*Fundamenta Informaticae 176 (2020) 183–203*
*DOI 10.3233/FI-2020-1970*
*IOS Press*



# Structured Gaussian Process Regression of Music Mood

**Santosh Chapaneri***
*Dept. of Electronics and Telecommunication Engineering*
*St. Francis Institute of Technology, University of Mumbai, India*
*santoshchapaneri@sfit.ac.in*

**Deepak Jayaswal**
*Dept. of Electronics and Telecommunication Engineering*
*St. Francis Institute of Technology, University of Mumbai, India*
*djjayaswal@sfit.ac.in*

**Abstract.** Modeling the music mood has wide applications in music categorization, retrieval, and recommendation systems; however, it is challenging to computationally model the affective content of music due to its subjective nature. In this work, a structured regression framework is proposed to model the valence and arousal mood dimensions of music using a single regression model at a linear computational cost. To tackle the subjectivity phenomena, a confidence-interval based estimated consensus is computed by modeling the behavior of various annotators (e.g. biased, adversarial) and is shown to perform better than using the average annotation values. For a compact feature representation of music clips, variational Bayesian inference is used to learn the Gaussian mixture model representation of acoustic features and chord-related features are used to improve the valence estimation by probing the chord progressions between chroma frames. The dimensionality of features is further reduced using an adaptive version of kernel PCA. Using an efficient implementation of twin Gaussian process for structured regression, the proposed work achieves a significant improvement in $R^2$ for arousal and valence dimensions relative to state-of-the-art techniques on two benchmark datasets for music mood estimation.

**Keywords:** Music mood, Structured regression, Crowdsourced annotations

*Address for correspondence: Dept. of Electronics and Telecommunication Engineering, St. Francis Institute of Technology, Mount Poinsur, SVP Road, Borivali (West), Mumbai 400103, India

# 1. Introduction

Music has the ability to convey emotions to listeners and is often referred to as the language of emotions. Although advancements have been done in the field of music and emotion[1] research, mood estimation still remains a challenging problem. Mood as a music descriptor is frequently used as a social tag and can be used for organizing large music collection on various smart devices thus requiring modern user interfaces for intuitive music selection and automatic playlist generation. However, the perception of music mood is difficult to quantify due to its subjective nature. Mood can be modelled using the categorical approach (sad, happy, surprise, etc.) or the dimensional approach (typically using valence and arousal dimensions). Using the dimensional approach is beneficial since the categorical approach cannot exhaustively cover the wide range of moods [1, 2]. Music mood estimation and retrieval is a popular service in several commercial apps (e.g. Musicovery, Spotify, Wynk, etc.) that suggests an automated playlist based on the current mood query of the user. Improving the mood estimation of a particular audio clip and enhanced recommendations are among the major focus areas of the digital music industry.

Several benchmark datasets have been developed for dimensional music mood estimation using crowdsourced annotations to handle the subjectivity issue, where multiple annotators are asked to provide their opinions on the valence and arousal values for specific music clips. The perceived mood is studied rather than the felt mood as this alleviates the burden of several physiological factors that come into play for a layman listener [3]. The labeling of music mood can be inconsistent due to various personal and situational aspects such as personality, context, cultural background, etc. A typical approach is to estimate the ground truth of each music clip by averaging multiple annotations given to it. This assumes that all annotators are equally reliable which may not be a valid assumption in practice [4], since it often ignores the annotator errors (e.g. low-attention) and outliers (e.g. adversarial behavior) that can have a significant impact on the consensus. Thus, it is important to model the behavior and obtain the reliability of each annotator and consider this factor to determine the estimated consensus.

Structured prediction is an actively studied topic due to its prevalence in real-world applications (e.g. human pose estimation, monocular depth estimation, music mood estimation, etc.) where the target variables are inter-dependent. It is worthwhile to note that the two dimensions of valence and arousal are not completely uncorrelated [5], hence a structured prediction framework such as [6] is needed to model the affective content of music mood. Another issue is designing an appropriate feature representation to model the music mood. While conventional acoustic features can be used, they are often computed at the frame-level for each music clip thus yielding high feature dimensionality. Among the two dimensions of music mood, valence estimation is quite challenging and in this work, chord related features are exploited by probing the progression of chords between successive frames.

This paper is an extended version of the work published in [7]. We extend our previous work by (i) modeling the various behaviors of annotators to determine the estimated consensus from crowdsourced annotations, (ii) modifying the chord progression histogram to improve the valence estimation, and

[1]The terminology of mood and emotion is used interchangeably by the MIR (Music Information Retrieval) community, although there is a subtle difference between the two from the psychology perspective.

(iii) evaluating the performance of proposed work on two benchmark datasets for music mood estimation. The rest of this paper is organized as follows: Section 2 discusses the related work in literature for music mood estimation and Section 3 explains the proposed methodology. Section 4 presents the experimental validation followed by conclusions in Section 5.

**Contributions** The contributions of this work are four-fold:

(a) An EM algorithm is proposed for multi-dimensional continuous output to determine the estimated consensus from crowdsourced mood annotations by learning the behavior of annotators.
(b) Feature representation of music clips is computed using the variational Bayesian acoustic Gaussian mixture model and a modified version of chord progression histogram to improve the valence estimation.
(c) A *single* computationally efficient twin Gaussian process regressor is used for structured regression of two-dimensional music mood estimation.
(d) The performance of proposed work is evaluated on two benchmark datasets for music mood estimation: AMG and DEAM.

## 2. Related work

The task of music mood estimation is an active research topic [4, 5, 7–12] due to its predominant difficulty in accurately characterizing valence and a comprehensive review of this task is presented in [13]. Music mood being a highly subjective phenomenon, multiple annotations per each music clip are required through crowdsourcing to capture the variability of responses. An often overlooked (and less reliable) assumption in most supervised learning techniques for music mood estimation is that the average valence and arousal values of multiple annotator responses serve as the ground truth. This leads to the problem of truth discovery analysis for finding the estimated consensus among various annotators [4, 14–20]. Raykar *et al*. [14, 15] proposed methods to learn the annotator behaviors for crowdsourced labeling (classification) tasks. In [16], a detailed review of various judgement analysis techniques is presented to determine the crowdsourced consensus considering problem difficulty, spammer identification, constrained judgement, etc. Another survey on truth discovery analysis methods is presented in [17] focusing on crowdsourced opinions of multi-source data from categorical as well as continuous domains. In [18], an uncertainty-aware modeling approach is proposed to estimate the kernel density from multiple sources and learn trustworthy opinions. In [19], the joint distribution of annotators is considered to learn the estimated consensus. A non-parametric Gaussian process model is proposed in [20] to learn the regression function as well as the behavior of annotators.

An appropriate prototype is needed for feature representation to model the acoustic features by avoiding the curse of dimensionality. Kernel density estimation is used in [4] to represent the VA (valence-arousal) space as a probability density function (PDF) and an audio space dictionary is learned to map the acoustic features. Rhythmic and melodic features are proposed in [5] using musical concepts for the purpose of music emotion classification. An acoustic Gaussian mixture model (GMM) is proposed in [9] to represent the acoustic features as a posterior probability feature vector. In

the preliminary version of this paper [7], novel features based on acoustic GMM using Bayesian inference are proposed by automatically determining the number of latent audio topics (mixtures) without risking over-fitting. Histogram density modeling approach is used in [10] to represent the VA space as a heatmap and the block-level GMM posterior probability feature vectors are mapped to the latent histograms. In [21], the frame-level features are stacked by computing their statistics over multiple windows. Feature selection techniques are used in [22] to select appropriate features using shrinkage methods for the emotion classification task. In [23], various acoustic features are used for mood estimation across culturally different music clips to determine their generalizability and it was shown that the support vector regression (SVR) model resulted in a performance generalizable to cross-cultural (Western and Chinese music) datasets. Recently, deep learning architectures, specifically convolutional neural networks (CNN), are also proposed in [24,25]; however, they solve the problem of music emotion classification instead of regression. Several chord estimation techniques are proposed in the literature for music summarization and retrieval applications: chromagram feature representation is used for harmonic analysis of music in [26], chord histogram is used for music classification in [27], chord progression histogram is proposed in [28] using multi-probing to improve the music retrieval performance.

The commonly used regression models such as Support Vector Regression (SVR), Adaboost.RT, and Gaussian Process Regression (GPR) do not handle multi-variate responses, thus 2 different models need to be trained for VA responses. GPR [29] has been shown to outperform SVR for music emotion recognition task [21] due to its capability of hyper-parameter learning and also providing uncertainties in output prediction resulting in a soft decision which is suitable for music mood data. Recently, structured regression was performed for computer vision tasks [6, 30] using Twin Gaussian Process (TGP) to predict the multi-variate output effectively. TGP was first introduced in [6] to solve the structured regression problem by minimizing the Kullback-Leibler (KL) divergence (KLTGP) between the input and output marginal Gaussian Processes (GP) and exploiting the dependencies between multi-dimensional structured output. TGP was applied to human pose estimation and was shown to perform remarkably well relative to conventional GP regression and $K$-nearest neighbors regression techniques. A generic version of TGP was proposed in [30] that measures the Sharma-Mittal (SM) divergence between the marginal GPs. Sharma-Mittal TGP (SMTGP) was shown to outperform KLTGP while having the same quadratic computational complexity as that of KLTGP. In [31], a computationally efficient version of [6] known as dTGP (direct TGP) was proposed resulting in a closed-form solution that can be analytically solved without the use of quasi-Newton optimizers.

## 3. Proposed methodology

Fig. 1 shows the architecture of the proposed methodology where for a given dataset consisting of music clips and the corresponding annotations obtained via crowdsourcing, the estimated consensus $\mathbf{Y}$ is derived using an EM algorithm (Sec. 3.1). For each music clip, the feature representation (Sec. 3.2) in terms of BAGMM (Bayesian Acoustic Gaussian Mixture Model) posterior probabilities and CPH (chord progression histogram) features is computed. The dimensionality of features is further reduced using an adaptive kernel PCA (Sec. 3.3) resulting in $\mathbf{X}$. The features and multi-valued target

are then used to train a single computationally efficient structured Gaussian process regression model (Sec. 3.4).

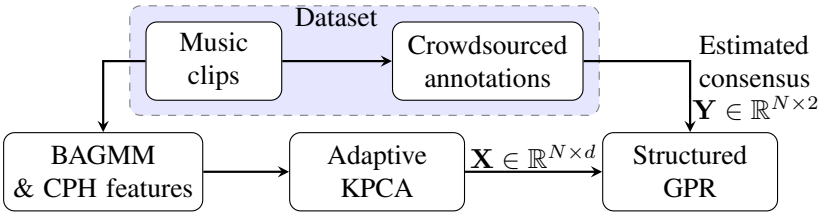



Figure 1: Architecture of the proposed methodology

### 3.1. Estimating consensus from crowdsourced annotations

For music mood estimation, two benchmark datasets, AMG [3] and DEAM [32], are used that contains crowdsourced annotations of (AMG: 1608, DEAM: 1802) songs annotated by (AMG: 665, DEAM: 194) users, along the valence and arousal (VA) dimensions with values in the range $[-1, 1]$. To measure the inter-annotator agreement, the consistency score $J$ is evaluated given by Eq. (1) for $N$ songs in the dataset. For each song $i$, its median annotation $y_{i_m}$ is computed over all its annotated responses $y_i^j$ and $\mathcal{A}_i$ is the set of workers that annotated song $i$. Intuitively, $J$ measures the average deviation relative to the median, thus lower value of $J$ is better.

$$J = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\frac{\sum_{j\in\mathcal{A}_i}(y_i^j - y_{i_m})^2}{|\mathcal{A}_i|}} \tag{1}$$

For the AMG dataset, $J_{\text{Arousal}} = 0.3580$ and $J_{\text{Valence}} = 0.3751$, whereas for the DEAM dataset, $J_{\text{Arousal}} = 0.3156$ and $J_{\text{Valence}} = 0.3261$; these values indicate that not all annotators agree on the VA estimates of given songs. Since there is no gold standard (ground truth) response available for each song, we must estimate it from the available crowdsourced data. A trivial approach is to compute average values along valence and arousal dimensions, however, this assumes that all annotators are equally reliable which may not be true in practice. Extending the work of learning from crowdsourced data [14] for single-dimension target regression case, we derive a maximum-likelihood solution to determine the two-dimensional estimated consensus as well as model the behavior of each annotator. Several annotator behaviors (spammer, adversarial, biased, competent, etc.) have been modeled in the literature for classification tasks (e.g. [15]), however, for the regression task, the annotator behavior modeling remains a problem to be solved. These behaviors can occur due to varying expertise, bad intent or low-attention of annotators. Inferring such behavior can be helpful to determine the annotation consensus. In this work, we model the behavior of adversarial, biased and reliable annotators for the multi-dimensional regression task.

Consider the dataset $\mathcal{D} = \{\mathbf{y}_i^1, \mathbf{y}_i^2, \ldots, \mathbf{y}_i^R\}_{i=1}^N$ containing two-dimensional VA responses of $N$ music clips by maximum $R$ annotators, where each worker annotates only a subset of $N$ songs through a crowdsourcing platform such as Amazon Mechanical Turk (AMT). To model the behavior of $j^{\text{th}}$ annotator, three parameters are considered:

a) **adversariness** $a^j$: if the annotator is adversarial, $a^j = -1$, else $a^j = 1$, thus $a^j \in \{-1, 1\}$ and $\left(a^j\right)^2 = 1$;
b) **bias** $b^j$: a Normal prior $\mathcal{N}(b^j|\mu_b, s_b)$ is used, where $\mu_b = 0$ to favor unbiased annotators and $s_b = 0.05$ to allow for some positive and negative bias;
c) **variability** $\alpha^j$: this measures the variance of the $j^{\text{th}}$ annotator, thus lower is better.

With $\mathbf{y}_i \in \mathbb{R}^2$ as the unknown ground truth for the $i^{\text{th}}$ music clip, a Gaussian distribution model $\mathcal{N}(\mathbf{y}_i^j|\mathbf{y}_i, \alpha^j, a^j, b^j)$ is assumed where $\mathbf{y}_i^j$ is the response of the $j^{th}$ annotator. The parameters to be estimated are $\boldsymbol{\theta} = \{\mathbf{y}, \boldsymbol{\alpha}, \mathbf{a}, \mathbf{b}\}$ with the likelihood given by Eq. (2) assuming all clips are annotated independently by $R$ annotators.

$$\begin{aligned} P(\mathcal{D}|\boldsymbol{\theta}) &= \prod_{i=1}^{N}\prod_{j=1}^{R}\mathcal{N}(\mathbf{y}_i^j|\mathbf{y}_i, \alpha^j, a^j, b^j) \times \prod_{j=1}^{R}\mathcal{N}(b^j|\mu_b, s_b) \\ &= \prod_{i=1}^{N}\prod_{j=1}^{R}\frac{1}{\sqrt{2\pi\alpha^j}}\exp\left[\frac{-1}{2\alpha^j}\|\mathbf{y}_i^j - a^j(\mathbf{y}_i + b^j\mathbf{1})\|_2^2\right] \times \prod_{j=1}^{R}\frac{1}{\sqrt{2\pi s_b}}\exp\left[\frac{-1}{2s_b}\left(b^j - \mu_b\right)^2\right] \end{aligned} \tag{2}$$

In general, since not all instances will be annotated by each annotator, define $\mathcal{A}_i$ as the set of annotators providing the response for the $i^{\text{th}}$ song and $\mathcal{R}_j$ as the set of songs for which the $j^{\text{th}}$ annotator provided the response. Obtaining the gradients of log-likelihood $\ln P(\mathcal{D}|\boldsymbol{\theta})$ with respect to the parameters results in the maximum likelihood solution (derived in Sec. Appendix) given by Eq. (3). This is equivalent to the EM (Expectation-Maximization) algorithm where the E-step determines the estimated consensus $\hat{\mathbf{y}}_i$ and the M-step determines the annotator parameters (adversariness $\hat{a}^j$, bias $\hat{b}^j$ and variability $\hat{\alpha}^j$). These two steps are iterated till convergence (e.g. delta change of $\|\hat{\mathbf{y}}\| < 10^{-6}$). The EM algorithm is initialized with $\hat{b}^j = 0$, $\hat{\mathbf{y}}_i$ as the median of $\{\mathbf{y}_i^j\}_{j\in\mathcal{A}_i}$, $\hat{\alpha}^j$ as the variance of $\{\mathbf{y}_i^j\}_{j\in\mathcal{A}_i}$ and $\hat{a}^j$ as given in Eq. (3).

$$\begin{aligned} \hat{\mathbf{y}}_i &= \frac{1}{\sum_{j\in\mathcal{A}_i}\frac{1}{\hat{\alpha}^j}}\sum_{j\in\mathcal{A}_i}\frac{1}{\hat{\alpha}^j}(\hat{a}^j\mathbf{y}_i^j - \hat{b}^j\mathbf{1}), \quad \hat{a}^j = \text{sgn}\left(\sum_{i\in\mathcal{R}_j}\mathbf{y}_i^{j\intercal}(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\right) \\ \hat{b}^j &= \frac{1}{|\mathcal{R}_j| + \frac{\hat{\alpha}^j}{s_b}}\left(\sum_{i\in\mathcal{R}_j}\left(\hat{a}^j\mathbf{y}_i^j - \hat{\mathbf{y}}_i\right)^\intercal\mathbf{1} + \frac{\hat{\alpha}^j\mu_b}{s_b}\right), \quad \hat{\alpha}^j = \frac{1}{|\mathcal{R}_j|}\sum_{i\in\mathcal{R}_j}\|\mathbf{y}_i^j - \hat{a}^j(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\|_2^2 \end{aligned} \tag{3}$$

Since both AMG and DEAM datasets have the (commonly occurring) long-tail problem, the solution of Eq. (3) will be over-optimistic since very few annotators (36 out of 665 in case of AMG and 27 out of 194 in case of DEAM) provided responses to more than 200 music clips and most of the clips were annotated by few users. To handle this problem, we consider the $(1-\beta)$ confidence interval $(CI)$ of annotator variability where $\beta$ is the significance value (e.g. 5%). Since the sum of squares of Gaussian random variables follows a $\chi^2$ distribution, the $(1-\beta)$ confidence interval is obtained given by Eq. (4).

$$\frac{1}{\alpha^j}\sum_{i\in\mathcal{R}_j}\|\mathbf{y}_i^j - \hat{a}^j(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\|_2^2 = \frac{|\mathcal{R}_j|\hat{\alpha}^j}{\alpha^j} \sim \chi^2(|\mathcal{R}_j|); \quad CI_{1-\beta} = \left\{\frac{|\mathcal{R}_j|\hat{\alpha}^j}{\chi^2_{(1-\frac{\beta}{2},|\mathcal{R}_j|)}}, \frac{|\mathcal{R}_j|\hat{\alpha}^j}{\chi^2_{(\frac{\beta}{2},|\mathcal{R}_j|)}}\right\} \tag{4}$$

Table 1: Confidence intervals of estimated annotator variabilities

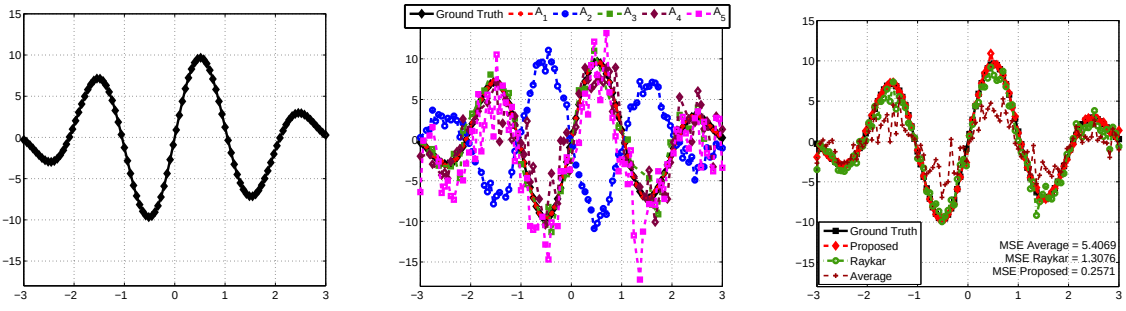
| AnnotatorID | #Annotations | Variability ($\hat{\alpha}$) | 95% Conf. Int. ($CI$) |
|---|---|---|---|
| 159 (AMG) | 924 | 0.4150 | $(0.3610, 0.4208)$ |
| 664 (AMG) | 240 | 0.3879 | $(0.2853, 0.3828)$ |
| **13 (AMG)** | **12** | **0.4983** | $(\mathbf{0.2404}, \mathbf{0.9670})$ |
| **647 (AMG)** | **2** | **0.4981** | $(\mathbf{0.1473}, \mathbf{8.6009})$ |
| 1 (DEAM) | 751 | 0.3360 | $(0.3158, 0.3743)$ |
| 48 (DEAM) | 37 | 0.3568 | $(0.2845, 0.6168)$ |
| **82 (DEAM)** | **12** | **0.3691** | $(\mathbf{0.2371}, \mathbf{0.9540})$ |
| **118 (DEAM)** | **3** | **0.3899** | $(\mathbf{0.2276}, \mathbf{5.0545})$ |

From Table 1, we observe that annotators with IDs 13 and 647 of AMG as well as IDs 82 and 118 of DEAM obtained similar variance $\hat{\alpha}^j$ with Eq. (3), whereas the upper bound (UB) of confidence interval provides a realistic solution. Also, since outliers may exist in the annotated data, instead of removing these outliers with techniques such as minimum covariance determinant (MCD) [33], we use the computationally inexpensive weighted median which is less sensitive to outliers compared to the weighted mean. The resulting equations for estimated consensus are given by Eq. (5), which are iterated till convergence. Note that the updates for $\hat{a}^j$ and $\hat{b}^j$ remain unchanged as in Eq. (3).

$$\hat{\mathbf{y}}_i = \texttt{wMedian}\left(\hat{a}^j \mathbf{y}_i^j - \hat{b}^j \mathbf{1}, \frac{1}{\hat{\alpha}^j}\right), \quad \hat{\alpha}^j = \frac{1}{\chi^2_{(\frac{\beta}{2}, |\mathcal{R}_j|)}} \sum_{i \in \mathcal{R}_j} \|\mathbf{y}_i^j - \hat{a}^j(\hat{\mathbf{y}}_i + \hat{b}^j \mathbf{1})\|_2^2 \tag{5}$$

The proposed algorithm is validated on synthetic as well as benchmark datasets as follows:

(i) For illustrative purposes, 5 annotators are simulated with varying behavior and synthetic data of 100 samples from $y_i^j = f(x_i) + \epsilon^j$ is used. The ground truth data is $f(x) = 10\sin(3x)\cos(\frac{1}{2}x)$ and $\epsilon^j \sim \mathcal{N}(0, \alpha^j)$. The variability levels of annotators are $\boldsymbol{\alpha} = \{0.1, 0.8, 1.5, 2.2, 3\}$ (lower




(a) Ground truth data (b) Data from annotators (c) Estimated Consensus

Figure 2: Illustration of the proposed EM algorithm for obtaining estimated consensus

is better), the number of samples annotated per worker is $|\mathcal{R}| = \{90, 95, 40, 70, 85\}$, the $2^{nd}$ annotator is assumed to be adversarial and the $5^{th}$ annotator is biased. The ground truth data is shown in Fig. 2(a), the annotations are shown in Fig. 2(b) and the estimated consensus is shown in Fig. 2(c), which is also compared with the average estimate as well as the method proposed by Raykar *et al.* [14]. The proposed EM algorithm is effective in determining the estimated consensus as very close to the ground truth data with smaller mean square error (MSE = $5.4069$ with average calculation, $1.3076$ of [14] and $0.2571$ of proposed algorithm), since it can identify the adversarial, biased as well as reliable annotators.

(ii) For the synthetic dataset, we generate $200$ data samples and $300$ annotators having variability levels $\alpha^j$, $j = 1, \ldots, 300$. We consider $300 \times p$ annotators as unreliable (out of which a fraction $q$ are assumed to be adversarial and a fraction $r$ are biased with $s_b \sim \mathcal{U}(0.01, 0.05)$), and the remaining $300 \times (1 - p)$ annotators as reliable. For the $j^{th}$ reliable annotator, we generate $\alpha^j \sim \mathcal{U}(0.01, 0.05)$ and for the unreliable annotator, $\alpha^j$ is generated from $\mathcal{U}(1, 5)$. For every $i^{\text{th}}$ data sample, the number of annotators providing responses $|\mathcal{A}_i|$ is generated from a Poisson distribution $\mathcal{P}(\lambda)$, and $|\mathcal{A}_i|$ annotators are randomly selected to provide the responses. The ground truth is assumed to be $y_i = 1 \; \forall \; i$ and the response $y_i^j$ is generated from a Gaussian distribution $\mathcal{N}(y_i, \alpha^j)$. This implies that the unreliable annotators have significant variability and their responses are likely to be extreme values. The parameter $p$ denotes the fraction of unreliable annotators, $q$ denotes the fraction of adversarial annotators, $r$ denotes the fraction of biased annotators, and $\lambda$ denotes the average number of responses for each data sample. We test the proposed algorithm with $p = 0.25, q = 0.2, r = 0.2, \lambda = 10$. To reduce random errors, we generate $50$ datasets and report the average MSE.

(iii) Next, we consider the benchmark Housing dataset from the UCI machine learning repository consisting of $506$ data samples with 'MEDV' as the ground truth target value $y_i$. In this dataset, $16$ samples have the target value of $50.0$ indicating either missing or censored values, hence these samples are discarded resulting in $490$ samples. The annotators are simulated similar to the setup described in (ii) with $300$ annotators.

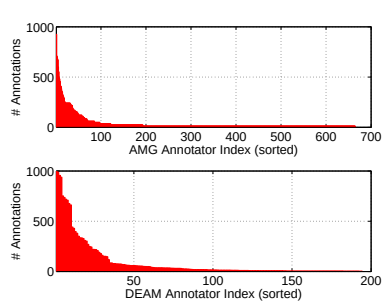
(iv) Finally, the benchmark fact-finding Population dataset [34] is used to validate the proposed algorithm. This dataset reflects the Wikipedia edit history regarding city population for specific years. It consists of $1124$ data samples (city names), $2344$ annotators and $4008$ responses obtained from the crowdsourced data. Among these samples, $308$ samples are labeled with true populations. Pre-processing is done to retain only the latest claim (based on time-stamp) made by annotators for a specific city, and unreasonable claims such as $0$ and $6.5979 \times 10^{18}$ are discarded [18].

Table 2 shows the mean square error (MSE) results on the synthetic and benchmark datasets for the average consensus, consensus obtained with Raykar *et al.* [14] and the proposed algorithm. For the synthetic and Housing datasets, the average MSE of $50$ simulations is reported. In all cases, the proposed algorithm achieves lower MSE due to its ability to identify adversarial as well as biased annotators. The long-tail phenomenon of AMG and DEAM datasets is shown in Fig. 3(a) which is nicely handled by the proposed algorithm and the annotator variabilities are shown in Fig. 3(b)-3(c) where we observe that the method of [14] results in almost similar variability for all annotators

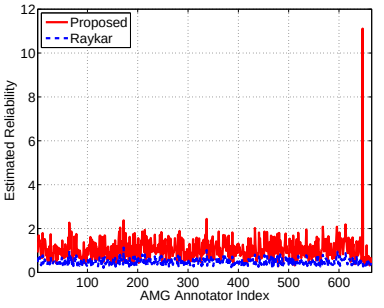
Table 2: Evaluation results (MSE) per dataset and per method

| | Synthetic | Housing | Population |
|---|---|---|---|
| Average | 0.5631 | 0.6548 | 126, 198 |
| Raykar *et al*. [14] | 0.3872 | 0.4317 | 8, 513 |
| Proposed | **0.1436** | **0.2391** | **7, 154** |

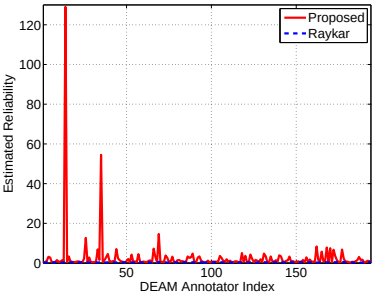
whereas the proposed algorithm considers the effort of $j^{th}$ annotator with respect to the number of songs annotated ($|\mathcal{R}_j|$) using the upper bound of confidence interval. Fig. 3(d) and 3(e) shows the histogram of estimated valence and arousal values for the AMG and DEAM datasets obtained with the proposed algorithm. While AMG spans almost the entire VA space, DEAM is more biased towards the $1^{st}$ quadrant of VA space. An example of the estimated consensus for a sample song of AMG (SongID 714) is shown in Fig. 3(f) along with 15 annotator responses and their variabilities. We observe that the annotators having high variance (less reliability) are given less importance for estimating the consensus, whereas the average value gets biased due to few outliers.



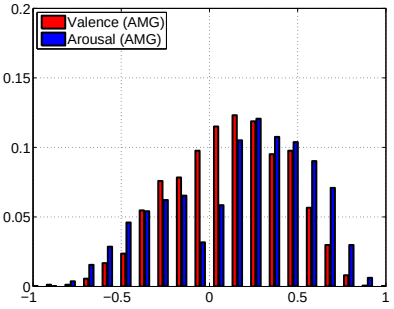

(a) Long tail phenomenon



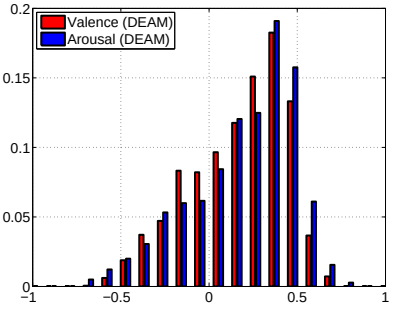

(b) Variabilities (AMG)



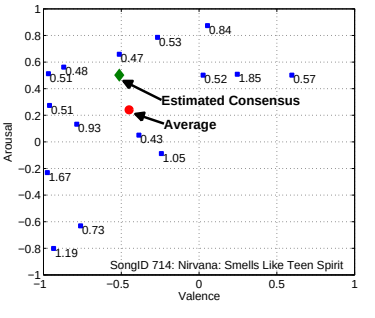

(c) Variabilities (DEAM)




(d) Histogram (AMG)




(e) Histogram (DEAM)




(f) Annotator responses and variabilities

Figure 3: Analysis of AMG and DEAM datasets

## 3.2. Feature representation

For an effective feature representation of each music clip, variational Bayesian inference for acoustic Gaussian Mixture Model (GMM) is proposed to capture the long-term acoustic characteristics of music in a $K$-dimensional probabilistic feature vector. To further improve the valence estimation, chord progression features are computed from the chroma frames.

### 3.2.1. Bayesian acoustic GMM posterior probability

For each music clip in the dataset, standard acoustic features across four categories are computed using the *MIRToolBox* [35]: dynamics (root-mean-square energy), spectral (centroid, spread, skewness, kurtosis, entropy, flatness, $85\%$ roll-off, $95\%$ roll-off, brightness, roughness, irregularity), timbral (zero-crossing rate, flux, 13-dimensional MFCCs, delta MFCCs, delta-delta MFCCs) and tonal (key clarity, musical mode, harmonic change likelihood, 12-bin chroma vector, chroma peak, chroma centroid), resulting in a 70-dimensional feature vector per frame of 50 msec duration with $50\%$ overlap [9]. Each feature dimension is normalized to zero mean and unit standard deviation. Block-level features are computed to capture the temporal characteristics across frames where each block comprises of 16 consecutive frames with an overlap of 12 frames. The block-level feature vector $\mathbf{v}_t$ consists of the mean and standard deviation of frame-based feature vectors [10]. To represent the block-level features as a fixed dimensional vector, [21] computes the mean and standard deviation across all blocks and stacks these by aggregation into a single vector.

For an effective prototypical representation, we adopt the EM-GMM clustering approach of [9] resulting in AGMM (Acoustic Gaussian Mixture Model) posterior probabilities $x_{nk}$ given by Eq. (6), where the universal background model (UBM) parameters $\{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$ indicate the weight, mean and covariance of the $k^{\text{th}}$ latent audio topic (or mixture), and $\mathbf{x}_n \in \mathbb{R}^K$ is the acoustic posterior probability feature vector of the song $s_n$ consisting of $F_n$ blocks. Using the EM algorithm [36], the UBM model is trained with randomly selected $25\%$ block-level feature vectors across the entire dataset (spanning the whole range of VA space) resulting in $215,000$ vectors.

$$p(\mathbf{v}_t) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\mathbf{v}_t | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right); \; x_{nk} = \frac{1}{F_n} \sum_{t=1}^{F_n} \left( \frac{\pi_k \mathcal{N}\left(\mathbf{v}_t | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)}{\sum_{h=1}^{K} \pi_h \mathcal{N}\left(\mathbf{v}_t | \boldsymbol{\mu}_h, \boldsymbol{\Sigma}_h\right)} \right) \tag{6}$$

The most crucial problem of AGMM is determining the exact number of latent audio topics $K$ that can explain the given data. In [9], various values of $K(16, 32, 64, 128, 256, 512)$ were used to determine the regression performance. However, this is ad-hoc as it could lead to an over-fitting of the data. To determine the optimal number of mixtures, we resort to variational Bayesian inference framework [36] resulting in Bayesian Acoustic GMM (BAGMM) posterior probability feature representation for each music clip. With the Bayesian treatment, all parameters of AGMM are assigned conjugate priors: the weight is assigned Dirichlet prior $p(\boldsymbol{\pi}) = C(\alpha_0) \prod_{k=1}^{K} \pi_k^{\alpha_0 - 1}$ with hyper-parameter $\alpha_0$ and $C(\alpha_0)$ as the normalizing constant; mean and covariance $(\boldsymbol{\Lambda}^{-1})$ are assigned Gaussian-Wishart prior $p(\boldsymbol{\mu}, \boldsymbol{\Lambda}) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{\mu}_k | \mathbf{m}_0, (\beta_0 \boldsymbol{\Lambda}_k)^{-1}) \mathcal{W}(\boldsymbol{\Lambda}_k | \mathbf{W}_0, v_0)$ with hyper-parameters $\mathbf{m}_0, \beta_0, \mathbf{W}_0, v_0$. Due to the intractability of true posterior distribution, variational Bayesian inference is used for its approximation by minimizing the Kullback-Leibler (KL) divergence between

the true and approximate posteriors. The algorithm is initialized with $\alpha_0 = 0.001$ (so that the posterior will be influenced primarily by the data), $\mathbf{m}_0$ as the $k$-means centroid of training data (to speed up the convergence), $\beta_0 = 1$, $v_0 = D + 1$, and $\mathbf{W}_0 = 10\mathbf{I}$ to avoid mixtures getting trapped in local maximum, i.e. no single Gaussian can stay in the neighborhood of a possible bad saddle point. Here, $\mathbf{I}$ is the identity matrix and $D$ is the dimensionality of block-level features. We refer the reader to [36] (Chap. 10) for the detailed formulation of variational posteriors and the variational EM algorithm.

The expected value of mixing weights in the posterior distribution of BAGMM is given by Eq. (7) with $N_k$ as the number of samples in the $k^{\text{th}}$ mixture, where we observe that for uninformative priors ($\alpha_0 \to \infty$), the expected value converges to a small value of $\xi$ ($0 < \xi < \frac{1}{K}$). Thus, the mixtures having $N_k \approx 0$ and $\alpha_k \approx \alpha_0$ do not play a significant role and are discarded in the next update. BAGMM thus determines the optimal number of latent audio topics without using cross-validation ($K_{opt} = 117$ in this work), overcomes the problem of singularity [36] (that occurs in AGMM) due to the positive-definite hyper-parameter $\mathbf{W}_0$, and prevents over-fitting of data. Each music clip is thus represented as an acoustic posterior probability feature vector $\mathbf{x}_n \in \mathbb{R}^{K_{opt}}$.

$$E[\pi_k] = \frac{\alpha_k + N_k}{K\alpha_0 + \sum_{j=1}^{K} N_j} = \frac{\alpha_0 + 2N_k}{K\alpha_0 + N} \approx \frac{\alpha_0/N}{K\alpha_0/N + 1} = \frac{1}{K + N/\alpha_0} \tag{7}$$

### 3.2.2. Chord progression histogram

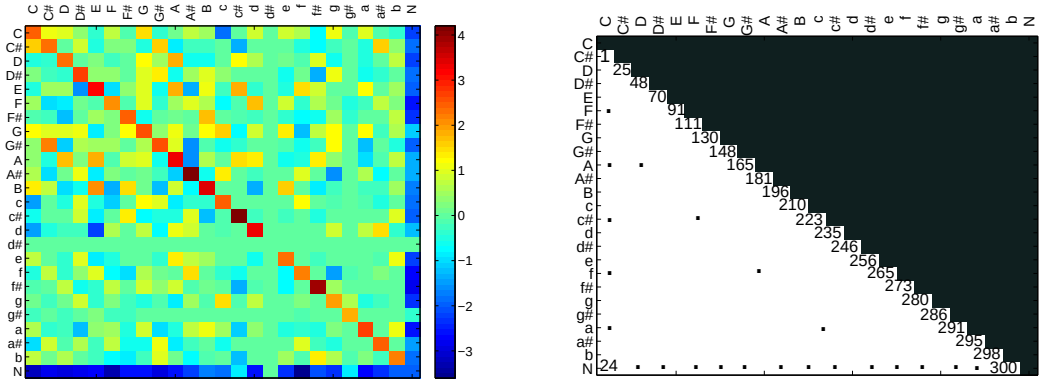
To effectively estimate the valence dimension, we compute chord related features since it is well-known from music theory that major chords induce positive valence while minor chords induce negative valence [37]. For example, valence increment is observed when a $C$ major chord is followed by an $F$ major chord, whereas valence decreases when a $C$ major chord is followed by an $a$ minor chord. By considering the 12 octave-equivalent (mod 12) pitch classes, various possibilities of chords emerge (augmented, diminished, etc.); however, this results in $\sum_{n=1}^{12} \binom{12}{n} = 2^{12} - 1$ combinations that can result in the over-fitting of regression models. Similar to existing work [27, 28], we thus consider the following vocabulary of most frequently occurring chords: 12 major chords ($C$, $C\sharp$, $D$, $D\sharp$, $E$, $F$, $F\sharp$, $G$, $G\sharp$, $A$, $A\sharp$, $B$), 12 minor chords ($c$, $c\sharp$, $d$, $d\sharp$, $e$, $f$, $f\sharp$, $g$, $g\sharp$, $a$, $a\sharp$, $b$) and a neutral chord used during non-music or silence periods. These chords are mapped to the numbers $1, 2, ..., M$ respectively, where 1 represents the $C$ major chord and $M = 25$ represents the neutral chord. Also, enharmonic equivalence is considered, i.e. $C\sharp \equiv D\flat$, $a\sharp \equiv b\flat$, etc.

In [27], chord histogram features were extracted to solve the problem of music emotion classification where the performance of valence estimation was shown to improve considerably (the chord features did not have a significant impact on the arousal dimension). However, it only considers the frequency of chords occurring in the song and ignores the context in which the chords are played. Two songs may have the same chord histograms but the chords could have been played in different orders. In [38], chord progressions were considered using bi-gram and tri-gram modeling resulting in improved valence estimation. In general, $n$-gram modeling can be applied, however, this increases the computational cost substantially; for example, in [38], the 2-step transition resulted in $25^2 = 625$ possibilities, while the 3-step transition resulted in $25^3 = 15,625$ possibilities. In [28], a compact representation of chroma features was computed using chord progression histogram (CPH), which uses multi-probing by considering multiple dominant chords per chroma frame and multiple chord

progressions between two chroma frames. CPH was applied for efficient music retrieval in large-scale datasets using the idea of locality sensitive hashing (LSH). In this work, CPH is modified as a concise feature vector of each music clip to effectively characterize the valence dimension.

The $SVM^{hmm}$ model described by Eq. (8) was used by [28] to identify the dominant chord sequence $\mathbf{s}'$ (SVM for chord recognition per chroma-frame and HMM for chord progressions), where $\mathbf{w}_C$ is a $M \times D_0$ matrix, $D_0$ is the dimensionality of each chroma frame $\mathbf{x}_t$ (extracted using CompFeat [39]), $\varphi_C$ is a $1 \times M$ indicator vector for the chord $s_t$, $\varphi_T$ is an indicator matrix denoting the chord progressions from $s_{t-1}$ to $s_t$, and $\mathbf{w}_T$ is an $M \times M$ transition matrix indicating the likelihood of chord progressions between adjacent frames. The parameters $\mathbf{w}_T$ (shown in Fig. 4(a)) and $\mathbf{w}_C$ are learned using the manually annotated "Beatles" public dataset [40]. Since this model returns only the most dominant chord sequence having the highest score, local multi-probing was used in [28] for determining the chord progressions by which the scores of $M$ paths leading to the same chord $j$ for frame $t$ is recorded. The highest scoring $N_C$ chords are found for each frame and the top $N_P$ chord progressions are recorded in a tuple $\mathbf{z}_t = (i, j, rank_{i,j})$. The histogram of chord progressions is then found using $CPH(h) = CPH(h) + w$, where $h = (i-1) \times M + j$ is the histogram bin ($h \in [1, 625]$) that map the chord indices $i \rightarrow j$ to the bin $h$ and $w = N_P - rank_{i,j} + 1$ reflects the weight to be assigned to each chord progression. The CPH is updated for each chroma frame resulting in a $1 \times M^2$ histogram of chord progressions. It was shown in [28] that the CPH obtained using multi-probing with $N_C = 16$ and $N_P = 90$ serves as a good summary of the music clip resulting in an effective music retrieval performance. Note that the multi-probed chord progression histogram is an effective improvement over the $n$-gram modeling used in earlier work; however, this advantage has not been studied for music mood estimation.

$$\mathbf{s}' = \underset{\mathbf{s}}{\operatorname{argmax}} \left[ \sum_{t=1,\ldots,l} \varphi_C(s_t) \times \mathbf{w}_C \times \mathbf{x}_t + \varphi_T(s_{t-1}, s_t) \times \mathbf{w}_T \right] \tag{8}$$




(a) Transition matrix $\mathbf{w}_T$ (b) Indices for $CPH$

Figure 4: Chord progression histogram (a) transition matrix, and (b) indices selected to calculate $1 \times 300$ probability feature vector

In this work, further modifications are made by exploiting the properties of chord progression histogram: (i) only the inter-chord progressions (e.g. $d\sharp \rightarrow a\sharp$) are retained and the intra-chord progressions (e.g. $a\sharp \rightarrow a\sharp$) are discarded to consider only the changing melody information; (ii) since the $CPH$ matrix (reshaped as $M \times M$) is somewhat symmetric [28], we consider only $\frac{M(M-1)}{2} = 300$ bins with $CPH(i \rightarrow j) = \max(CPH(i \rightarrow j), CPH(j \rightarrow i))$. The resulting selected indices of the $CPH$ matrix are shown in Fig. 4(b), which is then normalized to obtain a $1 \times 300$ probability feature vector corresponding to the chord progressions. To investigate its effectiveness, the modified CPH is obtained for each music clip of the AMG dataset and a histogram of the most frequently occurring chord progressions in AMG is shown in Fig. (5). It can be observed that the music clips having positive valence consists of many major chord progressions while the music clips having negative valence consists of many minor chord progressions that are rarely occurring or missing for the clips having positive valence. Thus, the modified CPH can serve as a useful feature for estimating the valence dimension of music clips. The extracted BAGMM and modified CPH features are concatenated resulting in the feature vector $\mathbf{x} \in \mathbb{R}^D$ for each music clip, where $D = K_{opt} + 300$.

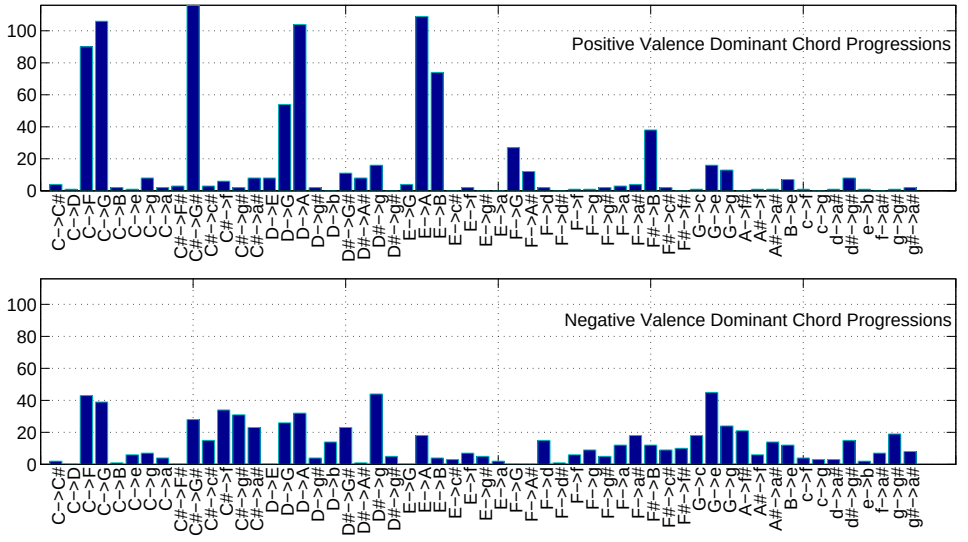



Figure 5: Histogram of dominant chord progressions in the AMG dataset

## 3.3. Dimensionality reduction with adaptive kernel PCA

To reduce the feature dimensionality, kernel principal component analysis (KPCA) [36] is used to handle non-linear projection of the data. However, the kernel parameters usually need to be fine-tuned through cross-validation for an effective performance. An adaptive version of KPCA was proposed in [41] to address this issue and the solution was obtained in an iterative manner. An approximate version of adaptive KPCA was also proposed in [41] resulting in a closed-form solution that can be obtained analytically. Since the feature vectors obtained from BAGMM and CPH representations lie in a higher dimensional space $\mathbf{X} \in \mathbb{R}^{N \times D}$, we use approximate adaptive KPCA (AKPCA) to project this data non-linearly and reduce the feature dimensionality in an unsupervised manner.

In AKPCA, multiple kernels are used with varying parameters instead of using a single kernel and 2D feature matrices $\mathbf{\Phi}_1, \mathbf{\Phi}_2 \dots \mathbf{\Phi}_N$ are constructed that represent the set of non-linear mappings $\phi_k : \mathbf{x} \in \mathbf{X} \rightarrow \phi_k(\mathbf{x}) \in \mathcal{H}_k$ for $k = 1, 2, \dots f$, from the original input space $\mathbf{X}$ to the high dimensional feature space $\mathcal{H}_k$, where $f$ is the number of kernels used and $\mathbf{\Phi}_n = (\phi_1(\mathbf{x}_n), \phi_2(\mathbf{x}_n) \dots \phi_f(\mathbf{x}_n))$. For each input training instance $\mathbf{x}_n$, a kernel matrix $\mathbf{K}_n \in \mathbb{R}^{N \times f}$ is constructed having elements $\mathbf{K}_n(i,k) = \mathcal{K}^k(\mathbf{x}_i, \mathbf{x}_n) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_n\|^2}{2\sigma_k^2}\right)$ when the radial basis function kernel is used with the parameter $\sigma_k^2 = k \times s_0$, where $s_0$ is the variance of the training data $\mathbf{X}$. The optimization function of approximate AKPCA seeks to find two matrices $\mathbf{L}, \mathbf{R}$ and is given by Eq. (9) where $\mathbf{L} \in \mathbb{R}^{N \times d}$ and $\mathbf{R} \in \mathbb{R}^{f \times g}$ (here, $F$ is the Frobenius norm and $\mathbf{I}$ is the identity matrix).

$$\begin{aligned} \max_{\mathbf{L}, \mathbf{R}} \quad & \sum_{n=1}^{N} \|\mathbf{L}^\intercal \mathbf{K}_n\|_F^2 + \sum_{n=1}^{N} \|\mathbf{K}_n \mathbf{R}\|_F^2 \\ \text{s.t.} \quad & \mathbf{L}^\intercal \mathbf{L} = \mathbf{I},\ \mathbf{R}^\intercal \mathbf{R} = \mathbf{I} \end{aligned} \tag{9}$$

The solution is obtained as $\mathbf{L}$ being the $d$ eigen-vectors corresponding to the $d$ largest eigen-values of the matrix $\mathbf{M}_L = \sum_{n=1}^{N} \mathbf{K}_n \mathbf{K}_n^\intercal$ and $\mathbf{R}$ being the $g$ eigen-vectors corresponding to the $g$ largest eigen-values of the matrix $\mathbf{M}_R = \sum_{n=1}^{N} \mathbf{K}_n^\intercal \mathbf{K}_n$. For extracting non-linear features of a specific data sample $\mathbf{x}_*$, the kernel matrix $\mathbf{K}_*(i,k) = \mathcal{K}^k(\mathbf{x}_i, \mathbf{x}_*)$ is constructed and projected according to $\mathbf{x}_{\text{akpca}} = \mathbf{L}^\intercal \mathbf{K}_* \mathbf{R}$. Thus, we obtain $\mathbf{X} \in \mathbb{R}^{N \times d}$ that represents features with reduced dimensionality $d << D$. In this work, we use $f = 10, g = 1$ and $d$ was obtained by finding the intrinsic dimensionality of $\mathbf{X}$ retaining $95\%$ of its explained variance.

### 3.4. Structured regression

The conventional regression approach of learning independently two different regressors does not consider any correlation that may exist in the multivariate responses. From the estimated consensus of AMG dataset, we find that the correlation coefficient between valence and arousal is $0.325$, which should be accounted for during training of the regression model. We follow the work of [6] that introduced Twin Gaussian Processes (TGP) with KL divergence measure and [30] that used Sharma-Mittal (SM) divergence measure. SMTGP was shown to perform better than KLTGP for structured regression. However, both KLTGP and SMTGP have quadratic computational complexity since they need to solve a non-linear optimization problem requiring L-BFGS solver. An efficient approach termed as direct TGP (dTGP) was proposed in [31] that require only $\mathcal{O}(N)$ computations compared to $\mathcal{O}(N^2)$ of KLTGP and SMTGP.

In TGPs, the joint distributions of input data given by $p(\mathbf{X}, \mathbf{x}) = \mathcal{N}_{\mathbf{X}}(\mathbf{0}, \mathbf{K}_{\mathbf{X} \cup \mathbf{x}})$, and output data given by $p(\mathbf{Y}, \mathbf{y}) = \mathcal{N}_{\mathbf{Y}}(\mathbf{0}, \mathbf{K}_{\mathbf{Y} \cup \mathbf{y}})$ are used, where the joint kernels are defined in Eq. (10) with $\mathbf{x}$ as the new test point corresponding to the unknown (multi-dimensional) output $\mathbf{y}$. Using Gaussian-RBF kernel functions, the similarity kernels for input and output are given by Eq. (11), where $\rho_{\mathbf{X}}$ and $\rho_{\mathbf{Y}}$ correspond to the kernel bandwidths, $\lambda_{\mathbf{X}}$ and $\lambda_{\mathbf{Y}}$ are the regularization parameters to avoid over-fitting, and $\delta$ is the Kronecker delta function.

$$\mathbf{K}_{\mathbf{X}\cup\mathbf{x}} = \begin{bmatrix} \mathbf{K}_{\mathbf{X}} & \mathbf{k}_{\mathbf{X}}^{\mathbf{x}} \\ \mathbf{k}_{\mathbf{X}}^{\mathbf{x}\intercal} & k_{\mathbf{X}}(\mathbf{x},\mathbf{x}) \end{bmatrix},\ \mathbf{K}_{\mathbf{Y}\cup\mathbf{y}} = \begin{bmatrix} \mathbf{K}_{\mathbf{Y}} & \mathbf{k}_{\mathbf{Y}}^{\mathbf{y}} \\ \mathbf{k}_{\mathbf{Y}}^{\mathbf{y}\intercal} & k_{\mathbf{Y}}(\mathbf{y},\mathbf{y}) \end{bmatrix} \tag{10}$$

$$(\mathbf{K}_{\mathbf{X}})_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\rho_{\mathbf{X}}^2}\right) + \lambda_{\mathbf{X}}\delta_{ij};\quad (\mathbf{K}_{\mathbf{Y}})_{ij} = \exp\left(-\frac{\|\mathbf{y}_i - \mathbf{y}_j\|^2}{2\rho_{\mathbf{Y}}^2}\right) + \lambda_{\mathbf{Y}}\delta_{ij} \tag{11}$$

Given $N$ training instances of feature vectors $\mathbf{X} \in \mathbb{R}^{N\times d}$ and the corresponding output estimated consensus $\mathbf{Y} \in \mathbb{R}^{N\times 2}$, dTGP optimizes Eq. (12), i.e. the value of output kernel function $\mathbf{t}_{\mathbf{y}} = \mathbf{k}_{\mathbf{Y}}^{\mathbf{y}}$ having elements $(\mathbf{k}_{\mathbf{Y}}^{\mathbf{y}})_i = \mathbf{K}_{\mathbf{Y}}(\mathbf{y}_i, \mathbf{y})$ instead of directly optimizing for $\mathbf{y}$. Here, $\mathbf{u}_{\mathbf{X}} = \mathbf{K}_{\mathbf{X}}^{-1}\mathbf{k}_{\mathbf{X}}^{\mathbf{x}}$ and $\eta_{\mathbf{X}} = k_{\mathbf{X}}(\mathbf{x},\mathbf{x}) - \mathbf{k}_{\mathbf{X}}^{\mathbf{x}\intercal}\mathbf{u}_{\mathbf{X}}$ is the Schur complement of the joint kernel $\mathbf{K}_{\mathbf{X}\cup\mathbf{x}}$. The optimal solution is given by Eq. (13) found analytically using simple algebraic manipulations with complexity $\mathcal{O}(N)$ instead of using a quasi-Newton optimizer.

$$\begin{aligned} \min_{\mathbf{t}_{\mathbf{Y}}} \quad & \left[1 + \lambda_{\mathbf{Y}} - 2\mathbf{t}_{\mathbf{Y}}^{\intercal}\mathbf{u}_{\mathbf{X}} - \eta_{\mathbf{X}}\log(1 + \lambda_{\mathbf{Y}} - \mathbf{t}_{\mathbf{Y}}^{\intercal}\mathbf{K}_{\mathbf{Y}}^{-1}\mathbf{t}_{\mathbf{Y}})\right] \\ \text{s.t.} \quad & 0 \le t_{\mathbf{Y},i} \le 1 + \lambda_{\mathbf{Y}},\ i = 1, 2, \ldots N \end{aligned} \tag{12}$$

$$\begin{aligned} \hat{\mathbf{t}}_{\mathbf{Y}} &= \min\left((1+\lambda_{\mathbf{Y}})\mathbf{1}, \max(\mathbf{0}, \zeta\mathbf{K}_{\mathbf{Y}}\mathbf{u}_{\mathbf{X}})\right) \\ \zeta &= \frac{-\eta_{\mathbf{X}} + \sqrt{\eta_{\mathbf{X}}^2 + 4\mathbf{u}_{\mathbf{x}}^{\intercal}\mathbf{K}_{\mathbf{Y}}\mathbf{u}_{\mathbf{X}}\,(1+\lambda_{\mathbf{Y}})}}{2\mathbf{u}_{\mathbf{x}}^{\intercal}\mathbf{K}_{\mathbf{Y}}\mathbf{u}_{\mathbf{X}}} \end{aligned} \tag{13}$$

The parameters of dTGP were found empirically via grid-search: $\lambda_{\mathbf{X}} = 10^{-4}, \lambda_{\mathbf{Y}} = 10^{-4}, 2\rho_{\mathbf{X}}^2 = 100, 2\rho_{\mathbf{Y}}^2 = 1$. To simplify computations, dTGP further finds $P$ nearest neighbors $\{\hat{\mathbf{y}}_p, \hat{t}_{\mathbf{Y},p}\}_{p=1}^{P}$ from $\{\hat{\mathbf{y}}_i\}_{i=1}^{N}$ for a specific test sample based on the weights $\gamma_p$ and computes the bi-variate output estimate $\mathbf{y}$ as the weighted sum of $P$ nearest training instances ($P$ = 30 set empirically) given by Eq. (14). The weight $\gamma_p$ captures the input-output as well as between-output correlation for structured regression [31].

$$\gamma_p = \frac{\hat{t}_{\mathbf{Y},p}}{\sum_{p=1}^{P}\hat{t}_{\mathbf{Y},p}};\ \mathbf{y} = \sum_{p=1}^{P}\gamma_p\hat{\mathbf{y}}_p \tag{14}$$

# 4. Experimental results

The performance of proposed work is evaluated on AMG and DEAM datasets with two metrics: the coefficient of determination $R^2$ and the root mean square error $RMSE$ given by Eq. (15) to measure the regression performance between the predicted VA values and the estimated consensus, where $\hat{y}$ is the predicted mood dimension (either valence or arousal), $y^*$ is the estimated consensus and $\bar{y}$ is the corresponding average value.

$$R^2 = 1 - \frac{\sum_i(\hat{y}_i - y_i^*)^2}{\sum_i(\hat{y}_i - \bar{y})^2};\quad RMSE = \sqrt{\sum_i(\hat{y}_i - y_i^*)^2} \tag{15}$$

Table 3: Performance of estimating music mood on the AMG dataset

| Method | $R^2_{\text{Arousal}}$ | $RMSE_{\text{Arousal}}$ | $R^2_{\text{Valence}}$ | $RMSE_{\text{Valence}}$ |
|---|---|---|---|---|
| GPR [21] | $0.654 \pm 0.046$ | $0.247 \pm 0.026$ | $0.429 \pm 0.071$ | $0.283 \pm 0.036$ |
| HDM ($G$=7) [10] | $0.632 \pm 0.013$ | $0.258 \pm 0.031$ | $0.352 \pm 0.064$ | $0.291 \pm 0.028$ |
| Aggregate GPR [8] | $0.678 \pm 0.029$ | $0.203 \pm 0.019$ | $0.437 \pm 0.058$ | $0.236 \pm 0.018$ |
| {BAGMM, AKPCA, KLTGP} | $0.717 \pm 0.051$ | $0.184 \pm 0.024$ | $0.502 \pm 0.074$ | $0.237 \pm 0.024$ |
| {**BAGMM, AKPCA, SMTGP**} | $\mathbf{0.721} \pm 0.067$ | $\mathbf{0.165} \pm 0.017$ | $0.512 \pm 0.051$ | $0.224 \pm 0.016$ |
| {BAGMM, AKPCA, dTGP} | $0.718 \pm 0.048$ | $0.168 \pm 0.028$ | $0.508 \pm 0.052$ | $0.231 \pm 0.012$ |
| {CPH, AKPCA, dTGP} | $0.704 \pm 0.037$ | $0.176 \pm 0.015$ | $\mathbf{0.521} \pm 0.042$ | $\mathbf{0.218} \pm 0.019$ |
| {**BAGMM, CPH, AKPCA, dTGP**} | $\mathbf{0.718} \pm 0.025$ | $\mathbf{0.161} \pm 0.034$ | $\mathbf{0.529} \pm 0.039$ | $\mathbf{0.212} \pm 0.014$ |

Table 4: Performance of estimating music mood on the DEAM dataset

| Method | $R^2_{\text{Arousal}}$ | $RMSE_{\text{Arousal}}$ | $R^2_{\text{Valence}}$ | $RMSE_{\text{Valence}}$ |
|---|---|---|---|---|
| GPR [21] | $0.631 \pm 0.031$ | $0.247 \pm 0.042$ | $0.429 \pm 0.041$ | $0.283 \pm 0.041$ |
| HDM ($G$=7) [10] | $0.647 \pm 0.046$ | $0.258 \pm 0.073$ | $0.352 \pm 0.032$ | $0.291 \pm 0.053$ |
| Aggregate GPR [8] | $0.681 \pm 0.052$ | $0.203 \pm 0.089$ | $0.437 \pm 0.044$ | $0.236 \pm 0.029$ |
| {BAGMM, AKPCA, KLTGP} | $0.714 \pm 0.063$ | $0.189 \pm 0.046$ | $0.513 \pm 0.053$ | $0.238 \pm 0.033$ |
| {**BAGMM, AKPCA, SMTGP**} | $\mathbf{0.729} \pm 0.029$ | $\mathbf{0.161} \pm 0.033$ | $0.524 \pm 0.071$ | $0.225 \pm 0.046$ |
| {BAGMM, AKPCA, dTGP} | $0.722 \pm 0.071$ | $0.172 \pm 0.029$ | $0.517 \pm 0.019$ | $0.226 \pm 0.072$ |
| {CPH, AKPCA, dTGP} | $0.718 \pm 0.021$ | $0.169 \pm 0.014$ | $\mathbf{0.534} \pm 0.022$ | $\mathbf{0.204} \pm 0.029$ |
| {**BAGMM, CPH, AKPCA, dTGP**} | $\mathbf{0.727} \pm 0.023$ | $\mathbf{0.163} \pm 0.025$ | $\mathbf{0.538} \pm 0.031$ | $\mathbf{0.184} \pm 0.031$ |

Tables 3 and 4 show the performance of existing state-of-the-art techniques and proposed work. For each dataset, the data is randomly partitioned into 10 mutually exclusive subsets $\mathcal{Q}_1, \mathcal{Q}_2, \ldots, \mathcal{Q}_{10}$, each of approximately equal size. In the $i^{\text{th}}$ experiment $(i = 1, \ldots, 10)$, the testing subset $\mathcal{Q}_i$ is used for model evaluation and the remaining training subsets are used for fitting the regression model. The means and standard deviations of the evaluation metric values resulting from these 10 experiments are reported to measure the performance (10-fold cross-validation) [42].

For [21], conventional acoustic features aggregated with mean and standard deviation were used and GPR was used with the combination of Squared Exponential and Rational Quadratic kernels [29]. For histogram density modeling (HDM) of mood annotations, the number of latent histograms was set to 256 as reported in [10]. For [8], conventional acoustic features were used with an adaptive aggregation of GP regressors. For GPR [21] and Aggregate GPR [8], two independent regression models were used for valence and arousal estimation with average annotation value set as the ground truth, while HDM [10] needed fine-tuning of number of latent histograms to model each latent audio topic as well as the choice of grid size $G \times G$ to represent the VA space as a heatmap.

The structured regression performance of TGPs is evaluated using the proposed feature representation (BAGMM, CPH, AKPCA) and the estimated consensus set as ground truth with a single regression model. For KLTGP and SMTGP, BAGMM features are used followed by dimensionality reduction with AKPCA. Compared to estimating valence and arousal dimensions separately using [8], [10] and [21], the structured regression techniques (KLTGP, SMTGP and dTGP) perform better due to the joint estimation of bi-variate output. Also, there is a significant improvement in valence estimation due to the inclusion of modified chord progression histogram features. The performance of dTGP is similar to that of SMTGP for arousal estimation, but note that SMTGP incurs quadratic computational complexity due to the inversion of kernel matrices during optimization, while dTGP has a linear computational cost. Overall, we observe an improvement in $R^2$ of $9.7\%$ for arousal and $21.9\%$ for valence for the AMG dataset and an improvement of $15.2\%$ for arousal and $23.7\%$ for valence for the DEAM dataset relative to the baseline GPR [21].

## 5. Conclusions

A structured regression framework is presented in this work for affective modeling of music mood using TGPs and is shown to perform better than independently learning different regressors for valence and arousal dimensions. An EM algorithm is proposed to model the varying behavior of annotators and the confidence-interval based estimated consensus is derived to serve as the ground truth for regression. Using Bayesian inference, acoustic features are represented with BAGMM posterior probabilities where the optimal number of Gaussian mixtures are determined automatically from the training data and the drawback of singularity and over-fitting of conventional GMM is simultaneously avoided. The modified chord progression histogram features are obtained by exploiting the properties of CPH and used for effective valence estimation. An adaptive KPCA dimensionality reduction technique is used to reduce the feature dimensions in a non-linear projected feature space. The proposed framework achieves improvement in music mood estimation with the direct TGP implementation that achieves optimization at a linear computational cost and avoids the use of non-linear quadratic optimization of KLTGP and SMTGP.

## Appendix

## A. Derivation of estimated consensus

The log-likelihood of Eq. (2) is given by Eq. (16) where the constant $C = \frac{-(N+1)R}{2}\ln(2\pi) - \frac{R}{2}\ln(s_b)$. Equating its gradients with respect to each parameter of $\boldsymbol{\theta}$ to 0, the updates are obtained given by Eq. (17) - (20) using the fact that $\|\mathbf{z}\|_2^2 = \mathbf{z}^\intercal\mathbf{z}$ and $\frac{d}{d\mathbf{z}}\mathbf{z}^\intercal\mathbf{z} = 2\mathbf{z}^\intercal$.

$$\ln P(\mathcal{D}|\boldsymbol{\theta}) = C - \frac{N}{2}\sum_{j=1}^{R}\ln(\alpha^j) - \sum_{i=1}^{N}\sum_{j=1}^{R}\frac{1}{2\alpha^j}\|\mathbf{y}_i^j - a^j(\mathbf{y}_i + b^j\mathbf{1})\|_2^2 - \sum_{j=1}^{R}\frac{1}{2s_b}\left(b^j - \mu_b\right)^2 \qquad (16)$$

$$\frac{\partial \ln P(\mathcal{D}|\boldsymbol{\theta})}{\partial \mathbf{y}_i} = \mathbf{0} \implies \sum_{j=1}^{R} \frac{1}{\hat{\alpha}^j} \left(\mathbf{y}_i^j - \hat{a}^j(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\right)^\intercal \hat{a}^j = \mathbf{0},$$

$$\text{Applying transpose on both sides, } \sum_{j=1}^{R} \frac{\hat{a}^j}{\hat{\alpha}^j}\mathbf{y}_i^j = \sum_{j=1}^{R} \frac{1}{\hat{\alpha}^j}\left(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1}\right), \text{ since } \left(\hat{a}^j\right)^2 = 1, \tag{17}$$

$$\hat{\mathbf{y}}_i \sum_{j=1}^{R} \frac{1}{\hat{\alpha}^j} = \sum_{j=1}^{R} \frac{\left(\hat{a}^j\mathbf{y}_i^j - \hat{b}^j\mathbf{1}\right)}{\hat{\alpha}^j}, \quad \therefore \boxed{\hat{\mathbf{y}}_i = \frac{1}{\sum_{j=1}^{R} \frac{1}{\hat{\alpha}^j}} \sum_{j=1}^{R} \frac{\left(\hat{a}^j\mathbf{y}_i^j - \hat{b}^j\mathbf{1}\right)}{\hat{\alpha}^j}}$$

$$\frac{\partial \ln P(\mathcal{D}|\boldsymbol{\theta})}{\partial a^j} = 0 \implies -\frac{1}{\hat{\alpha}^j} \sum_{i=1}^{N} \left(\mathbf{y}_i^j - a^j(\mathbf{y}_i + b^j\mathbf{1})\right)^\intercal \times \left(\mathbf{y}_i + b^j\mathbf{1}\right) \times (-1) = 0,$$

$$\sum_{i=1}^{N} \mathbf{y}_i^{j\intercal}(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1}) = \hat{a}^j \sum_{i=1}^{N} \|\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1}\|_2^2 \tag{18}$$

$$\therefore \boxed{\hat{a}^j = \text{sgn}\left(\sum_{i=1}^{N} \mathbf{y}_i^{j\intercal}(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\right)}, \text{ since } \|\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1}\|_2^2 \text{ is always positive}$$

$$\frac{\partial \ln P(\mathcal{D}|\boldsymbol{\theta})}{\partial b^j} = 0 \implies \sum_{i=1}^{N} \frac{1}{\hat{\alpha}^j} \left(\mathbf{y}_i^j - \hat{a}^j(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\right)^\intercal \times (-\hat{a}^j\mathbf{1}) + \frac{1}{s_b}(\hat{b}^j - \mu_b) = 0$$

$$\therefore \sum_{i=1}^{N} \frac{\hat{a}^j}{\hat{\alpha}^j} \left(\mathbf{y}_i^j - \hat{a}^j(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\right)^\intercal \mathbf{1} = \frac{1}{s_b}(\hat{b}^j - \mu_b)$$

$$\therefore \sum_{i=1}^{N} \left(\hat{a}^j\mathbf{y}_i^j - \hat{\mathbf{y}}_i\right)^\intercal \mathbf{1} - N\hat{b}^j = \frac{\hat{\alpha}^j}{s_b}(\hat{b}^j - \mu_b), \text{ since } \left(\hat{a}^j\right)^2 = 1 \tag{19}$$

$$\text{Rearranging, } \boxed{\hat{b}^j = \frac{1}{N + \frac{\hat{\alpha}^j}{s_b}} \left(\sum_{i=1}^{N} \left(\hat{a}^j\mathbf{y}_i^j - \hat{\mathbf{y}}_i\right)^\intercal \mathbf{1} + \frac{\hat{\alpha}^j\mu_b}{s_b}\right)}$$

$$\frac{\partial \ln P(\mathcal{D}|\boldsymbol{\theta})}{\partial \alpha^j} = 0 \implies \frac{-N}{2\hat{\alpha}^j} + \frac{1}{2(\hat{\alpha}^j)^2} \sum_{i=1}^{N} \|\mathbf{y}_i^j - \hat{a}^j(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\|_2^2 = 0$$

$$\therefore \boxed{\hat{\alpha}^j = \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{y}_i^j - \hat{a}^j(\hat{\mathbf{y}}_i + \hat{b}^j\mathbf{1})\|_2^2} \tag{20}$$

Since not all workers may annotate all songs, the indices $N$ (number of songs) and $R$ (number of annotators) are replaced with the set of songs $\mathcal{R}_j$ for which the $j^{\text{th}}$ annotator provided the response and the set of annotators $\mathcal{A}_i$ providing the response for the $i^{\text{th}}$ song, respectively, resulting in Eq. (3).

# References


[1] Brinker Bd, Dinther Rv, Skowronek J. Expressed music mood classification compared with valence and arousal ratings. *EURASIP Journal on Audio, Speech, and Music Processing*, 2012. **2012**(1):24. URL https://doi.org/10.1186/1687-4722-2012-24.

[2] Hu X. Music and mood: Where theory and reality meet. In: *Proc. of the* $5^{th}$ *iConference*. 2010 URL http://hdl.handle.net/2142/14956.

[3] Chen YA, Yang YH, Wang JC, Chen H. The AMG1608 dataset for music emotion recognition. In: *Proc. of the IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015 pp. 693–697. URL https://doi.org/10.1109/ICASSP.2015.7178058.

[4] Chin YH, Jia-Ching W, Wang JC, Yang YH. Predicting the probability density function of music emotion using emotion space mapping. *IEEE Trans. on Affective Computing*, 2018. **9**(4):541–549. URL https://doi.org/10.1109/TAFFC.2016.2628794.

[5] Kumar N, Guha T, Huang CW, Vaz C, Narayanan SS. Novel affective features for multiscale prediction of emotion in music. In: *Proc. of the IEEE Intl. Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2016 pp. 1–5. URL https://doi.org/10.1109/MMSP.2016.7813377.

[6] Bo L, Sminchisescu C. Twin Gaussian processes for structured prediction. *Intl. Jour. of Computer Vision*, 2009. **87**(1):28. URL https://doi.org/10.1007/s11263-008-0204-y.

[7] Chapaneri S, Jayaswal D. Structured prediction of music mood with twin Gaussian processes. In: *Proc. of the Intl. Conf. on Pattern Recognition and Machine Intelligence (PReMI)*. Springer, 2017 pp. 647–654. URL https://doi.org/10.1007/978-3-319-69900-4_82.

[8] Fukuyama S, Goto M. Music emotion recognition with adaptive aggregation of Gaussian process regressors. In: *Proc. of the IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016 pp. 71–75. URL https://doi.org/10.1109/ICASSP.2016.7471639.

[9] Wang JC, Yang YH, Wang HM, Jeng SK. Modeling the affective content of music with a Gaussian mixture model. *IEEE Trans. on Affective Computing*, 2015. **6**(1):56–68. URL https://doi.org/10.1109/TAFFC.2015.2397457.

[10] Wang JC, Wang HM, Lanckriet G. A histogram density modeling approach to music emotion recognition. In: *Proc. of the IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015 pp. 698–702. URL https://doi.org/10.1109/ICASSP.2015.7178059.

[11] Liu Y, Liu Y, Zhao Y, Hua KA. What strikes the strings of your heart?–Feature mining for music emotion analysis. *IEEE Trans. on Affective Computing*, 2015. **6**(3):247–260. URL https://doi.org/10.1109/TAFFC.2015.2396151.

[12] Panda R, Malheiro RM, Paiva RP. Novel audio features for music emotion recognition. *IEEE Trans. on Affective Computing*, 2018. (1):1–1. URL http://doi.ieeecomputersociety.org/10.1109/TAFFC.2018.2820691.

[13] Yang YH, Chen HH. Machine recognition of music emotion: A review. *ACM Trans. on Intelligent Systems and Technology (TIST)*, 2012. **3**(3):40. URL https://doi.org/10.1145/2168752.2168754.

[14] Raykar VC, Yu S, Zhao LH, Valadez GH, Florin C, Bogoni L, Moy L. Learning from crowds. *Journal of Machine Learning Research*, 2010. **11**(Apr):1297–1322. URL http://www.jmlr.org/papers/v11/raykar10a.html.

[15] Raykar VC, Yu S. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *Journal of Machine Learning Research*, 2012. **13**(Feb):491–518. URL http://www.jmlr.org/papers/v13/raykar12a.html.

[16] Chatterjee S, Mukhopadhyay A, Bhattacharyya M. A review of judgment analysis algorithms for crowdsourced opinions. *IEEE Transactions on Knowledge and Data Engineering*, 2019. URL https://doi.org/10.1109/TKDE.2019.2904064.

[17] Li Y, Gao J, Meng C, Li Q, Su L, Zhao B, Fan W, Han J. A survey on truth discovery. *ACM SIGKDD Explorations Newsletter*, 2016. **17**(2):1–16. URL https://www.kdd.org/exploration_files/Article1_17_2.pdf.

[18] Wan M, Chen X, Kaplan L, Han J, Gao J, Zhao B. From truth discovery to trustworthy opinion discovery: An uncertainty-aware quantitative modeling approach. In: *Proc. of the ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*. ACM, 2016 pp. 1885–1894. URL https://doi.org/10.1145/2939672.2939837.

[19] Ramakrishna A, Gupta R, Grossman RB, Narayanan SS. An Expectation Maximization approach to joint modeling of multidimensional ratings derived from multiple annotators. In: *InterSpeech*. 2016 pp. 1555–1559. URL http://dx.doi.org/10.21437/Interspeech.2016-270.

[20] Xiao H, Xiao H, Eckert C. Learning from multiple observers with unknown expertise. In: *Proc. of the Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD)*, volume 7818. Springer, 2013 pp. 595–606. URL https://doi.org/10.1007/978-3-642-37453-1_49.

[21] Markov K, Matsui T. Music genre and emotion recognition using Gaussian processes. *IEEE Access*, 2014. **2**:688–697. URL https://doi.org/10.1109/ACCESS.2014.2333095.

[22] Zhang JL, Huang XL, Yang LF, Xu Y, Sun ST. Feature selection and feature learning in arousal dimension of music emotion by using shrinkage methods. *Multimedia Systems*, 2017. **23**(2):251–264. URL https://doi.org/10.1007/s00530-015-0489-y.

[23] Hu X, Yang YH. Cross-dataset and cross-cultural music mood prediction: A case on Western and Chinese pop songs. *IEEE Trans. on Affective Computing*, 2017. **8**(2):228–240. URL https://doi.org/10.1109/TAFFC.2016.2523503.

[24] Liu T, Han L, Ma L, Guo D. Audio-based deep music emotion recognition. In: *AIP Conference Proceedings*, volume 1967. 2018 p. 040021. URL https://doi.org/10.1063/1.5039095.

[25] Tripathi S, Acharya S, Sharma RD, Mittal S, Bhattacharya S. Using deep and convolutional neural networks for accurate emotion classification on DEAP dataset. In: *Proc. of Innovative Applications of Artificial Intelligence*. 2017 pp. 4746–4752. URL https://aaai.org/ocs/index.php/IAAI/IAAI17/paper/view/15007/13731.

[26] Ni Y, McVicar M, Santos-Rodriguez R, De Bie T. An end-to-end machine learning system for harmonic analysis of music. *IEEE Trans. on Audio, Speech and Language Processing*, 2012. **20**(6):1771–1783. URL https://doi.org/10.1109/TASL.2012.2188516.

[27] Cheng HT, Yang YH, Lin YC, Liao IB, Chen HH, et al. Automatic chord recognition for music classification and retrieval. In: *Proc. of the IEEE Intl. Conf. on Multimedia and Expo (ICME)*. IEEE, 2008 pp. 1505–1508. URL http://doi.ieeecomputersociety.org/10.1109/ICME.2008.4607732.

[28] Yu Y, Zimmermann R, Wang Y, Oria V. Scalable content-based music retrieval using chord progression histogram and tree-structure LSH. *IEEE Trans. on Multimedia*, 2013. **15**(8):1969–1981. URL https://doi.org/10.1109/TMM.2013.2269313.

[29] Williams CK, Rasmussen CE. *Gaussian processes for machine learning*. The MIT Press, 2006. URL http://www.gaussianprocess.org/gpml/.

[30] Elhoseiny M, Elgammal A. Generalized twin Gaussian processes using Sharma–Mittal divergence. *Machine Learning*, 2015. **100**(2-3):399–424. URL https://doi.org/10.1007/s10994-015-5497-9.

[31] Yamada M, Sigal L, Chang Y. Domain adaptation for structured regression. *Intl. Jour. of Computer Vision*, 2014. **109**(1-2):126–145. URL https://doi.org/10.1007/s11263-013-0689-x.

[32] Aljanaki A, Yang YH, Soleymani M. Developing a benchmark for emotional analysis of music. *PloS one*, 2017. **12**(3):e0173392. URL https://doi.org/10.1371/journal.pone.0173392.

[33] Rousseeuw PJ, Driessen KV. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 1999. **41**(3):212–223. URL https://www.tandfonline.com/doi/abs/10.1080/00401706.1999.10485670.

[34] Pasternack J, Roth D. Knowing what to believe (when you already know something). In: *Proceedings of the* $23^{rd}$ *International Conference on Computational Linguistics*. Association for Computational Linguistics, 2010 pp. 877–885. URL https://dl.acm.org/citation.cfm?id=1873880.

[35] Lartillot O, Toiviainen P, Eerola T. A Matlab toolbox for music information retrieval. In: *Data analysis, machine learning and applications: Studies in classification, data analysis, and knowledge organization*. Springer, 2008 pp. 261–268. URL https://doi.org/10.1007/978-3-540-78246-9_31.

[36] Bishop CM. *Pattern recognition and machine learning*. Springer, 2006. URL https://www.springer.com/in/book/9780387310732.

[37] Müller M. *Fundamentals of music processing: Audio, analysis, algorithms, applications*. Springer, 2015. URL www.music-processing.de.

[38] Cho YH, Lim H, Kim DW, Lee IK. Music emotion recognition using chord progressions. In: *Proc. of the IEEE Intl. Conf. on Systems, Man, and Cybernetics (SMC)*. IEEE, 2016 pp. 2588–2593. URL https://doi.org/10.1109/SMC.2016.7844628.

[39] Ellis DP, Weller AV. The 2010 LABROSA chord recognition system. 2010. URL https://doi.org/10.7916/D8TT5193.

[40] Harte C, Sandler M. Automatic chord identifcation using a quantised chromagram. In: *Proc. of the* $118^{th}$ *Audio Engineering Society Convention*. AES, 2005 URL http://www.aes.org/e-lib/browse.cfm?elib=13128.

[41] Zhang D, Zhou ZH, Chen S. Adaptive kernel Principal Component Analysis with unsupervised learning of kernels. In: *Proc. of the IEEE Intl. Conf. on Data Mining (ICDM)*. IEEE, 2006 pp. 1178–1182. URL https://doi.org/10.1109/ICDM.2006.14.

[42] Han J, Pei J, Kamber M. *Data mining: Concepts and techniques*. Elsevier, 2011. URL https://www.sciencedirect.com/book/9780123814791/data-mining-concepts-and-techniques.