Determining Feature Relevance in Subject Responses to Musical Stimuli

Morwaread M. Farbood[1] and Bernd Schoner[2]

[1] New York University, New York, NY 10012, USA
[2] ThingMagic Inc., Cambridge, MA 02142, USA

Abstract. This paper presents a method that determines the relevance of a set of signals (musical features) given listener judgments of music in an experimental setting. Rather than using linear correlation methods, we allow for nonlinear relationships and multi-dimensional feature vectors. We first provide a methodology based on polynomial functions and the least-mean-square error measure. We then extend the methodology to arbitrary nonlinear function approximation techniques and introduce the Kullback-Leibler Distance as an alternative relevance metric. The method is demonstrated first with simple artificial data and then applied to analyze complex experimental data collected to examine the perception of musical tension.

1 Introduction

There are two generic types of responses that can be collected in an experimental setting where subjects are asked to make judgments on musical stimuli. The first is a retrospective response, where the listener only makes a judgment after hearing the musical excerpt; the second is a real-time response where judgments are made while listening. The latter has become increasingly popular among experimental psychologists as an effective means of collecting data. In particular, studies on musical tension have often employed real-time collection methods (Nielsen 1983; Madson and Fredrickson 1993; Krumhansl 1996; Bigand et al. 1996; Bigand & Parncutt 1999; Toiviainen & Krumhansl 2003; Lerdahl & Krumhansl 2007). The validity of this type of data collection is indicated by the high inter- and intra-subject correlation between subject responses and, more importantly, the indication that these responses correspond to identifiable musical structures (Toiviainen & Krumhansl 2003).

In this paper we propose a method to detect and quantify the relevance of individual features in complex musical stimuli where both the musical features describing the stimuli and the subject responses are real-valued. While the method can be used with most types of auditory or visual stimuli and most types of responses,[1] the method discussed here was developed for the purposes of understanding how musical structures affect listener responses to tension. Our analysis

[1] For example, the response signal can be brain activity, as measured by imaging technology (Schoner 2000), a general biological response such as skin conductivity (Picard et al. 2001), or direct subject input by means of a computer interface.

E. Chew, A. Childs, and C.-H. Chuan (Eds.): MCM 2009, CCIS 38, pp. 115–129, 2009.

is based on the assumption that perceived tension is a function of various salient musical parameters varying over time, such as harmony, pitch height, onset frequency, and loudness (Farbood 2006). It is the objective of this paper to formulate a mathematically sound approach to determine the relative importance of each individual feature to the perception of tension.

In the following sections, we will first provide a methodology based on polynomial functions and the least-mean-square error measure and then extend the methodology to arbitrary nonlinear function approximation techniques. We will first verify our approach with simple artificial data and then apply it to complex data from a study exploring the perception of musical tension.

2 Prior Work

In this paper we rely on prior art from two distinct fields: (A) the statistical evaluation of experimental and continuous data, mostly using variants of linear correlation and regression (Gershenfeld 1999b) and (B) feature selection for high-dimensional pattern recognition and function fitting in machine learning (Mitchell 1997).

(A) is helpful for our task at hand, but its limitation stems from the assumption of linearity. The importance of a feature is determined by the value of the correlation coefficient between a feature vector and a response signal: the closer the correlation value to 1 or to -1, the more important the feature. A variant of this approach—based on the same mathematical correlation—uses the coefficients in a linear regression model to indicate the relevance of a feature.

(B) offers a large amount of literature mostly motivated by high-dimensional, nonlinear machine-learning problems facing large data sets. Computational limitations make it necessary to reduce the dimensionality of the available feature set before applying a classifier algorithm or a function approximation algorithm. The list of common techniques includes Principle Component Analysis (PCA), which projects the feature space on the most relevant (linear) subset of components, and Independent Component Analysis (ICA), which is the nonlinear equivalent of PCA (Gershenfeld 1999b). Both PCA and ICA are designed to transform the feature set for the purpose of estimating the dependent signal, but they do not relate an individual feature to the dependent signal. In fact, most prior work in machine learning is focused on estimating the dependent signal, not the significance of individual features.

Prior art can also be found in the field of information theory. Koller & Sahami (1996) developed a methodology for feature selection in multivariate, supervised classification and pattern recognition. They select a subset of features using a subtractive approach, starting with the full feature set and successively removing features that can be fully replaced by a subset of the other features. Koller & Sahami use the information-theoretic cross-entropy, also known as KL-distance (Kullback & Leibler 1951) in their work.

3 Feature Relevance Measured by Polynomial Least-Mean Square Estimation

In this paper, we estimate the relevance of a particular musical feature x_i by computing the error between the actual subject response signal y and the estimation $\hat{y}$ of the same. We first build a model based on the complete feature set F and derive the least-mean-square error E from $\hat{y}$ and y. We then build models for each of the feature sets F_i, where F_i includes all the features except x_i, and compute the errors E_i based on $\hat{y}_i$ and y. We define the Relevance Ratio $R_i = E/E_i$ and postulate that R_i is a strong indicator of the relevance of x_i for y.

We start by selecting an appropriate model to estimate $\hat{y}$, keeping in mind our goal of overcoming the linearity constraint of common linear techniques. We consider nonlinear function fitting techniques for the underlying estimation framework, and observe that such techniques can be classified into two major categories: linear coefficient models (discussed in this section) and nonlinear models (discussed in the next section). Linear coefficient models and generalized linear models use a sum over arbitrary nonlinear basis functions $f_k(\mathbf{x})$ weighted by linear coefficients a_k,

$$y(\mathbf{x}) = \sum_{k=1}^{K} a_k \; f_k(\mathbf{x}). \tag{1}$$

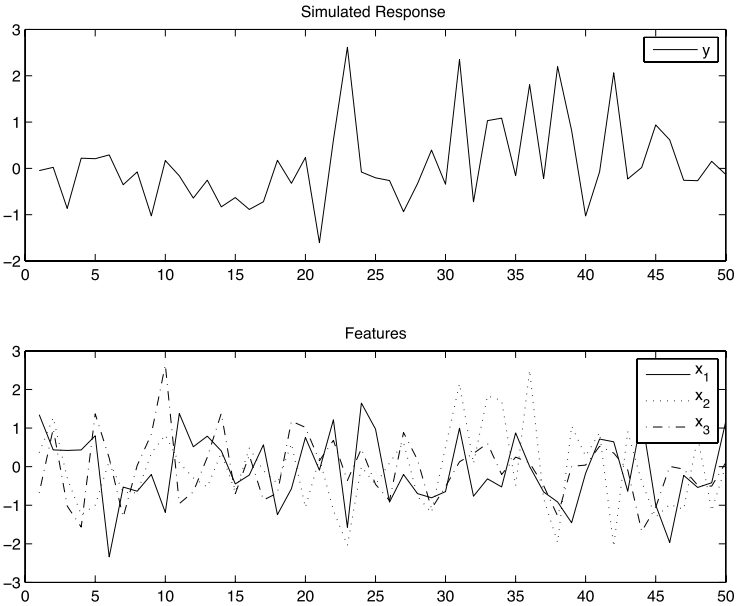
A prominent example of this architecture is the class of polynomial models, which takes the form

$$f(\mathbf{x}) = a_0 + \sum_{m=1}^{M} a_m \Psi_m(\mathbf{x}), \text{with} \tag{2}$$
$$\Psi_m(\mathbf{x}) = \prod_i x_i^{e_{i,m}}.$$

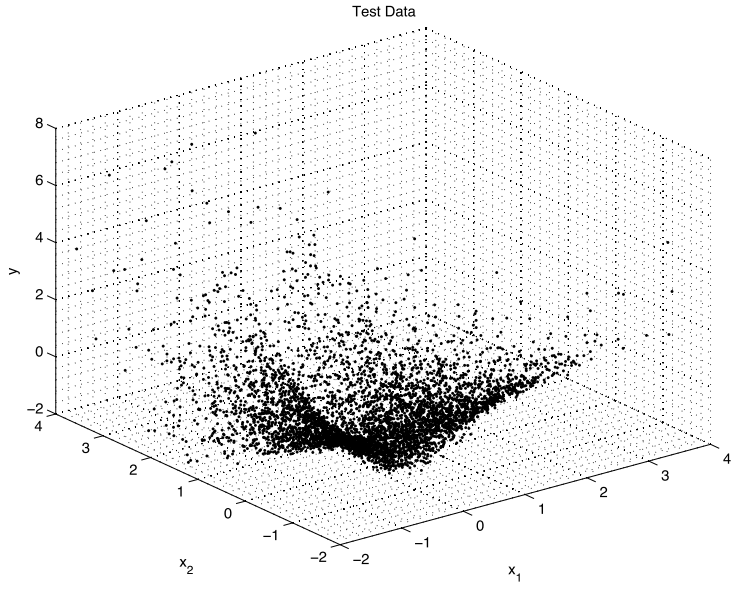
M denotes the number of basis functions and $e_{i,m}$ depends on the order of polynomial approximation. For example, a two-dimensional quadratic model includes a total of $M = 5$ basis functions: (x_1), (x_2), $({x_1}^2)$, $(x_1 x_2)$ and $({x_2}^2)$. The parameters in this model are typically estimated in a least-mean-square fit over the experimental data set, which is computationally inexpensive for small to medium dimensional feature sets (Gershenfeld 1999b). Using the model we compute $\hat{y} = f(x)$ for all data points $(\mathbf{x}_n, y_n)$, and subsequently derive $E = \sum_N (\hat{y}_n - y_n)^2/N$.

It is a well-known fact that we can cause the error E to shrink to an arbitrarily small value by adding more and more resources to the model—that is, by increasing the number of parameters and basis functions. However, in doing so we are likely to model noise rather than the underlying causal data structure. In order to avoid this problem, we cross-validate our model and introduce a global regularizer that constrains our model to the "right size."

We divide the available data into two data sets. The training data set $(\mathbf{x}, y)_{tr}$ is used to optimize the parameters of the model, whereas the test data set $(\mathbf{x}, y)_{test}$



(a)



(b)

Fig. 1. (a) 1-D plot of features x_1, x_2, x_3, and function y_B and (b) 3-D plot of function y_B (4)

is used to validate the model using E_{test}. As we slowly increase the number of model parameters, we find that the test data estimation error E_{test} decreases initially, but starts to increase as soon as the extra model parameters follow the randomness in the training data. We declare that the model resulting in the smallest estimation error

$$E_m = \sum_{N_{test}} (\hat{y}_{n,m} - y_n)^2 / N_{test} \tag{3}$$

represents the best model architecture for the data set at hand.

Given these considerations, we can now provide a step-by-step algorithm to determine the Relevance Ratio R_i:

1. Divide the available experimental data into the training set $(\mathbf{x}, y)_{tr}$ and $(\mathbf{x}, y)_{test}$. $(\mathbf{x}, y)_{test}$ typically represents $10\% - 30\%$ of the data. If the amount of data set is very limited more sophisticated bootstrapping techniques can be applied (Efron 1983).
2. Build a series of models m based on the complete feature set F, slowly increasing the complexity of the model, i.e. increasing the polynomial order.
3. For each model m compute the error $E_m = \sum_{N_{test}} (\hat{y}_m - y)^2 / N_{test}$. Choose the model architecture m that results in the smallest E_m. Next, build models m_i for all sets $(\mathbf{x}_i, y)$, where the vector $\mathbf{x}_i$ (F_i) includes all features F, except for x_i.
4. Compute $E_i = \sum_{N_{test}} (\hat{y}_i - y)^2 / N_{test}$ for all feature sets F_i and derive the Relevance Ratio $R_i = E_m / E_i$ for all features x_i.

$R_i = 1$ indicates that a feature x_i is irrelevant for the response y. A value of R_i close to 1 indicates little relevance whereas a small value of R_i indicates a high level of relevance. R_i is dimensionless.

Table 1. Application of the polynomial estimator to functions y_A and y_B (4): (a) indicates the error for the different model m based on $\mathbf{x}$; (b) and (c) indicate the resulting Relevance Ratios for features x_1, x_2, and x_3

(a)

Function		**Polynomial Order** 1	2	3	4	5	6
A	Training Set Error	0.8960	0.0398	0.0398	0.0397	0.0395	0.0392
	Test Set Error	0.8938	0.0396	0.0396	0.0398	0.0399	0.0413
B	Training Set Error	0.9989	0.1123	0.1121	0.0740	0.0728	0.0546
	Test Set Error	1.0311	0.1204	0.1210	0.0848	0.0898	0.0924

(b)

Function A	Feature Set F_1	F_2	F_3
Error Training Set	0.8960	0.1452	0.0398
Error Test Set	0.8938	0.1480	0.0396
	x_1	x_2	x_3
Relevance Ratio	0.0443	0.2674	0.9995

(c)

Function B	Feature Set F_1	F_2	F_3
Error Training Set	0.2328	0.8406	0.0745
Error Test Set	0.2478	0.8590	0.0842
	x_1	x_2	x_3
Relevance Ratio	0.3423	0.0988	1.0072

Before we use the method on experimental data, we demonstrate it here on two artificial data sets: A three dimensional set of 5,000 feature data is generated using $x_i = 10 \cdot N(0,1)$, where $N(\mu, \sigma)$ denotes the normal distribution. We define functions A and B as

$$\begin{aligned} y_A &= x_1^2 + 5 \cdot x_2 + 0 \cdot x_3 + 30 \cdot N(0,1) \\ y_B &= 100 \cdot log(x_1) + x_2^2 + 0 \cdot x_3 + 20 \cdot N(0,1) \end{aligned} \tag{4}$$

Figure 1 shows a one-dimensional and three-dimensional plot of $\mathbf{x}$ and y_B.

Applying our algorithm we obtain the results indicated in Table 1. In the case of function A it can be seen that the polynomial model correctly determines that the data is drawn from a second-order model. For both function A and B, the model correctly assigns a value of $R_3 = 1$ indicating that x_3 was not used to generate y as is indeed the case.

4 Extension to General Nonlinear Estimators and Probabilistic Models

Polynomial models and generalized linear models have many nice properties, including the fact that parameter sets are easily understood. The drawback of these models is that the number of basis terms increases exponentially with the dimensionality of $\mathbf{x}$, making them computationally prohibitive for high-dimensional data sets.

The second category of nonlinear models uses variable coefficients inside the nonlinear basis functions

$$y(\mathbf{x}) = \sum_{k=1}^{K} f(\mathbf{x}, \mathbf{a}_k). \tag{5}$$

The most prominent examples of this class of models are artificial neural networks, graphical networks, and Gaussian mixture models (GMM). The models are exponentially more powerful, but training requires an iterative nonlinear search. Here we demonstrate the methodology with GMM's which, as a subclass of Bayesian networks, have the added benefit of being designed on probabilistic principles.

GMM's are derived as the joint probability density $p(\mathbf{x}, y)$ over a set of data $(\mathbf{x}, y)$. $p(\mathbf{x}, y)$ is expanded as a weighted sum of Gaussian basis terms and hence takes on the form

$$p(y, \mathbf{x}) = \sum_{m=1}^{M} p(y, \mathbf{x}, c_m) \tag{6}$$

$$= \sum_{m=1}^{M} p(y|\mathbf{x}, c_m) p(\mathbf{x}|c_m) p(c_m) \quad . \tag{7}$$

Table 2. Application of the GMM estimator to functions y_A and y_B (4): (a) indicates the error for the different model m based on $\mathbf{x}$; (b) and (c) indicate the resulting Relevance Ratios for features x_1, x_2, and x_3

(a)

Function		Number of Clusters 2	4	6	8	10	12	14	16	18	20
A	Training Set error	0.414	0.056	0.044	0.043	0.041	0.040	0.041	0.041	0.041	0.040
	Test Set Error	0.413	0.056	0.046	0.044	0.041	0.041	0.042	0.042	0.043	0.041
B	Training Set Error	0.343	0.151	0.095	0.075	0.052	0.044	0.040	0.035	0.033	0.028
	Test Set Error	0.362	0.161	0.106	0.081	0.056	0.050	0.046	0.041	0.038	0.033
		22	24	26	28	30	32	34	36	38	40
A	Training Set error	0.040	0.040	0.039	0.039	0.039	0.039	0.039	0.039	0.039	0.039
	Test Set Error	0.043	0.042	0.041	0.041	0.041	0.041	0.041	0.043	0.042	0.041
B	Training Set Error	0.027	0.029	0.024	0.025	0.024	0.022	0.021	0.024	0.024	0.021
	Test Set Error	0.029	0.035	0.027	0.030	0.028	0.024	0.026	0.029	0.029	0.026

(b)

Function A	Feature Set F_1	F_2	F_3
Error Training Set	0.8950	0.1464	0.0403
Error Test Set	0.8958	0.1515	0.0408
	x_1	x_2	x_3
Relevance Ratio	0.0453	0.2676	0.9928

(c)

Function B	Feature Set F_1	F_2	F_3
Error Training Set	0.2502	0.7630	0.0209
Error Test Set	0.2575	0.8646	0.0233
	x_1	x_2	x_3
Relevance Ratio	0.0912	0.0272	1.0096

We choose

$$p(\mathbf{x}|c_k) = \frac{|\mathbf{P}_k^{-1}|^{1/2}}{(2\pi)^{D/2}} e^{-(\mathbf{x}-\mathbf{m}_k)^T \cdot \mathbf{P}_k^{-1} \cdot (\mathbf{x}-\mathbf{m}_k)/2} \quad , \tag{8}$$

where $\mathbf{P}_k$ is the weighted covariance matrix in the feature space. The output distribution is chosen to be

$$p(\mathbf{y}|\mathbf{x}, c_k) = \frac{|\mathbf{P}_{k,y}^{-1}|^{1/2}}{(2\pi)^{D_y/2}} e^{-(\mathbf{y}-\mathbf{f}(\mathbf{x},\mathbf{a}_k))^T \cdot \mathbf{P}_{k,y}^{-1} \cdot (\mathbf{y}-\mathbf{f}(\mathbf{x},\mathbf{a}_k))/2} \quad , \tag{9}$$

where the mean value of the output Gaussian is replaced by the function $\mathbf{f}(\mathbf{x}, \mathbf{a}_k)$ with unknown parameters $\mathbf{a}_k$.

From this we derive the conditional probability of y given $\mathbf{x}$

$$\begin{aligned} \langle \mathbf{y}|\mathbf{x} \rangle &= \int \mathbf{y}\ p(\mathbf{y}|\mathbf{x})\ d\mathbf{y} \\ &= \frac{\sum_{k=1}^{K} \mathbf{f}(\mathbf{x}, \mathbf{a}_k)\ p(\mathbf{x}|c_k)\ p(c_k)}{\sum_{k=1}^{K} p(\mathbf{x}|c_k)\ p(c_k)} \quad , \end{aligned} \tag{10}$$

which serves as our estimator of $\hat{y}$. The model is trained using the well-known Expectation-Maximization algorithm.

The number of Gaussian basis functions and the complexity of the local models serve as our global regularizers, resulting in the following step-by-step algorithm analogous to the polynomial case discussed before:

1. Divide the data into training set $(\mathbf{x}, y)_{tr}$ and test set $(\mathbf{x}, y)_{test}$.
2. Build a series of models m based on the complete feature set F, slowly increasing the number of Gaussian basis functions.
3. For each model m compute the error $E_m = (\hat{y}_m - y)^2/N_{test}$. Choose the model architecture m that results in the smallest E_m. Build models m_i for all sets $(\mathbf{x}_i, y)$.
4. Compute $E_i = \sum_{N_{test}} (\hat{y}_i - y)^2/N_{test}$ for all feature sets F_i and derive the Relevance Ratio $R_i = E_m/E_i$ for all features x_i.

Applying this new approach to our artificial data sets from before (4), we obtain the results in Table 2.

5 Kullback-Leibler Distance

The linear least-mean-square error metric is without doubt the most commonly used practical error metric, however, other choices can be equally valid. The framework of the Gaussian mixture model allows for the introduction of a probabilistic metric, known as the cross entropy or Kulback-Leibler distance (KL-Distance) (Kullback & Leibler 1951). The KL-Distance measures the divergence between two probability distributions $P(x)$ and $Q(x)$:

$$D_{KL}(P||Q) = \int_x P(x) log \frac{P(x)}{Q(x)} dx \tag{11}$$

where $P(x)$ is typically assumed to be the "true" distribution, and D_{KL} is a measure for how much $Q(x)$ deviates from the true distribution.

For our task at hand we are interested in how much the distribution $p(y|\mathbf{x}_i)$ deviates from $p(y|\mathbf{x}^*)$, where once again $\mathbf{x}_i$ includes all the elements of $\mathbf{x}$ except for x_i. This leads us to the definition

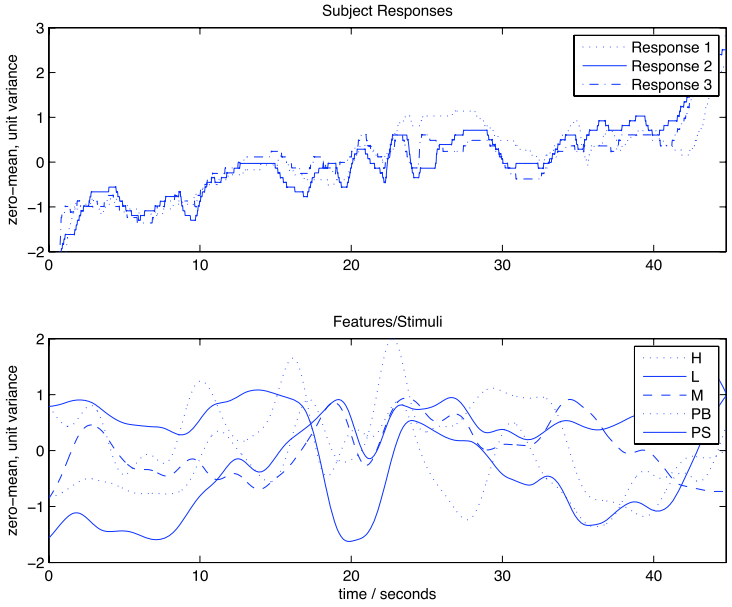
$$D_{KL}(p||p_i) = \int_{\mathbf{x},y} p(\mathbf{x}, y) log \frac{p(y|\mathbf{x})}{p(y|\mathbf{x}_i)} d\mathbf{x} dy \tag{12}$$

and given our definitions above, we obtain

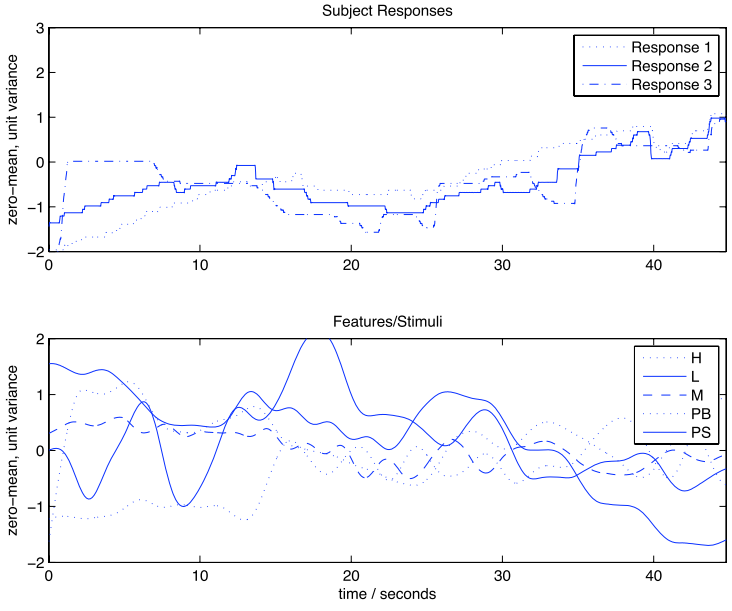
$$\begin{aligned} D_{KL}(p||p_i) &= \int_{\mathbf{x},y} p(\mathbf{x}, y)[log(p(y|\mathbf{x}) - log(p(y|\mathbf{x}_i))] d\mathbf{x} dy \\ &\approx \frac{1}{N} \sum_{n=1}^{N} [log(p(y_n|\mathbf{x}_n) - log(p(y_n|\mathbf{x}_{i,n}))] \quad , \end{aligned} \tag{13}$$

Here we replaced the integral over the density with the sum over the observed data (which itself is assumed to be drawn from the density).

To compute $D_{KL}(p||p_i)$ we need to first estimate $p(y_n|\mathbf{x}_{i,n})$. However, this step consists of estimating the local model parameters only, a relatively minor task. All other parameters needed to numerically evaluate this equation are already part of the model built in the first place.



(a)



(b)

Fig. 2. Features x_i and three subject responses (same subject) for (a) the Brahms excerpt (Fig. 5) and (b) the Bach-Vivaldi excerpt (Fig. 4). H = harmony, L = loudness, M = melodic expectation, PB = pitch height of bass line, PS = pitch height of soprano line.

6 Experimental Results

6.1 Data Set

Data was collected in an experiment that recorded real-time, continuous responses to musical stimuli. Thirty-five subjects, drawn from the faculty and student body at MIT, participated in the experiment. Subjects were asked to move a slider on a computer interface to indicate how they felt tension was changing in the music. They were instructed to raise the slider if they felt a general feeling of musical tension increasing, and to lower it when they felt it lessening. Each musical excerpt was played four times; after each iteration, subjects were asked to rate the confidence level of their response on a scale of 1 to 5. Slider positions were sampled at 50Hz.

Ten musical examples were used as stimuli in the experiment. Six of these examples were short (under 10 seconds) and composed specifically for the study. They featured simple and clear changes in tempo, onset frequency, loudness, harmony, and pitch contour. In addition, there were four excerpts taken from the classical repertoire: Schoenberg Klavierstück, Op. 11 No. 12, Beethoven Symphony No. 1 (Fig. 3), J. S. Bach's organ transcription of Vivaldi's D Major concerto (RV 208) (Fig. 4), and Brahms Piano Concerto No. 2 (Fig. 5). The longer examples were 20 seconds to one minute in length and considerably more complex than any of the short examples.

Musical parameters included in the feature set were harmonic tension, melodic expectation, pitch height of soprano and bass lines, onset frequency, and loudness. Not all features were relevant to all musical examples from the experiment. For the purposes of quantifying harmonic tension and melodic expectation, Lerdahl's (2001)[2] and Margulis's (2005) models were applied respectively.

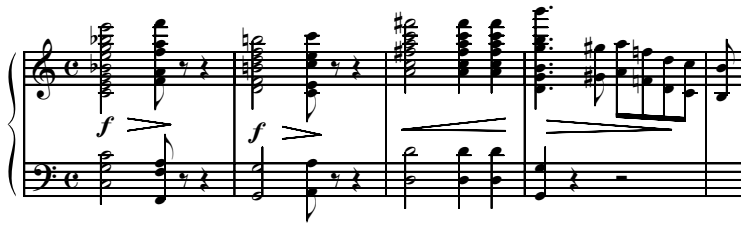


Fig. 3. Score of Beethoven excerpt

6.2 Results

The key results for all of the complex tonal examples are represented in Table 3. We use both the polynomial models and GMMs and apply our method to various subsets of the feature space. The results are largely robust against variations in

[2] Without the melodic attraction component; this factor is taken into account separately with Margulis's model.

Table 3. Summary of experimental results for the musical tension study. For each experiment we indicate the type of estimation (polynomial or GMM), the global regularizer (polynomial order or number of Gaussians) and the Relevance Ratio of each feature: H = harmony, L = loudness, M = melodic expectation, O = onset frequency, PB = pitch height of bass line, PS = pitch height of soprano line.

Brahms							
Type	POLY	**Relevance Ratio**					
Polynomial order	3	H	L	M	O	PB	PS
Num. Gaussians	N/A	1.0166	1.0099	1.0306	1.0247	1.0251	0.9571
Type	POLY	**Relevance Ratio**					
Polynomial order	3	H	L	M	PB	PS	
Num. Gaussians	N/A	0.8869	0.6133	0.8527	1.0099	0.8366	
Type	POLY	**Relevance Ratio**					
Polynomial order	4	H	L	PB	PS		
Num. Gaussians	N/A	0.8460	0.4795	0.6787	0.6367		
Type	POLY	**Relevance Ratio**					
Polynomial order	4	H	L	M			
Num. Gaussians	N/A	0.8623	0.3228	0.5750			
Type	GMM	**Relevance Ratio**					
Polynomial order	N/A	H	L	M	PB	PS	
Num. Gaussians	16	0.7230	0.2953	0.6583	0.7478	0.9509	
Bach-Vivaldi							
Type	POLY	**Relevance Ratio**					
Polynomial order	2	H	L	M	O	PB	PS
Num. Gaussians	N/A	0.6950	1.1549	0.9703	0.7653	0.8047	0.9472
Type	POLY	**Relevance Ratio**					
Polynomial order	3	H	L	M	PB	PS	
Num. Gaussians	N/A	0.7413	1.1131	0.9780	1.0115	0.9696	
Type	POLY	**Relevance Ratio**					
Polynomial order	3	H	M	PB	PS		
Num. Gaussians	N/A	0.6436	0.8953	0.9265	0.8112		
Type	POLY	**Relevance Ratio**					
Polynomial order	3	H	L	M	PS		
Num. Gaussians	N/A	0.7514	1.0195	0.8625	0.8717		
Type	GMM	**Relevance Ratio**					
Polynomial order	N/A	H	L	M	PS		
Num. Gaussians	N/A	0.6667	1.0439	0.9362	0.8945		
Beethoven							
Type	POLY	**Relevance Ratio**					
Polynomial order	2	H	L	M	O	PB	PS
Num. Gaussians	N/A	1.0575	0.9699	0.9689	0.9644	0.9375	1.0822
Type	POLY	**Relevance Ratio**					
Polynomial order	2	H	L	M	PB	PS	
Num. Gaussians	N/A	1.0607	0.9749	1.0604	1.0580	1.0252	
Type	POLY	**Relevance Ratio**					
Polynomial order	2	H	L	PB	PS		
Num. Gaussians	N/A	0.9502	0.4230	1.0448	1.0289		
Type	GMM	**Relevance Ratio**					
Polynomial order	N/A	H	L	PB	PS		
Num. Gaussians	4	1.2435	0.4087	0.8488	1.1299		

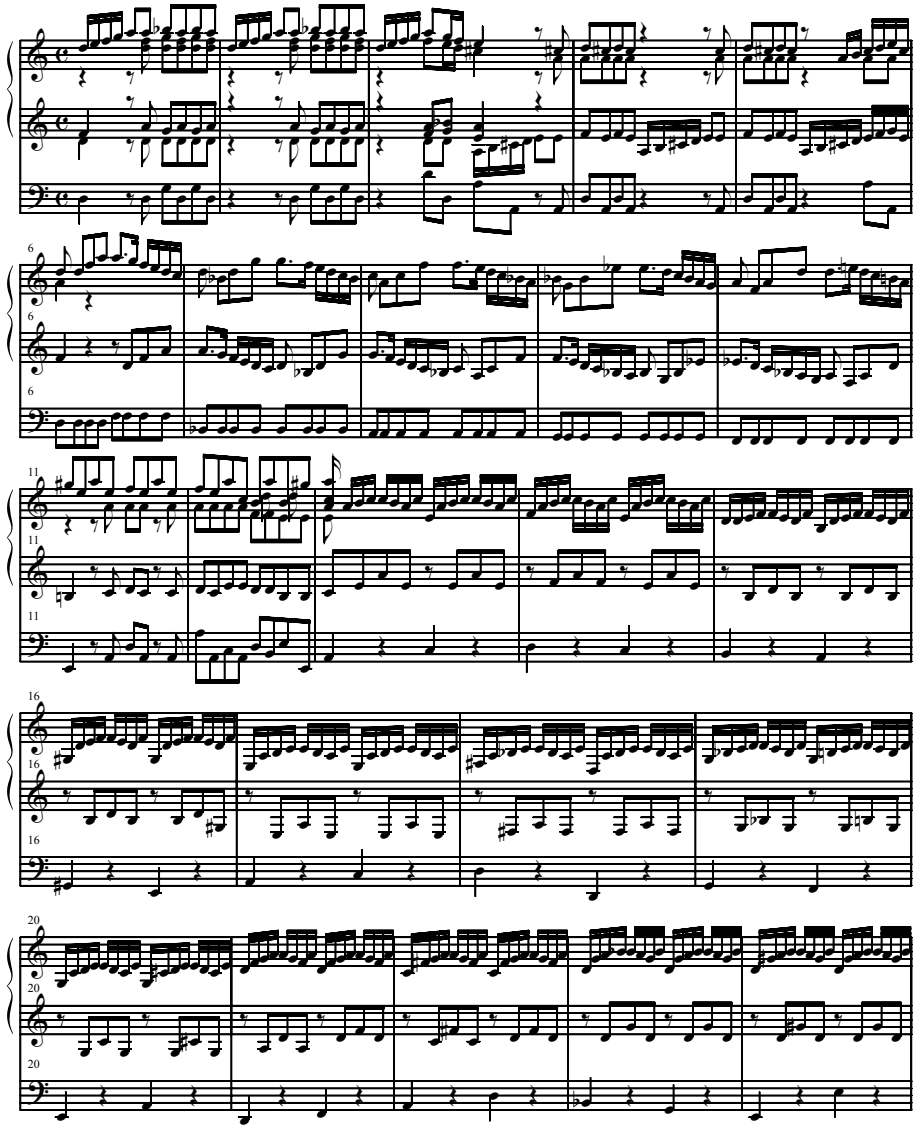


Fig. 4. Score of Bach-Vivaldi excerpt

the model architecture. Relevance is always rewarded with a Relevance Ratio significantly smaller than 1. However, the relative Ratio between features can vary from model to model.

We observe that the model performs best with a modest number of features. The fewer the available feature dimensions, the cleaner the results. We therefore start with a larger feature set and successively remove the least relevant features from the set until the model provides a robust estimate of the feature relevance.

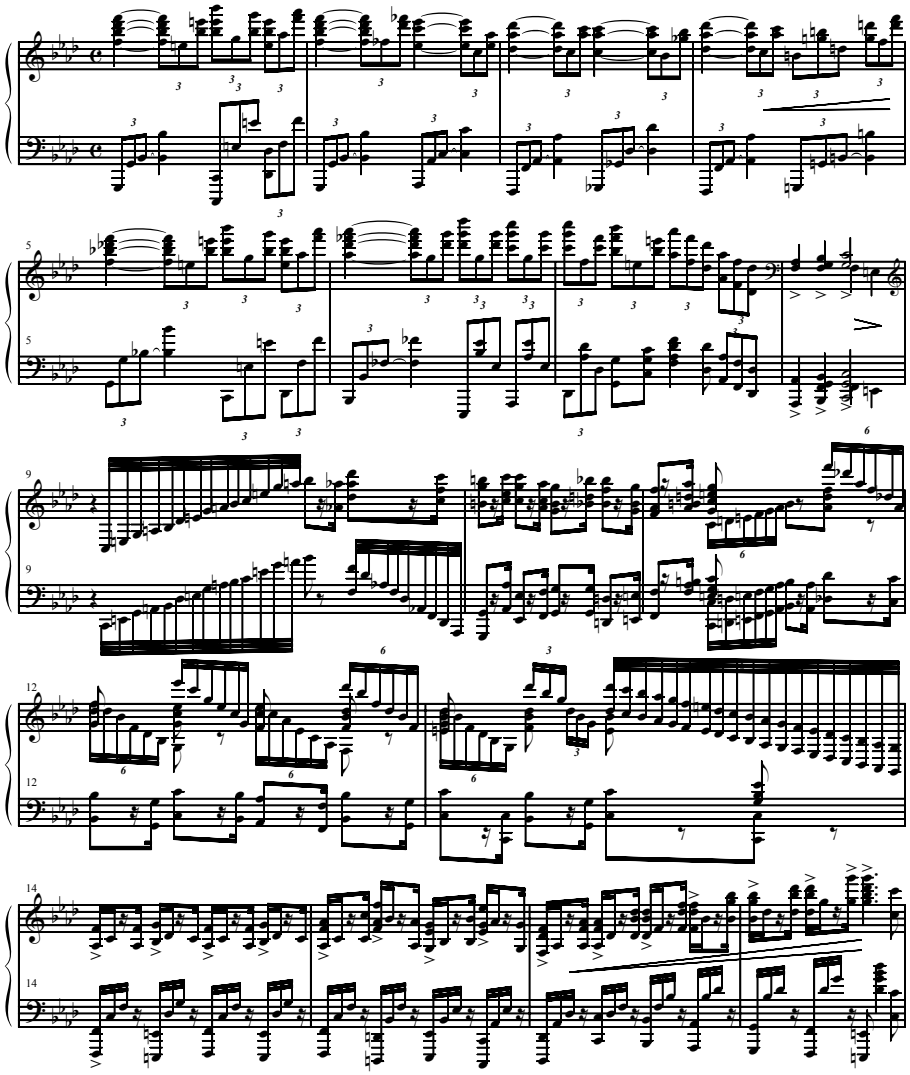


Fig. 5. Score of Brahms excerpt

Mathematically, this phenomenon can be explained by the fact that the features are not statistically independent and that the relevance of one feature may be entirely assumed by an other feature (or a set of features) (Koller & Sahami 1996).

We observe in the case of the Brahms excerpt that loudness is clearly the predominant feature and hence has the smallest Relevance Ratio. In the case of the Bach-Vivaldi excerpt, harmony is primarily responsible for perceived tension. In the Beethoven excerpt, like the Brahms, loudness has the most impact on the response. This makes qualitative sense, as there are no clear changes in the

dynamics for the Bach-Vivaldi example, unlike the case for the Brahms and Beethoven, where change in loudness is a salient feature.

The Relevance Ratio confirms that listeners relate salient changes in musical parameters to changes in tension. While there are multiple factors that contribute to how tension is perceived at any given moment, one particular feature may predominate, depending on the context. The Relevance Ratio reveals the overall prominence of each feature in the subject responses throughout the course of a given excerpt. While it could be argued that listeners respond more strongly to certain features (e.g. loudness over onset frequency), it is the *degree of change* in each parameter that corresponds most strongly to tension, regardless of whether the feature is purely musical, as in the case of harmony and melodic contour, or expressive, as in the case of tempo and dynamics.

Summary

We have introduced an new estimator called the Relevance Ratio that is derived from arbitrary nonlinear function approximation techniques and the least-mean-square error metric. To demonstrate the functionality of the Relevance Ratio, it was first applied to a set of artificial test functions where the estimator correctly identified relevant features. In a second step the estimator was applied against a data set of experimental subject responses where we gained valuable insights into the relevance of certain salient features for perceived musical tension. Additionally, we introduced the KL-Distance as an alternative estimator defined in purely probabilistic terms.

References

Bigand, E., Parncutt, R., Lerdahl, F.: Perception of musical tension in short chord sequences: The influence of harmonic function, sensory dissonance, horizontal motion, and musical training. Perception & Psychophysics 58, 125–141 (1996)

Bigand, E., Parncutt, R.: Perceiving music tension in long chord sequences. Psychological Research 62, 237–254 (1999)

Efron, B.: Estimating the Error Rate of a Prediction Rule: Improvements on Cross-Validation. Journal of the American statistical Asociation 78, 316–331 (1983)

Farbood, M.: A Quantitative, Parametric Model of Musical Tension. Ph.D. Thesis, Massachusetts Institute of Technology (2006)

Gershenfeld, N.: The Nature of Mathematical Modeling. Cambridge University Press, New York (1999)

Gershenfeld, N., Schoner, B., Metois, E.: Cluster-Weighted Modeling for Time Series Analysis. Nature 379, 329–332 (1999)

Koller, D., Sahami, M.: Toward Optimal Feature Selection. In: Proceedings of the Thirteenth International Conference on Machine Learning, pp. 284–292 (1996)

Krumhansl, C.L.: A perceptual analysis of Mozart's Piano Sonata K. 282: Segmentation, tension, and musical ideas. Music Perception 13, 401–432

Kullback, S., Leibler, R.A.: On Information and sufficiency. Annals of Mathematical statistics 22, 76–86 (1951)

Lerdahl, F.: Tonal Pitch Space. Oxford University Press, New York (2001)
Lerdahl, F., Krumhansl, C.L.: Modeling Tonal Tension. Music Perception 24, 329–366 (2007)
Madson, C.K., Fredrickson, W.E.: The experience of musical tension: A replication of Nielsen's research using the continuous response digital interface. Journal of Music Therapy 30, 46–63 (1993)
Margulis, E.H.: A Model of Melodic Expectation. Music Perception 22, 663–714 (2005)
Mitchell, T.: Machine Learning. McGraw-Hill, New York (1997)
Nielsen, F.V.: Oplevelse of Musikalsk Spaending. Akademisk Forlag, Copenhagen (1983)
Picard, R.W., Vyzas, E., Healey, J.: Toward Machine Emotional Intelligence: Analysis of Affective Physiological State. IEEE Transactions Pattern Analysis and Machine Intelligence 23(10), 1175–1191 (2001)
Schoner, B.: Probabilistic Characterization an Synthesis of Complex Driven Systems. Ph.D. Thesis, Massachusetts Institute of Technology (2000)
Toiviainen, P., Krumhansl, C.L.: Measuring and modeling real-time responses to music: The dynamics of tonality induction. Perception 32, 741–766 (2003)