction Hypothesis which postulates that predictions of the sensory inputs are based on the same internal model to modulate their neural encoding and facilitate their perception.

A large number of studies investigate predictions in music[10, 11, 9, 4], speech[12, 13], vision[14, 15], touch[16, 17], and even smell[18], many using computational models to account for human cognition[19, 20] or cortical activity[21, 22, 23]. Speech models are particularly varied and widespread and sometimes even include complex deep neural networks (DNN) implementations [24, 25, 26, 27] that are presumed to reflect different facets of human cognition[28, 29]. The community of music cognition has embraced this approach[30] and already provided evidence for its two hypotheses by demonstrating that explicit[31, 32, 33, 34, 35] and implicit[36, 37, 33, 38, 39, 40, 41, 42, 43, 22, 44, 45, 46] predictions correlate well with the probability of musical events in the listeners' culture[19]. Prediction signals have even been measured during moments of musical silence which correlated well with the probability of the absent note[23]. Moreover, it has also been determined that passive exposure to unfamiliar music engenders statistical learning consistent with the music heard[47, 48, 49, 50, 51]. For instance, passive exposure to Balkan music (chosen because of its non-isochronous time signatures) facilitates in young children and adults the detection of violations in new musical excerpts with similar time signatures [51]. Another study replicated this phenomenon for pitch with adult listeners who gained superior abilities to predict the next note in melodies sampled from random musical grammar after being passively exposed to different melodies sampled from the same musical grammar[48].

In general, predictions have accounted for many other facets of music cognition such as memory[52], emotions[53], reward[54] and neural activity[55, 23, 56] making the predictive coding framework a rich framework for future musical studies [57, 58]. Compared to speech, however, the modeling of music cognition has been dominated by a single powerful model: IDyOM[59] (Information Dynamics of Music), which has been used in myriad of studies about musical prediction and cited in over 300 articles. This model, however, is implemented in Lisp, which is not widely known anymore and is poorly supported on modern computers, making it difficult to use and modify it.

Here we propose a Python implementation of the IDyOM model with improvements such as an alternate technique for merging different Markov chains' orders. We also propose new features that have been used to explore new ideas about the brain, e.g., a model for computing the probability of having melodic notes during silent intervals and a model that monitors learning all along training for different testing datasets. Finally, we provide a quantitative comparison with the original Lisp implementation using both theoretical (based on generalization error) and cognitive (based on EEG decoding and self-reported data) measures. We demonstrate that this new implementation replicates the original Lisp implementation and improves on some of its findings. All codes can be found online[1].

[1] The IDyOMpy model can be found at https://github.com/GuiMarion/IDyOMpy. All the code and data related to the analyses presented here can also be found at https://github.com/GuiMarion/codeForPaper-IDyOMpy-.

2. IDyOM

The Information Dynamics of Music (IDyOM) model, developed by Marcus Pearce and published in 2005[59], is a statistical framework for predicting auditory expectations in music. It computes the expectedness of each note within a given context using measures of *information content* and *entropy*. To learn patterns of musical structure, IDyOM is first trained on a corpus of melodies, using the multiple viewpoint framework by Darrell Conklin [60] and the Prediction by Partial Matching (PPM) data compression algorithm [61, 62]. Once trained, it assesses new pieces by applying these learned statistics in two modes: a long-term model (LTM), which simulates listeners' long-term cultural exposure, and a short-term model (STM), which captures the local statistical features specific to the piece being evaluated.

2.1. Architecture

The model is based on variable-order Markov chains to capture long-term information. They form the two bricks which are the LTM and the STM.. Both of them rest on the same architecture, but the data they are trained on are different. An important limitation of Markov chains is that they are discrete models, making them unsuitable to work on continuous data such as raw audio waveforms or spectrograms. The use of IDyOM is therefore limited to symbolic musical scores.

2.1.1. Variable Order Markov Chains

A Markov chain describes a memoryless[2] process which means that the probability of any event is only a function of the previous one. Formally, for $\forall k, X_k$ sequential random variables,

$$P(X_k = x | X_{k-1}, X_{k-2}, ..., X_0) = P(X_k = x | X_{k-1})$$

Let Σ be the set of all possible notes, referred to as the *alphabet*, borrowing the term from formal languages. $P : \Sigma^2 \to [0, 1]$ is a function for the probabilities of transitions from note to note. Such a model can be expressed as an $n \times n$ matrix or a graph $G = (V, E)$ where V (vertexes) is the set of notes, and E (edges) indicates the transition probabilities. This model is known as a *first-order Markov Chain*.

[2] The next state only depends on the value of the current state. No memory is stored.

Fig. 1.A illustrates a simple example of a graph representation of a first-order Markov chain for music. It expresses the statistical model representing the beginning of the melody of *Au Clair de la Lune* (fig. 1.B).

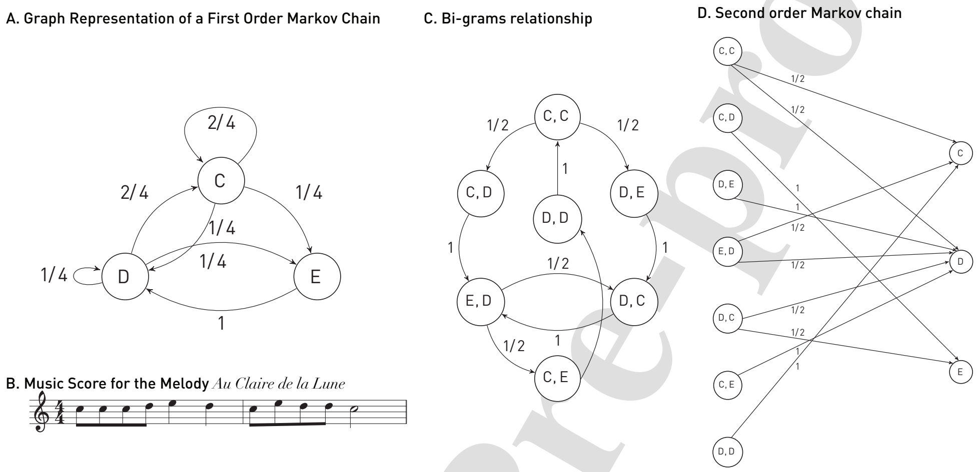

Figure 1: (**A**) Markov chains of order 1 and (**D**) order 2 corresponding to **(B)**, the beginning of the melody *Au Claire de la Lune*. Fractions correspond to the transition probabilities (aka frequencies in the data). (**C**) Probabilistic relationships between the different bi-grams of the data.

Because of the highly structured nature of music, it is reasonable to assume that note probabilities would depend on more than one prior note. Musical sentences are often constructed over a large number of previous notes and thus show long-term dependencies. By using n-grams as the alphabet of the Markov chain, it is still possible to use the Markov model and include long-term dependencies.

An n-gram is an ordered sequence of n elements of the alphabet Σ. For instance, if the alphabet is $\Sigma = \{a, b\}$, all the 2-grams are $\{aa, ab, ba, bb\}$. Formally, it is an element of the Cartesian product of the original set of states Σ. For instance, 2-gram $\in \Sigma \times \Sigma$, 3-gram $\in \Sigma \times \Sigma \times \Sigma$, ..., and so,

$$n\text{-gram} \in \prod_{k=1}^{n} \Sigma$$

By using n-grams as elements of V, the set of states of our Markov Chain, we can define the transition probability between length-n words ω, $\forall n$,

$$P(X_{k:k+n} = \omega | X_{k-n:k})$$

With $X_{n:m}$ the sequence of elements from n to m:

$$X_{n:m} = X_n, ..., X_{m-1}$$

Fig. 1.C shows a graph representation of probabilistic relationships between the 2-grams of the melody *Au Clair de la Lune*. The number of states hugely increased as compared to the 1-gram analysis; more data are needed to train the model accurately, but more complex structures can be captured. To compute the second-order Markov chain, we can collapse this representation by summing across all sequences starting with the same note and therefore get the probability to observe a given single note after a given context (c.f. fig. 1.C):

$$P(X_k = x | X_{k-n:k}) = \sum_{\{\omega\}|\omega_0 = x} P(X_{k:k+n} = \omega | X_{k-n:k})$$

Variable-order Markov chains have the flexibility to use n-grams of different lengths and to dynamically adapt the contribution of each order. The ability to embed n-long temporal dependencies allows for modeling melodic sentences.

2.1.2. Merging Different Orders (Original model)

It is generally difficult to merge all distributions (one per order) into a single one. In the original IDyOM, the Prediction by Partial Matching (PPM) algorithm is used to approximate the final $P(X_k = x | X_{k-n:k})$. PPM[61] is a data compression scheme in which the central component is an algorithm for performing back-off smoothing (escaping when a symbol is found) or interpolated smoothing (always computing a linear combination of predictions from all orders for all symbols) of the distributions from different orders. This technique allows for the probability estimation of events that have yet to be encountered.

For instance, the interpolated smoothing technique used in IDyOM uses the following definition:

$$P(X_k = x | X_{k-n:k}) = \alpha(x | X_{k-n:k}) + \gamma(X_{k-n:k}) \cdot P(X_k = x | X_{k-n+1:k})$$

The functions $\alpha()$ and $\gamma()$ are computed using the PPM algorithm[59](see [63] for the original method). By iterating recursively, they encounter all orders and assign a coefficient to each order. The IDyOM Lispdefault method (called C) is defined by:

$$\gamma(X_{k-n:k}) = \frac{t(X_{k-n:k})}{\#X_{k-n:k} + t(X_{k-n:k})}, \text{ and,}$$
$$\alpha(x|X_{k-n:k}) = \frac{\#X_{k-n:k} \cdot x}{\#X_{k-n:k} + t(X_{k-n:k})}$$

Where $t(C)$ denotes the total number of *symbol types* of Σ that have occurred with non-zero frequency in context C and $\#\omega$ denotes the number of occurrences of the sequence ω in the corpus. Therefore, $\gamma(X_{k-n:k})$ computes the quotient relationship between the prevalence of a context and the variety of symbols occurring after this context and $\alpha(x|X_{k-n:k})$ the frequency of the symbol x in the given context $X_{k-n:k}$.

This method allows one to account for the *diversity* of distributions. Thus, a distribution that only encountered a few n-grams will be less represented than a distribution that saw all the alphabet. Globally this technique tends to favor the lower orders.

2.1.3. Merging Different Orders: The New IDyOMpy Model

Instead of using the PPM algorithm to merge the different orders of the Markov chains, we propose a new method using an arithmetic mean weighted by the inverse of the normalized entropies of the distributions. Our new method accounts for the reliability of each distribution. It weights them by prioritizing more informative contexts which is presumably more effective information than the diversity of the contexts. This method has been successfully used in the original IDyOM implementation to merge the distributions of the STM and LTM. We expect this method to outperform the heuristic weights used in the PPM smoothing algorithm which don't consider the nature of the predictive distribution. We denote by NE_i the normalized entropy of the probability distribution given by the context $X_{k-i:k}$ corresponding to the i^{th}-order model.

$$P(X_k = x) = \frac{\sum_{i=1}^{n} P(X_k = x|X_{k-i:k}) \cdot NE(X_k|X_{k-i:k})^{-1}}{\sum_{i=1}^{n} NE(X_k|X_{k-i:k})^{-1}}$$

The normalized entropy[3] (NE) of a probability distribution is the Shannon entropy (E) normalized by the maximum entropy for its support. This

[3]Note that the original IDyOM implementation uses the term relative entropy, which

maximum entropy corresponds to the entropy of the discrete uniform distribution with the same number of elements, so it depends only on the size of the distribution's support. Normalizing in this way enables entropy comparisons across different orders of Markov chains. As a Markov chain's order increases, the number of represented states typically grows, which can artificially inflate entropy (more terms to sum). Normalization by maximum entropy compensates for this.

$$NE(X) = E(X)/Emax(X)$$

The Shanon entropy of a random variable quantifies the average level of uncertainty or information associated with the variable's potential states or possible outcomes:

$$E(X) = -\sum_{x} P(X = x) \cdot log_2(P(X = x))$$

The maximal entropy is defined by the entropy of the uniform distribution that shares the same support (number of states):

$$Emax(X) = -\sum_{n} 1/n \cdot log_2(1/n) = log_2(n)$$

Our Python implementation includes the option to use both merging techniques (PPM or entropy-based). The manuscript presents analyses of both models. We refer to the PPM method as IDyOMpy PPM and both IDyOMpy AE and IDyOMpy GE both use the entropy-based merging technique.

2.1.4. The Short-Term Model

The short-term model consists exactly of the same computational model as the long-term model described before but is not trained on a corpus. It is trained during the testing phase, therefore, it only takes into account the very local grammar of the tested piece. It is useful for accounting for local structures and repetitions within the pieces that do affect the predictions but do not come from a long-term statistical learning process[64] (key, modulations, theme repetitions, ...). The probability distributions of the short-term model and the long-term model are merged using the arithmetic mean weighted by

we chose to change to normalized entropy to avoid confusion with the Kullback-Leibler, which has a different formulation.

the inverse normalized entropies of the models (as described above for the different orders), with b as an additional parameter that allows for sharpening or smoothing the final distribution[4]

$$P(X_k = x) = \frac{NE_{LTM}^{-b}(X_k) \cdot P_{LTM}(X_k = x) + NE_{STM}^{-b}(X_k) \cdot P_{STM}(X_k = x)}{NE_{LTM}^{-b}(X_k) + NE_{STM}^{-b}(X_k)}$$

With E_{LTM} and E_{STM}, the entropies of respectively the long- and short-term models:

$$E_{LTM}(X_k) = -\sum_x P_{LTM}(X_k = x) \cdot log_2(P_{LTM}(X_k = x))$$

2.1.5. Entropy Approximation

A straightforward implementation would directly compute the entropy of the long- or short-term models to merge them, which is computationally very expensive.

We already computed the entropies of each individual order to merge the Markov chains' of both the LTM and STM, for instance for the LTM:

$$P_{LTM}(X_k = x) = \frac{\sum_{i=1}^{n} P_{LTM}(X_k = x|X_{k-i:k}) \cdot RE_{LTM}(X_k|X_{k-i:k})^{-1}}{\sum_{i=1}^{n} RE_{LTM}(X_k|X_{k-i:k})^{-1}}$$

Therefore, it is useful to find a way to compute the entropy of $P_{LTM}(X_k)$ only from the $E_{LTM}(X_k|X_{k-i:k})$ from all different order i. One possible approach is to use the mean of the self-weighted entropies which proved to be a good approximation that reduced computation times by a factor of 5:

$$E(X_k) = \frac{\sum_{i}^{n} E(X_k|X_{k-i:k}) \cdot E(X_k|X_{k-i:k})^{-1}}{\sum_{i}^{n} E(X_k|X_{k-i:k})^{-1}}$$

This approximation resulted in IC values that were very highly correlated with the IC computed using genuine entropy computations (r=0.87 on Bach

[4]In our implementation we use $b = 1$. This parameter can be changed within the code.

chorales). In addition, we ran the entire set of analyses presented above to compare the versions using the approximation *versus* the genuine computations of the entropies. Those analyses resulted in comparable values, we present them throughout the manuscript. The implementation of IDyOMpy includes both options.

2.2. Viewpoints

Music evolves across 5 dimensions: Pitch, duration, timbre, intensity, and spatialization[5]. IDyOM assumes that those dimensions are independent (or joint as multi-dimensional states) when computing their joint product. While the dimensions most often considered with IDyOM are pitch and duration of the notes, any other feature can be included in the model as long as it is discrete.

$$P(X_k = x) = P(Pitch_k = x_{pitch}) \cdot P(Duration_k = x_{duration})$$

X_k is a valid probability distribution (sums to 1) if $Pitch_k$ and $Duration_k$ are. p_i and d_i are all the different pitches and durations.

$$\sum_{x\in\Sigma} P(X_k = x) = \sum_{p_i}\sum_{d_i} P(Pitch_k = p_i) \cdot P(Duration_k = d_i)$$
$$\sum_{x\in\Sigma} P(X_k = x) = \sum_{p_i} P(Pitch_k = p_i) \cdot \sum_{d_i} P(Duration_k = d_i)$$
$$\sum_{x\in\Sigma} P(X_k = x) = 1$$

2.3. Training

During the training phase, the transition probabilities for each Markov chain are learned from a corpus of melodies by computing the frequencies (counts) over all the different values. For the IDyOMpy model (c.f. 2.1.3), we compute those probabilities directly from those counts. We use $\#\omega$ as the number of occurrences of the sequence ω in the corpus and $\cdot$ as the concatenation operator. Therefore, $\#X_{k-n:k} \cdot x$ denotes the count of words starting with $X_{k-n:k}$ and ending with x in the whole corpus.

[5]Spatialization refers to where we perceive the sounds to come from. For instance, in a string orchestra, violins are usually on the left, and cellos and basses are on the right of the stage. Spatialization became a key element of recorded contemporary music as pieces are delivered as stereo mixes (one track for each ear), with sometimes live changes of the instrument's position in the space. Numerically, spatialization can be encoded as the balance of the signal between the two ears, usually presented as between -50 (all left) to 50 (all right).

$$P(X_k = x|X_{k-n:k}) = \frac{\#X_{k-n}...X_{k-1}X_k}{\#X_{k-n}...X_{k-1}} = \frac{\#X_{k-n:k} \cdot x}{\#X_{k-n:k}}$$

2.4. Measures Computed by the Models

Information Content

The negative log-likelihood of a note x, referred to as its *information content* (IC), represents how well the model predicts it given the context $X_{k-n:k}$. This computation has an interpretation in terms of compressibility or information measurement. For instance, events with high information content are difficult to compress as they rarely occur: one can say that they contain a lot of information. This metric has been shown to provide good measures for psychological interpretations of perceptual data [65, 66].

$$IC(x|X_{k-n:k}) = -log_2(P(X_k = x|X_{k-n:k}))$$

Entropy

The *entropy* provides an approximation of the uncertainty given a context $X_{k-n:k}$. In information theory, this measure evaluates the amount of information contained in a signal (as opposed to an event, as for the IC). In the case of a probability distribution, it reflects the flatness of the distribution given by the model to estimate the confidence of the prediction. If all outcomes are equiprobable (the model cannot gather any information), the entropy will be maximum and the prediction will be highly uncertain. If one outcome has a probability of 1 and all others 0, the entropy will be minimum ($E = 0$) and the prediction is certain. For instance, the first note of a melody is very uncertain as almost all notes are equiprobable (high entropy), whereas, the next note during a repeated sequence is very certain as it is very likely to be the one we heard during the previous repetitions. This measure has been used as a descriptor of musical pieces (i.e. indicator of complexity) and has been shown to provide good measures for behavioral and neural measurements [54, 55].

$$E(X_{k-n:k}) = -\sum_{x \in \Sigma} P(X_k = x|X_{k-n:k}) \cdot log_2(P(X_k = x|X_{k-n:k}))$$

3. Methods For Evaluating Model Performance

To compare our new model with the previous Lisp version, we define a few metrics we will run through all models. We first present *theoretical measures* assessing how well each model generalizes to unseen data, and then assess *cognitive measures* through decoding of EEG recordings of participants listening to music and self-reporting liking of songs.

3.1. Generalization Errors

A common computational approach to evaluate different implementations consists of assessing how well the model generalizes to unseen portions of the dataset (theoretical evaluation), using the negative log-likelihood on testing fold T using k-fold cross-validation. This technique allows us to compare models trained on the same data in terms of computational generalization[6].

$$error = \sum_{x \in T} \frac{-log(P(X_k = x))}{|T|} \tag{1}$$

Using the average negative log-likelihood over unseen data is based on the idea that notes in an unseen score (underlined by the same distribution, i.e., same musical genre) should have on average (because of the law of large numbers) a greater probability than the ones that did not appear. Since the probability distribution must sum to 1, a more accurate distribution should generate a large probability (low negative log-likelihood) on the notes of the score. This is a common technique in the machine learning community[67].

To this end, we used three homogeneous datasets of melodies: Bach chorales, traditional Chinese melodies from the region of Shanxi, and a large database of Western folk melodies. All were sampled from the KernScores website[7]. We used k-fold cross-validation by dividing each dataset into 5 folds, meaning we trained a model on 4 of them and evaluated the remaining one. We then computed the average negative log-likelihood for each song and compared them between models.

3.2. Cultural Distance

IDyOM has proven to be an effective model for musical enculturation because it allows for the modeling of cultural distance [9]. This consists of examining the model's ability at distinguishing between melodies from different cultures. We train two models: one trained on Bach chorales and one trained on traditional music from the Shanxi region of China. We apply test/train splits to compute the incongruent models (i.e. trained on Bach and

[6] Discrete and continuous models cannot be compared using this technique as the support on which they compute the probabilities are different. Continuous models therefore usually result in lower probabilities (higher IC) without being necessarily worse at generalizing.

[7] http://www.esac-data.org/

evaluated on Shanxi music) and cross-validation to compute the congruent models. We extract the average generalization error for each musical excerpt according to both models. Next, we create a scatter plot where the x-axis represents the generalization errors for the Shanxi model, and the y-axis represents the errors for the Bach model, each point corresponding to a specific musical piece. A poorly classifying model would result in most points lying along the equality line, indicating that it cannot distinguish between the two cultures. In contrast, an efficient classifying model would clearly separate the points into two groups on either side of the equality line, successfully differentiating between the two musical corpora.

To quantify the extent to which the two cultures are separated we defined three measures:

Inter-cultural distance (interCD) represents the average Euclidean distance between each point of the first culture and each point of the second culture. A value of 0 means that all points collapse; the bigger the value the further the two cultures are in the model space.

Intra-cultural distance (intraCD) represents how close the pieces are within a culture. It is a proxy for the variability of the generalization error and the stability of the model. Small values mean a more stable model (less variance).

Clustering index $= \frac{interCD}{intraCD(A)/2+intraCD(B)/2}$ combines both inter- and intra-cultural distances into a composite measure that tells to which extent it is easy to classify the two cultures.

3.3. Music Listening EEG Recordings

IDyOM has been widely used in studies of the psychology and neuroscience of music. One recent application that is considered here involved relating note expectation values from IDyOM with EEG signals recorded during music listening. Here, we use publicly available data to replicate two recent studies (Study #1[68] and Study #2[22]) that related IDyOM with EEG through encoding models. We compared the results of the analyses when using the Lisp and Python implementations of IDyOM. Both experiments used a 64-channel Biosemi Active Two System. In Study #1, EEG signals were digitally filtered between 1 and 8 Hz using a Butterworth zero-phase filter (low- and high-pass filters both with order 2 and implemented with the function filtfilt) and down-sampled to 64 Hz. In Study #2, signals were filtered between 0.1 Hz and 30 Hz and down-sampled to 64 Hz. EEG channels with a variance exceeding three times that of the surrounding ones were replaced by an estimate calculated using spherical spline interpolation. All channels were then re-referenced to the average of the two mastoid channels for Study #1 and using global re-referencing for Study #2. The pre-processing was the exact same as in the original study. The stimuli were composed of 10 Bach partitas for Study #1 and 4 Bach chorales for Study #2.

Our analysis was conducted in a similar fashion as in the original studies by estimating temporal response functions (TRFs)[69, 70] with the mTRF-Toolbox[71]. The TRF model is a convolutional model relating an input signal $s(n)$ with an observed EEG signal $eeg(n,k)$ through the convolutional kernels w. Both the observed eeg signal and the estimated kernel w are functions of both times and electrodes. The input signal s is a function of time only, ϵ is the residual error.

$$eeg(n,k) = \sum_{m=0}^{M} s(n) \cdot w_k(n-m) + \epsilon$$

Using matrix rewriting and mean square error optimization, we can approximate w as:

$$\hat{w}_k = (S_k^T S_k)^{-1} S_k^T \cdot eeg_k$$

With S, the Toeplitz matrix of $s(t)$:

$$S = \begin{bmatrix} s(1) & S(N) & \cdots & s(1-N) \\ s(2) & S(1) & \cdots & s(2-N) \\ \vdots & \ddots & \ddots & \vdots \\ s(N) & S(N-1) & \cdots & s(1) \end{bmatrix}$$

This model is used to regress the IC signal computed by both implementations of IDyOM with the pre-processed EEG recordings using k-fold cross-validation. Pearson's correlation is computed between the predicted and original EEG signals.

$$r = corr(s * \hat{w}, eeg)$$

As the predicted EEG signal was only constructed from the IC signal, the correlation indicates the strength of its relationship with the EEG signal. An IC signal that more accurately matches human perception is expected to generate larger EEG prediction correlations. This method provides us with a tool for estimating the physiological validity of each model.

3.4. Behavioral Preference

A recent study[72] showed that the entropy from the IDyOM model explains 19% of the variance of 44 participants' behavioral liking measured using a 7-item Likert scale on 57 stimuli. Mixed-effects models[73] were fit between the mean duration-weighted entropy and the liking for each song.

$$y = X^2\beta + Zu + \epsilon$$

Where,

- y is a known vector of liking observations;
- X is the known design covariate matrix for the fixed effects mean duration-weighted entropy;
- Z is the known design covariate matrix for the random effects due to inter-participant variability (participants are treated as a random effect);
- β is an unknown vector of fixed effects relating X to y;
- u is an unknown vector of random effects, relating Z to y;
- ϵ is an unknown vector of residual random errors;

There was a significant Wundt (quadratic correlation, a.k.a. inverted-U shape well documented in the literature[74]) effect between the liking ratings for the songs and the mean duration-weighted Entropy of the same songs. We, therefore, used these data as a way to estimate the validity of the entropies computed by our model. To do so, we replicated the results of this study on the same observation data, and both models (Lisp and IDyOMpy) trained on the same musical corpus. We then compared the r^2 (variance explained by the entropy as computed by the mixed-effects models) using both Lisp and IDyOM implementations.

$$r^2 = 1 - \frac{\sigma^2}{\sigma_0^2} = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2}$$

With, $\hat{y}$ the approximation of y using the optimized β, u, and ϵ.

To compute the significance of the difference between the two models, we computed the distribution for each model using a Bootstrap method. We computed the r^2 values of sub-sampled data (80% sampled from participants and songs) 5000 times using the same indexes for each model. We then computed the difference distribution and the p-value for it being inferior or equal to 0 by counting to proportion of observations respecting the null hypothesis H_0 ($o_i \leq 0$). This p-value is reported in the result section. We also report the individual p-values computed during the correlations.

The same study also showed a Wundt effect between the IC and the liking ratings. However, in our replication, our model explained more variance using a linear regression instead of a quadratic one. It was therefore hard to compare the two, and we decided not to include the IC analysis in this study.

4. New Features

4.1. Missing Notes Detection

The predictive coding framework states that brain responses during music listening reflect a comparison between the bottom-up sensory responses and top-down prediction signals generated by an internal model that embodies the music exposure and expectations of the listener[75]. To attain a clear view of these predictive responses, previous work has eliminated the sensory inputs by inserting artificial silences (or sound omissions) that leave behind only the corresponding predictions of the thwarted expectations[76, 77]. We propose a new alternate approach that uses the IDyOMpy model to detect the natural silences during existing pieces that exhibit high probabilities of containing a note.

One of our recent studies used this technique to show that it is possible to decode predictions from EEG recordings during musical silences that IDyOM predicted to contain a note. Moreover, the amplitude of those neural responses was correlated with the probabilities computed by the model[23]. We present here this new feature of the IDyOMpy model: the *missing notes detection feature*.

This feature only uses the duration viewpoint by computing the probability distribution of the durations of the notes. It allows us to compute the probability of having a note played during the natural silences of an existing musical piece based on both the statistics of a training corpus (long-term) and the local statistics of the existing piece (short-term). This model uses conditional probabilities to explore all the different potential combinations of notes on all subdivisions of the tempo (up to 1/32th of a note) that lead to a note during the targeted silence. A detection threshold is set (0.2 by default) that will only take into account notes that have a probability higher than the threshold. Figure 2 shows examples of four Bach chorales run with this feature.

The probability P_s of having a note played during the silence at time t is driven by the probability distribution of the duration of the last note played at time t'.

$$P_s(X = t) = P(Duration_{t'} = t - t')$$

This feature is native to the IDyOMpy model.

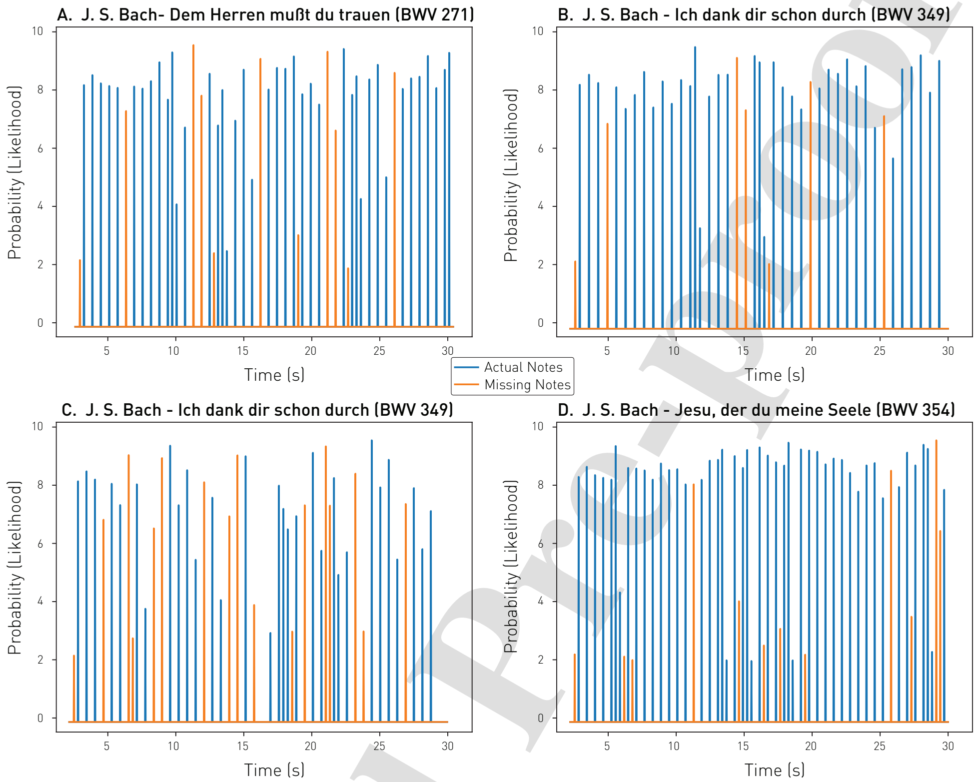

Figure 2: **Missing Notes Detection**: The model was run on the *duration* dimension to compute the probability of a note occurring at each tempo sub-division. A threshold of 0.2 was applied. Blue lines represent the likelihood of actual notes, and orange lines indicate probable notes during silences. Time is converted to seconds for readability.

4.2. Training Monitoring

Another new feature is the *Training Monitoring*. It allows monitoring of the training of the model by computing the corpus-specific mean IC at each update for a given corpus. The model can be trained on different corpora to emulate musical enculturation or look at the interactions between two inter-corpus statistics. One can, therefore, assess the amount of data needed for model convergence. Also, since it is possible to initialize the model with another dataset, this feature is a good way to compare inter- and intra-corpus variability. Figure 3 demonstrates results from two datasets of traditional Chinese music versus a large corpus of Western music. Finally, this monitoring can serve as a model for learning new music and musical enculturation as it simulates the learning of an unfamiliar musical grammar on top of an already familiar one. We used 2-fold cross-validation to compute the generalization errors, and it is possible to choose the number of pieces to test on. Note that the results may noticeably change depending on how the two sets are chosen (randomly done here). It is therefore recommended to compute the learning trace several times with different partitions of the data and take the average as the final trace. Also, note that IDyOM Lisp already partially offers this feature as incremental prediction performance during training tested on the same dataset.

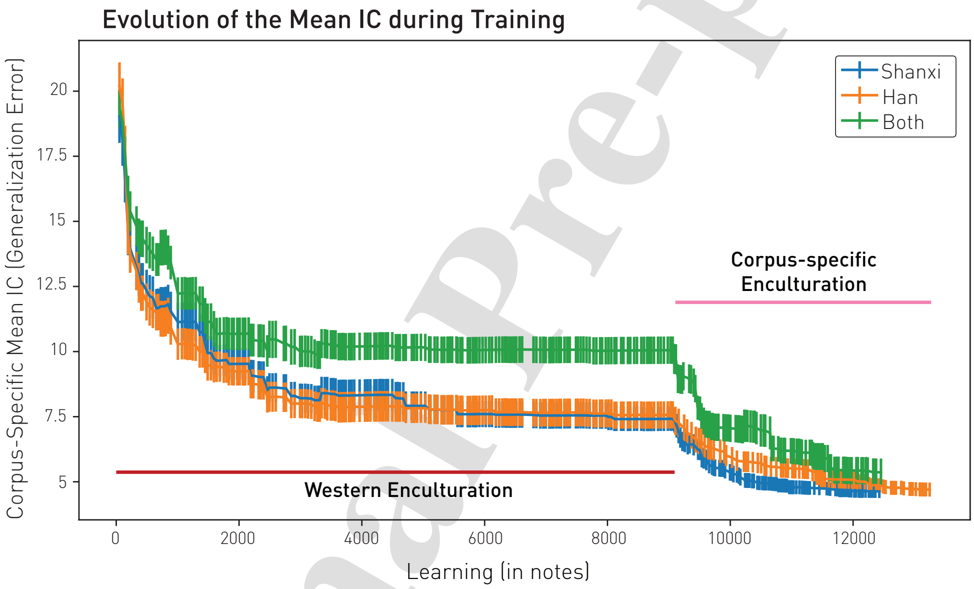

Figure 3: **Training Monitoring**: The model was first trained on Western melodies, then on three corpora (Shanxi, Han, or a mix of traditional Chinese music). There are therefore two training phases: common western enculturation and three different corpus-specific enculturations. Each line shows the generalization error during the training of those two phases. Deeper lines during Western Enculturation reflect how well the Western corpus accounts for Chinese grammar. Higher lines (e.g., for the mixed dataset) indicate more variability. The difference between convergence plateaus during Western and Corpus-Specific Enculturation reflects how much the specific corpus changed the model, serving as a proxy for corpus similarity.

5. Results

In this section, we present several analyses based on theoretical measures, as well as prediction of behavioral and neural data. Four models are investigated: IDyOM Lisp, the original Lisp implementation; IDyOMpy PPM the Python implementation of the original model, which uses the PPM algorithm instead of our entropy-weighted method to merge the different Markov chains orders; IDyOM py Approximated Entropies (AE) it is the new IDyOMpy model using the entropy weighted merging method, it also uses our entropy approximation method that reduces the computation time by a factor of 5; IDyOMpy Genuine Entropies (GE) which use the genuine calculation of the entropies instead of the approximation.

5.1. Information Content

We first used the generalization error (c.f. 3.1) to compare the models on different datasets (Fig. 4.A). We found that the new IDyOMpy models significantly outperformed the previous implementation in all three datasets: traditional Chinese music from Shanxi, Bach chorales, and a large Western database. IDyOM Lisp and IDyOMpy PPM performed similarly, suggesting that our new method for merging the Markov chains is more efficient than the PPM algorithm used in the Lisp version. IDyOMpy GE outperformed IDyOMpy AE, suggesting that using the genuine entropy calculation results in a more efficient model. We also used our new feature *Training Monitoring* (c.f. 5.2) to compare the trace of the generalization error over the course of the training. We observed that the final point of the Lisp model is reached with fewer data for IDyOMpy AE (Fig. 4.B). Finally, we correlated the raw IC for each note of each Bach chorale between the two models. We found a relatively strong correlation of $r = 0.7$ indicating that the two models are consistent but not identical.

We then plotted the cultural distance between traditional Chinese music from Shanxi and Bach chorales for IDyOM Lisp and IDyOMpy AE (Fig. 5.A&B). IDyOM Lisp outperformed all IDyOMpy models on all inter-cultural and intra-cultural distances (Table 1). IDyOMpy AE and IDyOMpy GE resulted in similar values.

Finally, we used the mTRF toolbox to predict EEG recordings of participants listening to Western music (in two different studies, c.f. 3.3) from the IC signal computed with IDyOM Lisp and IDyOMpy AE We found no significant difference in the accuracy (Fig. 5.C&D). However, we should note that the EEG recordings are extremely noisy signals and it is likely that subtle differences in the ICs would not result in significant differences in EEG predictions.

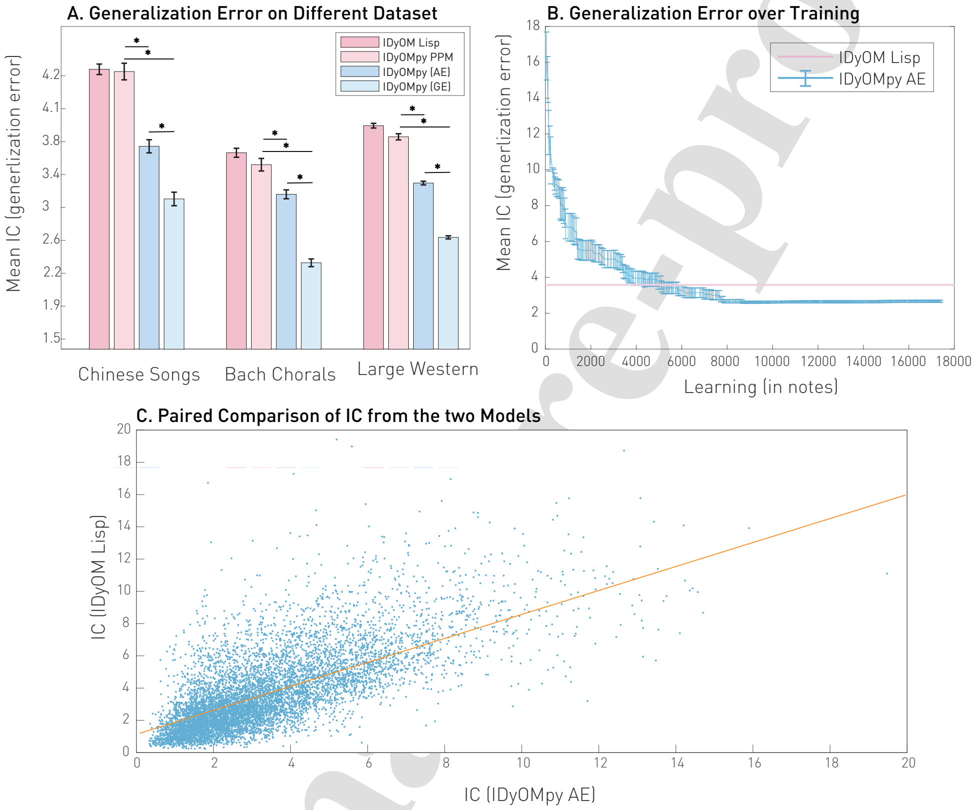

Figure 4: **Comparison of the Generalization Errors.** **A**: Average generalization error for different datasets ($* : p < 10^{-10}$). AE: Approximated Entropies. GE: Genuine Entropies. **B**; Generalization error over the course of the training of the model (training monitoring). **C**: Correlation of the IC for each note. Pearson's $r = 0.7$

	Inter-Cultural Distance	Intra-Cultural Distance on A	Intra-Cultural Distance on B	Clustering Index
IDyOM Lisp	2.5818	1.1439	0.66443	2.8554
IDyOMpy PPM	2.2866	1.4042	0.89402	1.9899
IDyOMpy AE	1.6777	1.6441	0.69206	1.4363
IDyOMpy GE	1.7282	1.5855	0.54608	1.6216

Table 1: **Cultural Classification Metrics for Both Models.** The metrics are defined in 3.2.

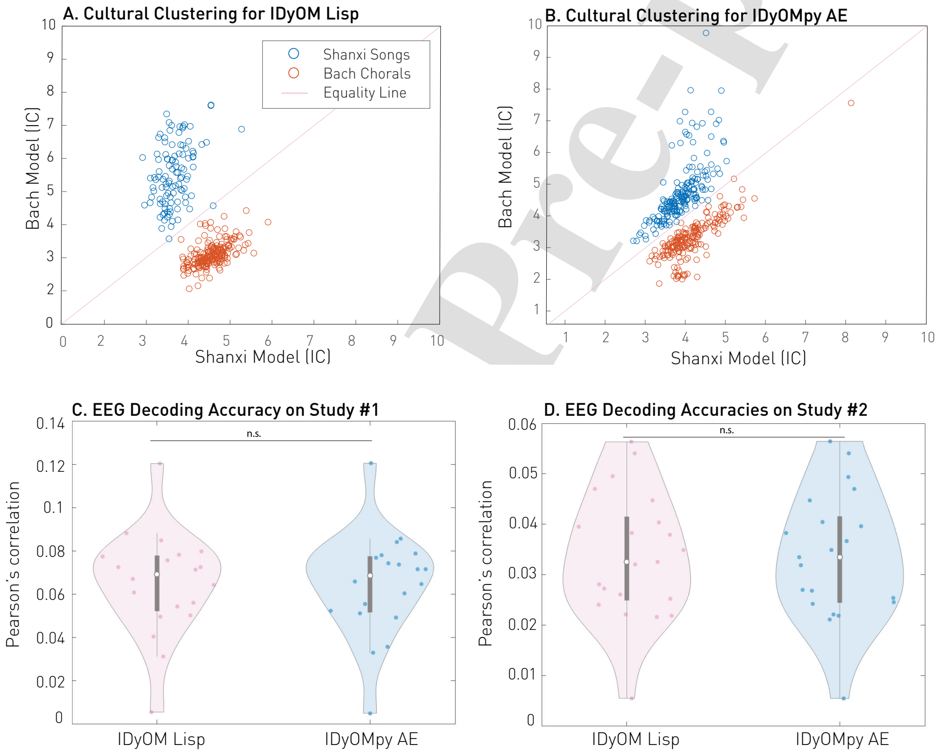

Figure 5: **Accuracies for cultural clustering and EEG decoding. A & B**: We plotted the piece-averaged IC for both an instance of the models trained on Shanxi traditional music (Chinese model) and an instance trained on Bach chorales (Bach model). Detailed values are noted in Table 1. **C & D**: We used the mTRF toolbox to encode the IC from each model (IDyOM Lisp and IDyOMpy AE) trained on the same large Western database into EEG recordings of participants listening to Western music (not in the training dataset). We did not observe any significant difference between the models.

5.2. Entropy

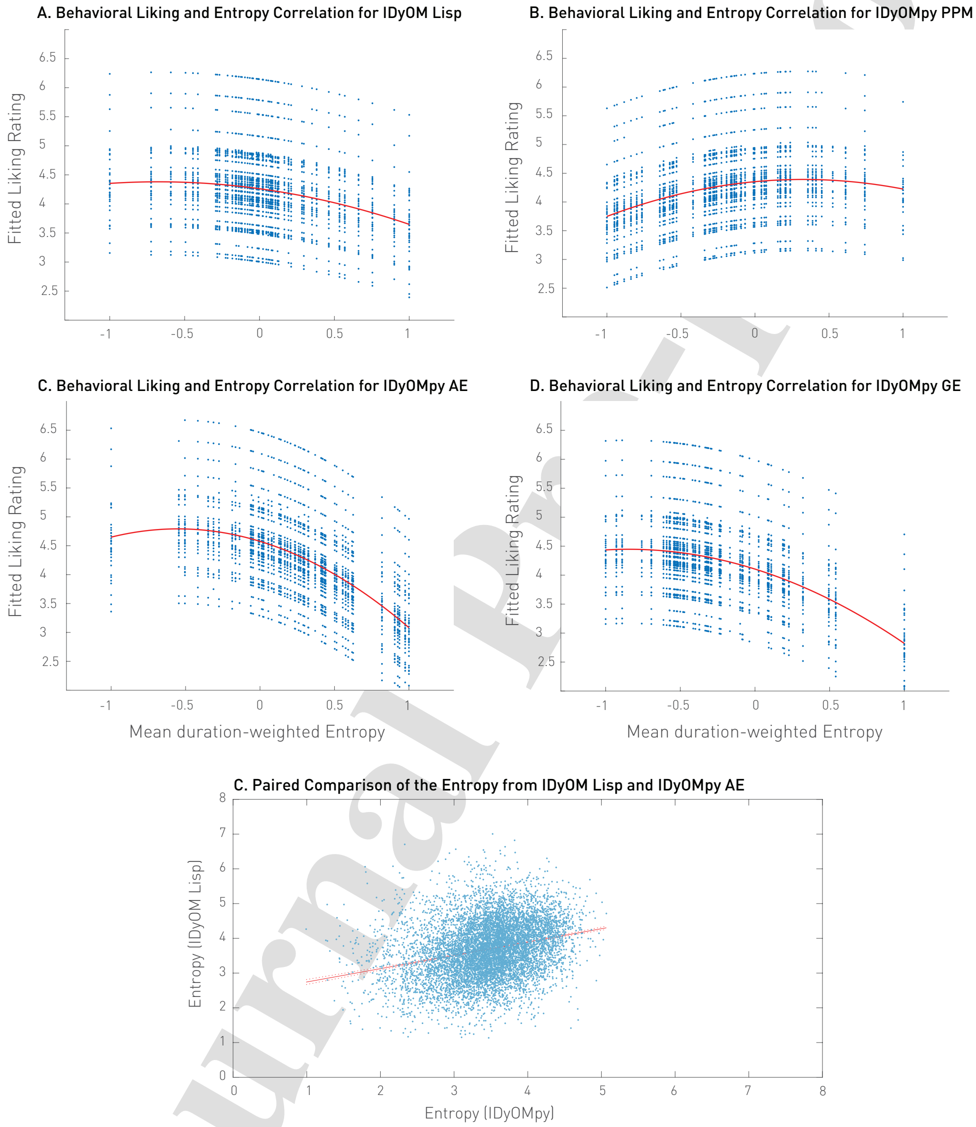

Figure 6: **Comparison and Validation of the Entropy. A, B, C & D**: Correlation of the Entropy from IDyOM Lisp, IDyOMpy PPM, IDyOMpy AE, and IDyOMpy GE respectively with the self-reported liking ratings from [54]. IDyOM Lisp explained 20% of the variance, IDyOMpy PPM 21%, IDyOMpy AE 29% and IDyOMpy GE 23%. IDyOM Lisp and IDyOMpy PPM did not significantly explain different amounts of variance. IDyOMpy AE significantly outperformed all other models. **C** : Correlation of the Entropy for each note. Pearson's $r = 0.3$

To compare the Entropies computed by both models, we first correlated the raw estimates from the two models for each note of each Bach chorale. We found a relatively weak correlation of r=0.31 (Fig. 6.C) indicating that the two models compute the entropy differently. We then used data from [54] to assess which model explains the most variance of the behavioral liking ratings (c.f. 3.4 for method). We found that the new IDyOMpy AE model explains 29% of the variance compared to 20% for the Lisp version (Fig. 6.A&B). A bootstrap procedure was used: we repeated the procedure 3000 times with sub-portions of the data (same sub-portions for each model) and generated a distribution of the adjusted r^2. The distributions for the two models did not overlap and resulted in a significant t-test; $p < 0.0001$. This result leads us to conclude that even if the two models compute Entropy somewhat differently, they both replicate results from (Gold et. al., 2019) and that IDyOMpy even outperforms the Lisp implementation in terms of variance explained giving it a cognitive validation of the Entropy computations of both models. IDyOM Lisp and IDyOMpy PPM did not significantly differ in terms of variance explained. However, IDyOMpy AE outperformed all models.

6. Discussion

In this report, we have presented IDyOMpy, a new statistical model of music in Python, based on the IDyOM architecture. This implementation differs in the way that the different Markov chains (for each order) are merged using an entropy-weighted linear combination (c.f. 2.1.3) and not the PPM algorithm as in the Lisp version (c.f. 2.1.2). By comparing IDyOMpy PPM and IDyOMpy we showed that, except for cultural classification, the use of our new entropy-weighted merging algorithm is more efficient than the PPM algorithm for combining different Markov chains orders. We also propose a way to approximate the entropy that reduces the computation time by at least a factor of 4 and only slightly affects the results discussed in this study. However, our findings indicate that calculating the exact entropies enhances IC generalization, though it reduces the entropy-based variance explanation of behavioral liking. This suggests that using entropy approximations should be carefully balanced depending on the specific application and the dataset size.

Our new IDyOMpy model generates overall comparable or superior results and allows for significant future improvements. We first showed that it performs better regarding generalization errors and the amount of training data needed to converge the model. Additionally, we showed that the IC computed from the two models were relatively close ($r = 0.7$, c.f. Fig 2) and resulted in comparable results for two EEG decoding experiments (c.f. Fig 3) thus confirming their consistent physiological relevance.

We showed that, even if the entropies only weakly correlate between the two models ($r = 0.3$), IDyOMpy generates results with better correlation with the behavioral data of self-reported liking ratings (c.f. Fig 4), thus providing a cognitive validation of the model's outcomes. In addition, we presented two original new features (missing notes detection and training monitoring, c.f. Section 5) that will be very useful for our community. A limitation of IDyOMpy that still needs to be resolved is the analysis of musical performances through audio recordings based solely on a rich representation of the sound (e.g. spectrograms as opposed to extracted features).

Finally, this Python implementation is easy to install, understand, and use from any computer and operating system. Furthermore, its internal code can be rapidly augmented with new features, as demonstrated in the present work. In sum, we think that IDyOMpy will be of high interest to the community and will facilitate rapid progress in the field of computational music cognition.

7. Ethics

Both EEG experiments were undertaken at Ecole Normale Supérieure in Paris, France following the Declaration of Helsinki and approved by the CERES committee of Paris Descartes University (CERES 2013-11); c.f. [22] and [23] for more information. The behavioral experiment was conducted at McGill in Montreal, Canada under ethics approval of the university; c.f. [72] for more information.

8. Acknowledgment

Author contributions: G.M. designed research; G.M. performed research; G.M. and G.M.D.L. and B.G. contributed unpublished reagents/analytic tools; G.M., G.M.D.L., F.G, and B.G. analyzed data; B.G. provided the behavioral data for the entropy analysis and wrote the initial analysis script for those data. F.G helped implement the PPM method within IDyOMpy and ran additional analyses required during the reviewing phase. G.M. wrote the first draft of the paper; G.M., G.M.D.L., B.G., and S.A.S. edited the paper.

We would like to thank Marcus Pearce (author of the first Lisp version of IDyOM) for the nice discussions and proofreading of the manuscript that greatly improved the quality of this manuscript. This work was supported by Advanced ERC Grant NEUME 787836 and Air Force Office of Scientific Research and National Science Foundation grants to S.A.S.; and FrontCog Grant ANR-17-EURE-0017, PSL Idex ANR-10- IDEX-0001-02, and a PhD scholarship from the Research Chair on Beauty Studies PSL–L'Oréal to G.M. The authors declare no competing financial interests. The code for IDyOMpy is available at: https://github.com/GuiMarion/IDyOMpy The code for the analysis is available at: https://github.com/GuiMarion/codeForPaper-IDyOMpy-
Correspondence should be addressed to Guilhem Marion at gm3557@nyu.edu

References

[1] L. Meyer, Emotion and meaning in music, University of Chicago Press, 1956.

[2] K. J. Friston, J. Daunizeau, J. Kilner, S. J. Kiebel, Action and behavior: a free-energy formulation, Biological cybernetics 102 (3) (2010) 227–260.

[3] A. Clark, Whatever next? predictive brains, situated agents, and the future of cognitive science, Behavioral and brain sciences 36 (3) (2013) 181–204.

[4] P. Vuust, O. Heggli, K. Friston, M. Kringelbach, Music in the brain, Nature Reviews Neuroscience 23 (03 2022). doi:10.1038/s41583-022-00578-5.

[5] M. A. Rohrmeier, S. Koelsch, Predictive information processing in music cognition. a critical review, International Journal of Psychophysiology 83 (2) (2012) 164–175.

[6] S. Koelsch, P. Vuust, K. Friston, Predictive processes and the peculiar case of music, Trends in Cognitive Sciences 23 (1) (2019) 63–77.

[7] G. B. Keller, T. D. Mrsic-Flogel, Predictive processing: a canonical cortical computation, Neuron 100 (2) (2018) 424–435.

[8] R. Näätänen, P. Paavilainen, T. Rinne, K. Alho, The mismatch negativity (mmn) in basic research of central auditory processing: a review, Clinical neurophysiology 118 (12) (2007) 2544–2590.

[9] M. T. Pearce, Statistical learning and probabilistic prediction in music cognition: mechanisms of stylistic enculturation, Annals of the New York Academy of Sciences (1423) (2018).

[10] M. A. Rohrmeier, S. Koelsch, Predictive information processing in music cognition. a critical review, International Journal of Psychophysiology 83 (2) (2012) 164–175.

[11] I. H. Witten, L. C. Manzara, D. Conklin, Comparing human and computational models of music prediction, Computer Music Journal 18 (1) (1994) 70–80.
URL http://www.jstor.org/stable/3680523

[12] D. Poeppel, The maps problem and the mapping problem: Two challenges for a cognitive neuroscience of speech and language, Cognitive Neuropsychology 29 (1-2) (2012) 34–55.

[13] D. Norris, J. M. McQueen, A. Cutler, Prediction, bayesian inference and feedback in speech recognition, Language, Cognition and Neuroscience 31 (1) (2016) 4–18.

[14] J. T. Enns, A. Lleras, What's next? new evidence for prediction in human vision, Trends in Cognitive Sciences 12 (9) (2008) 327–333.

[15] M. Kimura, Visual mismatch negativity and unintentional temporal-context-based prediction in vision, International Journal of Psychophysiology 83 (2) (2012) 144–155.

[16] K. Kilteni, H. H. Ehrsson, Sensorimotor predictions and tool use: Hand-held tools attenuate self-touch, Cognition 165 (2017) 1–9.

[17] R. I. Schubotz, Prediction of external events with our motor system: towards a new framework, Trends in Cognitive Sciences 11 (5) (2007) 211–218.

[18] C. Zelano, A. Mohanty, J. A. Gottfried, Olfactory predictive codes and stimulus templates in piriform cortex, Neuron 72 (1) (2011) 178–187.

[19] C. L. Krumhansl, P. Toivanen, T. Eerola, P. Toiviainen, T. Järvinen, J. Louhivuori, Cross-cultural music cognition: Cognitive methodology applied to north sami yoiks, Cognition 76 (1) (2000) 13–58.

[20] J. S. Nixon, F. Tomaschek, Prediction and error in early infant speech learning: A speech acquisition model, Cognition 212 (2021) 104697.

[21] M. P. Broderick, A. J. Anderson, G. M. Di Liberto, M. J. Crosse, E. C. Lalor, Electrophysiological correlates of semantic dissimilarity reflect the comprehension of natural, narrative speech, Current Biology 28 (5) (2018) 803–809.e3.

[22] G. Marion, G. M. Di Liberto, S. A. Shamma, The music of silence. part i: Responses to musical imagery encode melodic expectations and acoustics, Journal of Neuroscience (2021). arXiv:https://www.jneurosci.org/content/early/2021/07/23/JNEUROSCI.0183-21.2021.full.pdf, doi:10.1523/JNEUROSCI.0183-21.2021.
URL https://www.jneurosci.org/content/early/2021/07/23/JNEUROSCI.0183-21.2021

[23] G. M. Di Liberto, G. Marion, S. A. Shamma, The music of silence. part ii: Music listening induces imagery responses, Journal of Neuroscience (2021). arXiv:https://www.jneurosci.org/content/early/2021/07/22/JNEUROSCI.0184-21.2021.full.pdf, doi:10.1523/JNEUROSCI.0184-21.2021.
URL https://www.jneurosci.org/content/early/2021/07/22/JNEUROSCI.0184-21.2021

[24] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space (2013). doi:10.48550/ARXIV.1301.3781.
URL https://arxiv.org/abs/1301.3781

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need (2017). doi:10.48550/ARXIV.1706.03762.
URL https://arxiv.org/abs/1706.03762

[26] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners (2019).
URL https://www.semanticscholar.org/paper/Language-Models-are-Unsupervised-Multitask-Learners-Radford-Wu/9405cc0d6169988371b2755e573cc28650d14dfe

[27] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, W. E. Sayed, Mistral 7b (2023). arXiv:2310.06825.

[28] A. Goldstein, Z. Zada, E. Buchnik, M. Schain, A. Price, B. Aubrey, S. A. Nastase, A. Feder, D. Emanuel, A. Cohen, A. Jansen, H. Gazula, G. Choe, A. Rao, C. Kim, C. Casto, L. Fanda, W. Doyle, D. Friedman, P. Dugan, L. Melloni, R. Reichart, S. Devore, A. Flinker, L. Hasenfratz, O. Levy, A. Hassidim, M. Brenner, Y. Matias, K. A. Norman, O. Devinsky, U. Hasson, Shared computational principles for language processing in humans and deep language models, Nature Neuroscience 25 (3) (2022) 369–380.

[29] C. Caucheteux, J.-R. King, Brains and algorithms partially converge in natural language processing, Communications Biology 5 (1) (2022) 134.

[30] M. T. Pearce, G. A. Wiggins, Auditory expectation: The information dynamics of music perception and cognition, Topics in Cognitive Science 4 (4) (2012) 625–652. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1756-8765.2012.01214.x, doi:https://doi.org/10.1111/j.1756-8765.2012.01214.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1756-8765.2012.01214.x

[31] A. R. Fogel, J. C. Rosenberg, F. M. Lehman, G. R. Kuperberg, A. D. Patel, Studying musical and linguistic prediction in comparable ways: The melodic cloze probability method, Frontiers in psychology 6 (2015) 1718.

[32] D. R. Sears, M. T. Pearce, W. E. Caplin, S. McAdams, Simulating melodic and harmonic expectations for tonal cadences using probabilistic models, Journal of New Music Research 47 (1) (2018) 29–52.

[33] K. A. Corrigall, B. Tillmann, E. G. Schellenberg, Measuring children's harmonic knowledge with implicit and explicit tests, Music Perception: An Interdisciplinary Journal 39 (4) (2022) 361–370.

[34] E. Morgan, A. Fogel, A. Nair, A. D. Patel, Statistical learning and gestalt-like principles predict melodic expectations, Cognition 189 (2019) 23–34.

[35] M. T. Pearce, G. A. Wiggins, Expectation in melody: The influence of context and learning, Music Perception 23 (5) (2006) 377–405.

[36] N. Politimou, P. Douglass-Kirk, M. Pearce, L. Stewart, F. Franco, Melodic expectations in 5-and 6-year-old children, Journal of Experimental Child Psychology 203 (2021) 105020.

[37] D. Omigie, M. T. Pearce, L. Stewart, Tracking of pitch probabilities in congenital amusia, Neuropsychologia 50 (7) (2012) 1483–1493.

[38] R. Bianco, B. Gold, A. Johnson, V. Penhune, Music predictability and liking enhance pupil dilation and promote motor learning in non-musicians, Scientific reports 9 (1) (2019) 1–12.

[39] R. Bianco, L. E. Ptasczynski, D. Omigie, Pupil responses to pitch deviants reflect predictability of melodic sequences, Brain and Cognition 138 (2020) 103621.

[40] M. T. Pearce, M. H. Ruiz, S. Kapasi, G. A. Wiggins, J. Bhattacharya, Unsupervised statistical learning underpins computational, behavioural, and neural manifestations of musical expectation, NeuroImage 50 (1) (2010) 302–313. doi:https://doi.org/10.1016/j.neuroimage.2009.12.019.
URL https://www.sciencedirect.com/science/article/pii/S1053811909013068

[41] G. M. Di Liberto, C. Pelofi, R. Bianco, P. Patel, A. D. Mehta, J. L. Herrero, A. de Cheveigné, S. Shamma, N. Mesgarani, Cortical encoding of melodic expectations in human temporal cortex, eLife 9 (2020) e51784.

[42] D. Omigie, M. T. Pearce, V. J. Williamson, L. Stewart, Electrophysiological correlates of melodic processing in congenital amusia, Neuropsychologia 51 (9) (2013) 1749–1762.

[43] A. R. Halpern, I. Zioga, M. Shankleman, J. Lindsen, M. T. Pearce, J. Bhattacharya, That note sounds wrong! age-related effects in processing of musical expectation, Brain and cognition 113 (2017) 1–9.

[44] D. R. Quiroga-Martinez, N. C. Hansen, A. Højlund, M. Pearce, E. Brattico, P. Vuust, Decomposing neural responses to melodic surprise in

musicians and non-musicians: evidence for a hierarchy of predictions in the auditory system, NeuroImage 215 (2020) 116816.

[45] D. R. Quiroga-Martinez, N. C. Hansen, A. Højlund, M. Pearce, E. Brattico, P. Vuust, Musical prediction error responses similarly reduced by predictive uncertainty in musicians and non-musicians, European Journal of Neuroscience 51 (11) (2020) 2250–2269.

[46] D. Omigie, M. Pearce, K. Lehongre, D. Hasboun, V. Navarro, C. Adam, S. Samson, Intracranial recordings and computational modeling of music reveal the time course of prediction error signaling in frontal and temporal cortices, Journal of Cognitive Neuroscience 31 (6) (2019) 855–873.

[47] P. Loui, D. Wessel, Learning and liking an artificial musical system: Effects of set size and repeated exposure, Musicae Scientiae 12 (2) (2008) 207–230.

[48] P. Loui, D. L. Wessel, C. L. H. Kam, Humans rapidly learn grammatical structure in a new musical scale, Music perception 27 (5) (2010) 377–388.

[49] P. Loui, Learning and liking of melody and harmony: Further studies in artificial grammar learning, Topics in Cognitive Science 4 (4) (2012) 554–567.

[50] P. Loui, D. Wessel, C. H. Kam, Acquiring new musical grammars: a statistical learning approach (2006).

[51] E. E. Hannon, S. E. Trehub, Tuning in to musical rhythms: Infants learn more readily than adults, Proceedings of the National Academy of Sciences 102 (35) (2005) 12639–12643. arXiv:https://www.pnas.org/content/102/35/12639.full.pdf, doi:10.1073/pnas.0504254102. URL https://www.pnas.org/content/102/35/12639

[52] K. Agres, S. Abdallah, M. Pearce, Information-theoretic properties of auditory sequences dynamically influence expectation and memory, Cognitive Science 42 (2018) 43–76. doi:10.1111/cogs.12477.

[53] S. Sauvé, A. Sayed, R. Dean, M. Pearce, Effects of pitch and timing expectancy on musical emotion, Psychomusicology: Music, Mind, and Brain 28 (08 2018). doi:10.1037/pmu0000203.

[54] B. P. Gold, E. Mas-Herrero, Y. Zeighami, M. Benovoy, A. Dagher, R. J. Zatorre, Musical reward prediction errors engage the nucleus accumbens and motivate learning, Proceedings of the National Academy of Sciences 116 (8) (2019) 3310–3315.

[55] V. K. Cheung, P. M. Harrison, L. Meyer, M. T. Pearce, J.-D. Haynes, S. Koelsch, Uncertainty and surprise jointly predict musical pleasure and amygdala, hippocampus, and auditory cortex activity, Current Biology 29 (23) (2019) 4084–4092.

[56] R. J. Zatorre, V. N. Salimpoor, From perception to pleasure: Music and its neural substrates, Proceedings of the National Academy of Sciences 110 (Supplement 2) (2013) 10430–10437. arXiv:https://www.pnas.org/content/110/Supplement_2/10430.full.pdf, doi:10.1073/pnas.1301228110.
URL https://www.pnas.org/content/110/Supplement_2/10430

[57] C. Pelofi, G. Marion, G. Di Liberto, P. Ripolles, S. Shamma, Cross-cultural perspectives on the predictive coding perspective of music perception, In prep.

[58] P. Vuust, O. A. Heggli, K. J. Friston, M. L. Kringelbach, Music in the brain, Nature Reviews Neuroscience 23 (5) (2022) 287–305.

[59] M. T. Pearce, The construction and evaluation of statistical models of melodic structure in music perception and composition, Ph.D. thesis (2005).

[60] D. Conklin, I. H. Witten, Multiple viewpoint systems for music prediction, Journal of New Music Research 24 (1) (1995) 51–73.

[61] J. Cleary, I. Witten, Data compression using adaptive coding and partial string matching, IEEE Transactions on Communications 32 (4) (1984) 396–402. doi:10.1109/TCOM.1984.1096090.

[62] S. Bunton, Semantically Motivated Improvements for PPM Variants, The Computer Journal 40 (1997) 76–93. arXiv:https://academic.oup.com/comjnl/article-pdf/40/2_and_3/76/1032717/400076.pdf, doi:10.1093/comjnl/40.2_and_3.76.
URL https://doi.org/10.1093/comjnl/40.2_and_3.76

[63] A. Moffat, Implementing the ppm data compression scheme, IEEE Transactions on Communications 38 (11) (1990) 1917–1921. doi:10.1109/26.61469.

[64] D. Conklin, Prediction and entropy of music, University of Calgary, 1990.

[65] N. Chater, P. Vitányi, Simplicity: a unifying principle in cognitive science?, Trends in Cognitive Sciences 7 (2003) 19–22.

[66] F. Attneave, Some informational aspects of visual perception., Psychological review 61 3 (1954) 183–93.

[67] C. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, C. Hawthorne, A. M. Dai, M. D. Hoffman, D. Eck, An improved relative self-attention mechanism for transformer with application to music generation, CoRR abs/1809.04281 (2018). arXiv:1809.04281.
URL http://arxiv.org/abs/1809.04281

[68] G. M. D. Liberto, C. Pelofi, R. Bianco, P. Patel, A. D. Menhta, J. L. Herrero, N. Mesgarani, A. de Cheveigné, S. Shamma, Cortical encoding of melodic expectations in human temporal cortex, in preparation, url:https://www.biorxiv.org/content/10.1101/714634v2.
URL https://www.biorxiv.org/content/10.1101/714634v2

[69] E. C. Lalor, A. J. Power, R. B. Reilly, J. J. Foxe, Resolving precise temporal processing properties of the auditory system using continuous stimuli, Journal of Neurophysiology 102 (1) (2009) 349–359, pMID: 19439675. arXiv:https://doi.org/10.1152/jn.90896.2008, doi:10.1152/jn.90896.2008.
URL https://doi.org/10.1152/jn.90896.2008

[70] N. Ding, J. Z. Simon, Neural coding of continuous speech in auditory cortex during monaural and dichotic listening, Journal of Neurophysiology 107 (1) (2012) 78–89, pMID: 21975452. arXiv:https://doi.org/10.1152/jn.00297.2011, doi:10.1152/jn.00297.2011.
URL https://doi.org/10.1152/jn.00297.2011

[71] M. J. Crosse, G. M. D. Liberto, E. C. Lalor, The multivariate temporal response function (mtrf) toolbox: a matlab toolbox for relating neural

signals to continuous stimuli, Frontiers of Human Neuroscience 10 (604) (2016).

[72] B. P. Gold, M. T. Pearce, E. Mas-Herrero, A. Dagher, R. J. Zatorre, Predictability and uncertainty in the pleasure of music: a reward for learning?, Journal of Neuroscience 39 (47) (2019) 9397–9409.

[73] A. Gałecki, T. Burzykowski, A. Gałecki, T. Burzykowski, Linear mixed-effects model, Springer, 2013.

[74] A. Chmiel, E. Schubert, Back to the inverted-u for music preference: A review of the literature, Psychology of Music 45 (6) (2017) 886–909. arXiv:https://doi.org/10.1177/0305735617697507, doi:10.1177/0305735617697507.
URL https://doi.org/10.1177/0305735617697507

[75] M. W. Spratling, A review of predictive coding algorithms, Brain and Cognition 112 (2017) 92–97. doi:10.1016/j.bandc.2015.11.003.

[76] M. Heilbron, M. Chait, Great expectations: is there evidence for predictive coding in auditory cortex?, Neuroscience 389 (2018) 54–73. doi:10.1016/j.neuroscience.2017.07.061.

[77] K. S. Walsh, D. P. McGovern, A. Clark, R. G. O'Connell, Evaluating the neurophysiological evidence for predictive processing as a model of perception, Annals of the New York Academy of Sciences 1464 (2020) 242–268. doi:10.1111/nyas.14321.

Declaration of Interest Statement

Author contributions: G.M. designed research; G.M. performed research; G.M. and G.M.D.L. and B.G. contributed unpublished reagents/analytic tools; G.M., G.M.D.L., and B.G. analyzed data; G.M. wrote the first draft of the paper; G.M., G.M.D.L., B.G., and S.A.S. edited the paper.
We would like to thank Marcus Pearce (author of the first Lisp version of IDyOM) for the nice discussions and proofreading of the manuscript that greatly improved the quality of this manuscript.
This work was supported by Advanced ERC Grant NEUME 787836 and Air Force Office of Scientific Research
and National Science Foundation grants to S.A.S.; and FrontCog Grant ANR-17-EURE-0017, PSL Idex ANR-10-
IDEX-0001-02, and a PhD scholarship from the Research Chair on Beauty Studies PSL–L'Oréal to G.M.
The authors declare no competing financial interests.
The code for IDyOMpy is available at: https://github.com/GuiMarion/IDyOMpy
The code for the analysis is available at:
https://github.com/GuiMarion/codeForPaper-IDyOMpy-
Correspondence should be addressed to Guilhem Marion at gm3557@nyu.edu
Both EEG experiments were undertaken at Ecole Normale Supérieure in Paris, France following the Declaration of Helsinki and approved by the CERES committee of Paris Descartes University (CERES 2013-11); c.f. \cite{Marion2021} and \cite{DiLiberto2021} for more information. The behavioral experiment was conducted at McGill in Montreal, Canada under ethics approval of the university; c.f. \cite{gold2019predictability} for more information.