



Office of Graduate Studies

Dissertation/Thesis Approval Form

This form is for use by all doctoral and master's students with a dissertation/thesis requirement. Please print clearly as the library will bind a copy of this form with each copy of the dissertation/thesis. All doctoral dissertations must conform to university format requirements, which is the responsibility of the student and supervising professor. Students should obtain a copy of the Thesis Manual located on the library website.

Dissertation/Thesis Title: Modeling and Predicting Emotion in Music

Author: Erik M. Schmidt

This dissertation/thesis is hereby accepted and approved.

Signatures:

Examining Committee

Chair

Steven Weber

Members

Youngmoo Kim

John Walsh

Ben Taskar

Dan Ellis

Academic Advisor

Youngmoo Kim

Department Head

Moshe Kam

Modeling and Predicting Emotion in Music

A Thesis

Submitted to the Faculty

of

Drexel University

by

Erik M. Schmidt

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy

September 2012

© Copyright 2012
Erik M. Schmidt.

This work is licensed under the terms of the Creative Commons Attribution-ShareAlike
license Version 3.0. The license is available at
<http://creativecommons.org/licenses/by-sa/3.0/>.

Acknowledgments

First and foremost, I would like to thank my advisor, Professor Youngmoo Kim, as without his mentoring and support this work would not be possible. As an instructor, Youngmoo put unparalleled care into creating course content and offered some of the most engaging courses of my academic career. As a mentor in the Music and Entertainment Technology Laboratory (MET-lab), Youngmoo created a dynamic and interdisciplinary work environment at Drexel, and also one of mutual respect and admiration. I had the opportunity to be one of Youngmoo's first students here and I feel that I am very fortunate to have been given the opportunity to work with him.

I would also like to thank my thesis committee, who also contributed a great deal to the direction of this work. From Drexel, I thank Steven Weber for chairing the committee and John Walsh for many helpful discussions and constructive criticism. I am also extremely grateful to have had Ben Taskar from the University of Pennsylvania and Dan Ellis from Columbia University as part of my committee. I would specifically like to thank Ben Taskar for making himself accessible for numerous meetings and inspiring many of the approaches used in this work.

I would also like to thank Professor Robert Yantorno from my undergraduate days at Temple University for inspiring my interest in signal processing and motivating me to continue on to graduate school. At Temple, I would also like to thank Professors Li Bai, John Helferty, Frank Higgins, Iyad Obeid, and Dennis Silage.

Additionally, I would like to thank John Anastasio from Hunterdon Central High School for providing my first exposure to electrical engineering in the classroom. The title of his course was "Communication Technology," and I still remember the first day of the course when he showed us the little warning sticker on the back of a VCR stating that it should not be opened. In his course, he declared, that sticker would mean open me up and have a look inside. His course may in fact been the only course of my entire high school education that I got anything out of, though I am quite convinced that he got through to me.

I would also like to thank the many students and collaborators over the years that have contributed to making this work possible. I would like to thank Travis Doll, who began his work in the MET-lab a year before I began mine, and provided constant help and support as I got started. I would also like to thank Raymond Migneco, who started in the lab at the same time as me and was a collaborator on many projects in courses as well as research. I also thank my many other collaborators: Alyssa Batula, Brian Dolhansky, David Grunberg, Brandon Morton, Matthew Prockup, Patrick Richardson, Jeffrey Scott, Jacquelin Speck and Douglas Turnbull.

I would also like to thank Kaycie Hopkins, Jeffrey Scott, Dave Rosen, and David Grunberg for critical reading of the manuscript.

Finally, I would like to acknowledge my family, to whom I am profoundly grateful, as I would not be where I am today without their support. First, I must thank my girlfriend, Kaycie Hopkins, who became a part of my family early in the process of pursuing this degree and has been a constant source of love and support throughout, for which I am most grateful. I would like to thank my parents for their support throughout this process. My greatest debt of gratitude is owed to my father for his limitless support throughout my undergraduate education and for pushing me from a very young age to take my education as far as possible. With the culmination of this degree, it is clear that those those words most certainly resonated. I also must acknowledge that my father as both a musician and an engineer himself, provided the inspiration that led me to choose the career path that I continue to follow today. None of this would be possible without the love and support of my family and because of that, this thesis is dedicated to them.

Table of Contents

LIST OF TABLES	x
LIST OF FIGURES	xii
ABSTRACT	xv
1. INTRODUCTION	1
1.0.1 Thesis Contributions	2
1.1 Overview	6
2. BACKGROUND	7
2.1 Psychology Research on Emotion	7
2.1.1 Perceptual Considerations	7
2.1.2 Perception of Emotion Across Cultures	8
2.1.3 Perception of Emotion and Music Theory	8
2.1.4 Representations of Emotion	9
2.2 Framework for Emotion Recognition	11
2.3 Human Annotation	12
2.3.1 Annotation Games	13
2.3.2 Mechanical Turk	14
2.4 Content-Based Audio Analysis	15
2.4.1 Acoustic Features	15
2.4.2 Audio-Based Musical Mood Detection Systems	17
2.4.3 Emotion Recognition Over Time	20
3. DATA COLLECTION	22
3.1 MoodSwings	23
3.1.1 Summary of Data Collection	24
3.2 MoodSwings Lite Corpus	24

3.2.1	Classification Corpus	26
3.2.2	Emotion Regression Corpus	27
3.2.3	Emotion Distribution Regression Corpus	27
3.3	MoodSwings Turk Corpus	28
3.3.1	Mechanical Turk Data Collection	29
3.3.2	MoodSwings Analysis with MTurk Data	29
3.3.3	Label Sequence Statistical Analysis	31
3.3.4	Emotion Space Heatmaps	31
3.3.5	MTurk Discussion	32
3.4	Instrumental Data Corpus	33
3.4.1	Data Collection	33
3.4.2	Comparison to MoodSwings Pop Music Corpus	34
4.	ACOUSTIC FEATURE SELECTION	35
4.1	Data Collection	36
4.1.1	Song Clip Pair Selection	36
4.1.2	Mechanical Turk Annotation Task	37
4.2	Experiments and Results	37
4.2.1	Perceptual Evaluation of Acoustic Features	38
4.2.2	Relationships Between Musical Attributes and Emotional Affect	39
4.2.3	Identifying Informative Feature Domains	41
4.3	Discussion	42
5.	ACOUSTIC FEATURE LEARNING	44
5.1	Deep Belief Networks	45
5.1.1	Restricted Boltzman Machines	46
5.1.2	Constructing a Multi-Layer Network	47
5.1.3	Supervised Finetuning	48
5.2	Generating Features with DBNs	48

6. MACHINE LEARNING	50
6.1 Classification Methods	50
6.1.1 Support Vector Machines	51
6.1.2 Classifier Decision Level Fusion	52
6.2 Regression Methods	53
6.2.1 Least-Squares Regression	53
6.2.2 Support Vector Regression	54
6.2.3 Regression Decision-Level Fusion	55
6.3 A Kalman Filtering Approach	56
6.3.1 Estimating model parameters	56
6.3.2 Making Predictions using Kalman Smoothing	58
6.4 Conditional Random Fields	58
6.4.1 Definition	58
6.4.2 Arousal-Valence Heatmaps	59
7. MOODSWINGS LITE EXPERIMENTS	60
7.1 Music Emotion Classification	62
7.1.1 Single Feature Classification	62
7.1.2 Decision Fusion Classification	62
7.2 Music Emotion Regression	63
7.2.1 Single Feature Regression	64
7.2.2 Regressor Fusion	65
7.2.3 Test Data Projections	66
7.3 Emotion Regression Over Time	67
7.3.1 Test Data Projections	68
7.3.2 MoodSwings “AI”	68
7.4 Emotion Distribution Prediction	69
7.4.1 Single Feature Emotion Distribution Prediction	70

7.4.2	Emotion Distribution Feature Fusion	72
7.5	Time-Varying Emotion Distribution Prediction	74
7.6	A Kalman Filtering Approach	76
7.6.1	Kalman Data Preprocessing	76
7.6.2	Multiple Linear Regression	78
7.6.3	Kalman Filtering	78
7.6.4	Kalman Filter Mixtures	79
7.7	Emotion Regression with Learned Features	80
7.7.1	DBN Training	81
7.7.2	DBN Features in Emotion Prediction	81
7.7.3	Relating DBN Features to Frequency Content	82
7.8	Discussion	84
8.	MOODSWINGS TURK EXPERIMENTS	87
8.1	Emotion Prediction with MTurk Data	88
8.2	Emotion Prediction with Conditional Random Fields	89
8.2.1	A CRF Model for A-V Emotion Prediction	90
8.2.2	Applying the Model to MoodSwings Turk	92
8.3	Feature Learning Experiments	96
8.3.1	Short-Time Feature Learning	96
8.3.2	Multi-Frame Feature Learning	97
8.3.3	Universal Background Model Feature Learning	99
8.4	Conditional Random Fields with Learned Features	100
8.4.1	DBN Single-Frame Features	100
8.4.2	DBN Multi-Frame Features	101
8.5	Discussion	102
9.	INSTRUMENTAL DATASET EXPERIMENTS	105
9.1	Predicting A-V Gaussian Distributions	105

9.2	Conditional Random Fields	107
9.3	Discussion	108
10.	CONCLUSIONS AND FUTURE DIRECTIONS	110
10.1	Overall Conclusions	111
10.2	Future Directions	112
	APPENDIX A: ACOUSTIC FEATURE EXTRACTION	115
A.1	Mel-Frequency Cepstral Coefficients	115
A.1.1	Mel-Spaced Triangular Filters	116
A.2	Autocorrelation of chroma	117
A.3	Spectral Shape Features	117
A.3.1	Spectral Centroid	117
A.3.2	Spectral Flux	118
A.3.3	Spectral Rolloff	118
A.3.4	Spectral Flatness	118
A.4	Octave-Based Spectral Contrast	118
A.5	Echo Nest Timbre	120
	APPENDIX B: RESTRICTED BOLTZMAN MACHINE TUTORIAL	122
B.1	Energy-Based Models	122
B.2	Hidden Variables	122
B.3	Binomial Hidden Units	124
B.4	Sampling in RBMs	125
B.5	Training RBMs	126
	APPENDIX C: SUPPORT VECTOR MACHINE TUTORIAL	127
C.1	Perceptron	127
C.2	Obtaining a Unique Solution	128
C.3	Kernel Methods	130
	APPENDIX D: KALMAN/RAUCH-TUNG-STRIEBEL ESTIMATION	133

APPENDIX E: CONDITIONAL RANDOM FIELDS TUTORIAL	134
E.1 Definition	134
E.2 Estimating CRF Parameters	135
E.3 Inference	136
BIBLIOGRAPHY	137

List of Tables

2.1	Mood adjectives used in the MIREX Audio Mood Classification task.	10
2.2	Common acoustic feature types for emotion classification.	16
3.1	Quadrant-based class assignments of all MoodSwings Lite music clips.	26
3.2	Number of MTurk worker annotations for each clip before and after filtering.	29
3.3	Statistics of ground truth squared correlation coefficient (r^2) from one second to the next and from the first second to the last.	31
3.4	Statistics of ground truth squared correlation coefficient (r^2) from one second to the next and from the first second to the last.	34
4.1	Normalized difference error between the arousal/valence ratings for the reconstructions and the originals.	39
4.2	Percentage of paired comparisons that yielded the desired perceptual result for mode and tempo.	40
4.3	Normalized feature change with respect to musical mode and tempo alterations.	42
7.1	Single feature classification.	62
7.2	Classifier fusion results.	63
7.3	Regression results for fifteen second clips.	65
7.4	Time-varying regression results.	68
7.5	Distribution regression results for fifteen second clips.	71
7.6	Distribution regression results for short-time (one-second) A-V labels.	75
7.7	Results for emotion distribution prediction over time.	78
7.8	Results for emotion distribution prediction over time.	82
8.1	MLR results for short-time (one-second) A-V labels, repeating the experiments of Table 7.6 with labels collected via MTurk.	89
8.2	Computing time analysis for CRF training on each cross-validation set.	93
8.3	Emotion prediction results for conditional random fields (CRF) trained on sequence examples as well as independent examples (CRF-I). Multiple linear regression (MLR) results are provided as a baseline.	94

8.4	Emotion regression results for fifteen second clips. DBN-SF are features learned from single frames (SF), DBN-MF are features learned from multi-frame (MF) aggregations, and DBN-UBM are features learned with a universal background model (UBM) approach to DBN pretraining. KL-divergence is not applicable to model error.	98
8.5	Time-varying music emotion prediction using conditional random fields. DBN-SF are features learned from single frames (SF) and DBN-UBM are features learned with a universal background model (UBM) approach to DBN pretraining.	101
9.1	MLR results for short-time (one-second) A-V labels, repeating the experiments of Table 7.6 with labels collected via MTurk.	106
9.2	Emotion prediction results for conditional random fields (CRF) trained on sequence examples as well as independent examples (CRF-I). Multiple linear regression (MLR) results are provided as a baseline.	108

List of Figures

1.1	An overall summary of music emotion recognition.	3
2.1	The Valence-Arousal space, labeled by Russell’s direct circular projection of adjectives. Includes semantic of projected third affect dimensions: “tension,” “kinetics,” “dominance.”	11
2.2	Overall model of emotion classification systems	13
3.1	The MoodSwings gameboard.	23
3.2	A contour plot showing the distribution of the MoodSwings labels.	24
3.3	Progression of valence-arousal labels over an example clip. The ellipses represent the standard deviation across different players.	25
3.4	A contour plot showing the distribution of all MoodSwings Lite A-V labels.	25
3.5	Data collected from MoodSwings over a 15-second clip.	26
3.6	Modeling emotion space ground truth as a single A-V value. Left plot: modeling the clip as a single example. Right plot: modeling the clip as a sequence of fifteen examples (markers become darker as time advances). Both plots display the collected from MoodSwings (gray ●) and the ground truth representation (red ●).	27
3.7	Modeling emotion space ground truth as a stochastic distribution. Left plot: modeling the clip as a single example. Right plot: modeling the clip as a sequence of fifteen examples (markers become darker as time advances). Both plots display the collected from MoodSwings (gray ●) and the ground truth representation (red).	28
3.8	A-V distribution of data shown as a contour map. MoodSwings (left) and MTurk (right).	30
3.9	An emotion space heatmap.	32
3.10	A contour plot showing the A-V distribution of the instrumental dataset.	34
4.1	Histograms of normalized difference error between the arousal/valence ratings for the reconstructions and the originals.	40
5.1	MIREX mood classification task performance by year.	45
5.2	Restricted Boltzman machine topology.	46
5.3	Greedy stacking of restricted Boltzman machines.	47
5.4	Feature learning system architecture showing the temporal aggregation, deep belief network and subsequent training of linear regressors to predict multi-dimensional A-V distributions.	49

6.1	An example of a maximum margin classifier. In this example x_1 and x_2 are the dimensions of the input data, each blue X represents an example in the negative class and each red O represents an example in the positive class.	51
6.2	Ensemble-based decision-fusion system of differing acoustic feature types.	53
6.3	Multi-level regression topologies.	56
6.4	Heatmap visualization of CRF transition probabilities. Actual discretization is 11×11. .	59
7.1	Collected A-V labels and projections resulting from regression analysis. A-V labels: second-by-second labels per song (gray ●), μ of collected labels (red ●), σ of collected labels (red ellipse). Projections: least-squares spectral contrast projection (green +), SVR MFCC projection (blue O), least-squares multi-level combined (black X).	66
7.2	Window length analysis for different acoustic features.	67
7.3	A-V labels and projections over time for eight example 15-second music clips (markers become darker as time advances): second-by-second labels per song (gray ●), mean of the collected labels over 1-second intervals (red ●), and projection from acoustic features in 1-second intervals (blue X).	69
7.4	Collected A-V labels and distribution projections resulting from regression analysis. A-V labels: second-by-second labels per song (gray ●), Σ of collected labels (solid red ellipse), Σ of MLR projection from spectral contrast features (dash-dot blue ellipse), Σ of MLR Multi-Level combined projection (dashed green ellipse).	72
7.5	Multi-layer regression topologies.	73
7.6	Window length analysis for different acoustic features.	74
7.7	Time-varying emotion distribution regression results for three example 15-second music clips (markers become darker as time advances): second-by-second labels per song (gray ●), Σ of the collected labels over 1-second intervals (red ellipse), and Σ of the distribution projected from acoustic features in 1-second intervals (blue ellipse).	75
7.8	Emotion label preprocessing: gray dots indicate individual second-by-second labels collected from the MoodSwings game. Red ellipses are the estimates of the distribution; both become darker as time progresses.	77
7.9	Emotion label preprocessing: gray dots indicate individual second-by-second labels collected from the MoodSwings game, red ellipses indicate the estimates of the distribution, and blue ellipses indicate the predictions using Kalman filter mixtures; all three become darker as time progresses.	79
7.10	Log view of DBN magnitude spectra input.	83
7.11	DBN hidden layer outputs.	83
7.12	Log view of DBN magnitude spectra reconstruction.	84
8.1	Heatmap visualization of CRF transition probabilities. Actual discretization is 11×11. .	91

8.2	Emotion space heatmap prediction using conditional random fields. Shown is the predicted emotion from the beginning of the song “Something About You,” by Boston. These figures demonstrate the system tracking the emotion through the low-energy, negative-emotion introduction, and through the transition at second 29 into a high-energy, positive emotion rock verse. In these figures, red indicates the highest density and blue is the lowest.	96
8.3	Log-magnitude spectrogram of input audio.	98
8.4	DBN hidden layer outputs using the aggregate spectral frames as input.	99
8.5	Reconstruction of the original aggregated spectrogram used as the DBN input. Top is last one second aggregates, middle is last 2 seconds, bottom is last 4 seconds.	100
9.1	Time-varying A-V emotion distribution predictions of instrumental data using the M.L. combined method (see Section 7.5). Shown in gray are the individual user ratings, red ellipses are the estimated distribution of the collected data, and blue ellipses are the predicted distributions, all of which get darker over time.	107
A.1	Forty band mel-warped triangular filterbank	116
A.2	Graphical depiction of SSDs.	119
A.3	EchoNest timbre feature basis functions.	121
B.1	Restricted Boltzman machine topology.	123
B.2	Gibbs sampling in restricted Boltzman machines In this figure $v^{(0)}$ refers to all visible nodes and $h^{(0)}$ refers to all hidden nodes at time zero.	125
C.1	An example of a maximum margin classifier. In this example x_1 and x_2 are the dimensions of the input data, each blue X represents an example in the negative class, and each red O represents an example in the positive class.	127
C.2	Four valid perceptron classification boundaries that satisfy the training criteria described in Equation C.2.	128
C.3	Simple kernel projection example	131
C.4	SVM with a third order polynomial kernel (left) and a radial basis kernel (right). The two classes of data are denoted with X and O. The classification hyperplane is denoted as 0 with parallel transports +1 and -1. Examples that are support vectors are marked with a square \square	132

Abstract

Modeling and Predicting Emotion in Music

Erik M. Schmidt

Youngmoo E. Kim, Ph.D.

With the explosion of vast and easily-accessible digital music libraries over the past decade, there has been a rapid expansion of research towards automated systems for searching and organizing music and related data. Online retailers now offer vast collections of music, spanning tens of millions of songs, available for immediate download. While these online stores present a drastically different dynamic than the record stores of the past, consumers still arrive with the same requests—recommendation of music that is similar to their tastes; for both recommendation and curation, the vast digital music libraries of today necessarily require powerful automated tools.

The medium of music has evolved specifically for the expression of emotions, and it is natural for us to organize music in terms of its emotional associations. But while such organization is a natural process for humans, quantifying it empirically proves to be a very difficult task. Myriad features, such as harmony, timbre, interpretation, and lyrics affect emotion, and the mood of a piece may also change over its duration. Furthermore, in developing automated systems to organize music in terms of emotional content, we are faced with a problem that oftentimes lacks a well-defined answer; there may be considerable disagreement regarding the perception and interpretation of the emotions of a song or even ambiguity within the piece itself.

Automatic identification of musical mood is a topic still in its early stages, though it has received increasing attention in recent years. Such work offers potential not just to revolutionize how we buy and listen to our music, but to provide deeper insight into the understanding of human emotions in general. This work seeks to relate core concepts from psychology to that of signal processing to understand how to extract information relevant to musical emotion from an acoustic signal. The methods discussed here survey existing features using psychology studies and develop new features using basis functions learned directly from magnitude spectra. Furthermore, this work presents a

wide breadth of approaches in developing functional mappings between acoustic data and emotion space parameters. Using these models, a framework is constructed for content-based modeling and prediction of musical emotion.

Chapter 1: Introduction

With the explosion of vast and easily-accessible digital music libraries over the past decade, there has been a rapid expansion of research towards automated systems for searching and organizing music and related data. Along with the advent of portable music devices, such as the now ubiquitous Apple iPod, and so called “smart phones” that double as media players, a vast percentage of consumers are now carrying music libraries spanning thousands of songs in their pockets. Following the content demands these devices have created, music retail has seen a revolution of its own, as the sale of music has steadily moved away from the physical media of the past. Online retailers such as Amazon and Apple’s iTunes Store now offer vast collections of music, spanning over fourteen million songs, available for immediate download. While these online stores present a drastically different dynamic than the record stores of the past, consumers still arrive with the same requests—recommendation of music that is similar to their tastes. Oftentimes these are straightforward problems, such as requesting music of a specific artist, genre, or style, but many times consumers pose more challenging questions, such as music that reflects a specific mood or an artist that is similar to one they already like. In the record stores of the past, which offered only a few hundred titles, a well educated salesman could easily address most requests, however the vast digital music libraries of today necessarily require powerful automated tools.

The field of Music Information Retrieval (Music-IR) consists of a multi-disciplinary community of researchers from signal processing and machine learning backgrounds that seek to meet this demand with tools driven entirely by computer-based audio analysis and machine learning [1]. While music classification problems often seem straightforward when presented to humans, the development of such computational systems is far from trivial, with machines typically presenting myriad difficulties where humans find intuitive solutions. The first era of Music-IR research focused on problems of concrete ground truth. These systems were developed to perform tasks such as automatic artist identification, chord detection and cover song detection. Following reasonable progress in these areas,

Music-IR has moved on to tasks involving a more ambiguous ground-truth representation. Such tasks include loosely defined labels, such as musical genre (i.e. jazz, blues, rock), artist/album/song similarity, and human perceptual labeling, such as emotion or expression. In developing automated tools to organize music by its emotional content or mood, we are presented with a problem that is not only difficult for machines to solve, but which also does not have a well-defined answer. Such systems necessarily require human labeled observational data and are therefore only realizable through supervised machine learning.

Music is oftentimes referred to as a “language of emotion,” [2] and it is natural for us to categorize music in terms of its emotional associations. The potential impact of automatic systems for emotion-based music retrieval is far-reaching, offering potential not just to revolutionize how we buy and listen to our music, but to provide deeper insight into the understanding of human emotions. For instance, a fully trained model capable of relating acoustic content to emotion space parameters could potentially provide new insight into how human emotions are derived. At the same time, an end user application of this work could be a simple, easy-to-use retrieval interface that could be operated without any knowledge of the underlying system. Such a system could provide the ability to simply request “happy songs” while feeling unpleasant sitting in traffic, or perhaps to request songs that start out as “unpleasant” but over time become “happy.” While these goals are exciting, music emotion recognition is a highly non-trivial problem with research still in the early stages. The development of such content-based systems for musical emotion detection can be broken into three main areas: collecting and analyzing human labels of emotion in music, developing signal processing algorithms to extract informative acoustic feature data, and developing machine learning techniques to facilitate machine understanding of such data [3].

1.0.1 Thesis Contributions

This dissertation has a wide breadth of contributions, spanning psychology, signal processing, and machine learning. A variety of factors contribute to a person’s perception of musical mood, and therefore some variation and disagreement between user ratings of the same content is expected. Building this work on top of a psychology framework, human emotion responses to music are repre-

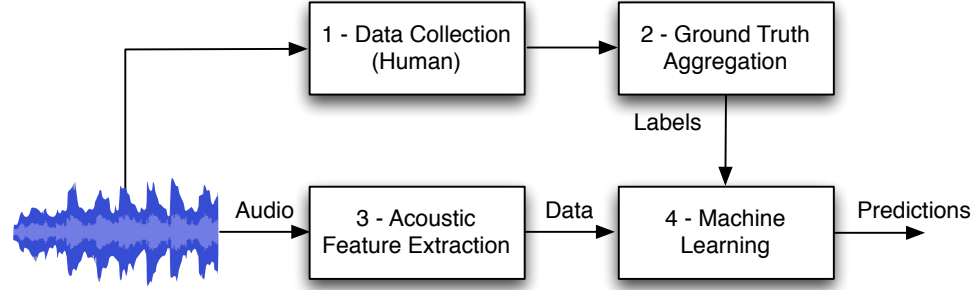


Figure 1.1: An overall summary of music emotion recognition.

sented with a two-dimensional parametric representation using the arousal-valence (A-V) space, a well-known representation of emotion in psychology research [4, 5]. The A-V space was developed to provide a continuous range of human emotion, where valence reflects positive vs. negative emotions and arousal indicates emotional intensity. The overall goal of this work should be seen as understanding human A-V responses to music through content-based (signal processing) techniques. The main contributions of this work are as follows:

1. Collects multiple datasets spanning several genres of music, including a pop music database collected via an online collaborative gaming approach [6], a pop music database crowdsourced through Amazon’s Mechanical Turk (MTurk) [7, 8], and a database that consists entirely of instrumental music.
2. Proposes a range of methods for representing arousal-valence disagreement in human responses to music, including aggregating the data from multiple subjects to a single A-V point [6], representing disagreement in collected data an A-V distribution [9, 10], and using emotion space heatmaps that can represent arbitrary distributions and modes [11].
3. Develops multiple frameworks for acoustic feature identification, including those based on perceptual evaluations of existing domains for a bottom-up style feature selection process [12], as well as methods that learn sets of basis functions directly from magnitude spectra that are informative to musical emotion [7, 13, 14].

4. Develops multiple machine learning approaches for relating high-dimensional acoustic content to low-dimensional A-V representations, including multi-class classification [6], direct A-V regression using a variety of methods [6], regression methods for the automatic parameterization of A-V Gaussian distributions [9, 10], and a final method using conditional random fields (CRFs) that is capable of automatically parameterizing A-V heatmaps over time [11].

As the training of a supervised machine learning system requires human-labeled examples, an online collaborative game called MoodSwings has been developed, which captures A-V labels dynamically (over time) to reflect emotion changes in synchrony with music [15]. This method potentially provides quantitative labels that are well-suited to computational methods, allowing the representation of disagreement in ratings as a *stochastic distribution*. Using initial data from the game, a corpus was collected of 240, 15-second pop music clips, annotated at one second intervals. This work also investigates the use of Amazon’s Mechanical Turk, an online platform for crowdsourcing, which is used to collect a much more densely annotated data collection on the same audio clips. That method validates the effectiveness of the game at collecting quality annotations, but also shows an alternative way to collect data faster. Furthermore, in seeking audio examples that display larger emotion-space dynamics, an additional audio corpus of 300, 30-second clips is constructed. These examples are instrumental music, removing the influence of lyrics, and are hand selected to contain dynamic A-V changes. Due to the success of MTurk on previous collections, it is employed to annotate this final set as well.

In following with the non-trivial nature of the problem, no dominant acoustic feature has yet emerged for representing musical emotion. Noting that there are many factors that contribute to human perception of emotion, in developing digital signal processing algorithms to extract emotion-related audio features, this work focuses on methods that combine multiple feature domains (e.g. loudness, timbre, harmony). In addition, following the dynamic nature of music, the focus has been restricted solely to time-varying features.

This work first provides perceptual studies using acoustic reconstructions of two of the most common features used in Music Information Retrieval: mel-frequency cepstral coefficients (MFCCs)

and the chromagram. The goal of these experiments is to quantitatively validate how informative these features are to music emotion recognition as a motivation for their use. Furthermore, this work also investigates how features could be selected through a bottom-up procedure using knowledge of variance and invariance of acoustic features to musical parameters (e.g., key, mode, tempo). However, while such an approach is promising, it is difficult to assign exact weight to specific musical parameters, and these experiments indicate that there are other sources of information that are much harder to quantify through music notation (e.g., musical expression). In the context of content-based emotion prediction, this work evaluates multiple sets of acoustic features, including psychoacoustic (mel-cepstrum and statistical frequency spectrum descriptors) and music-theoretic (estimated pitch chroma) representations of the labeled audio. While these features demonstrate very reasonable performance on a range of tasks, they do not fully solve the problem. In seeking to identify more informative feature representations, this work proposes methods to learn representations of music audio directly from magnitude spectra that are specifically optimized for the prediction of emotion.

In developing the proper machine learning approach for musical emotion recognition, this work largely focuses on using short-time segments as a means to track emotional changes over time. Human emotion responses to music are dynamic processes that evolve over time in synchrony with the music, so the application of a single, unvarying mood label across an entire song belies this nature. The initial experiments presented here demonstrate that not only does the emotional content change over time, but also that a distribution of (as opposed to singular) ratings is appropriate for even short time slices (down to one second). To perform the mapping from acoustic features to the A-V mood space, a variety of methods for multi-variate regression of distribution parameters are explored, such as multiple linear regression, partial least-squares regression, and support vector regression. The results demonstrate the effectiveness of this system in predicting emotion distributions for 15 second clips, as well as its ability to follow changes in A-V label distributions over time. In focusing further on time-varying emotion prediction, this work develops a systems approach using linear dynamical systems for modeling the evolution of emotion distributions, showing modest performance gains. Furthermore, in looking to move beyond the modeling limitations of the A-V Gaussian distributions,

an approach is developed using conditional random fields (CRFs), which are capable of modeling the temporal evolution of A-V heatmaps, representing arbitrary emotion-space distributions and modes.

1.1 Overview

The following chapters will proceed as follows:

- Chapter 2 will discuss background in data collection, feature extraction, and overall systems for the prediction of musical emotion. This survey spans psychology, signal processing, machine learning, and music information retrieval.
- Chapter 3 will present the multiple datasets and collection methods used in this work.
- Chapter 4 provides perceptual experiments using existing acoustic representations and discusses the overall topic of feature selection.
- Chapter 5 will present methods for learning acoustic representations that are specifically optimized for the prediction of emotion.
- Chapter 6 presents machine learning methods for modeling and predicting A-V emotion.
- Chapters 7 - 9 present the experiments, which are broken up by the various datasets (MoodSwings Lite, MoodSwings Turk, Instrumental Dataset, respectively).
- Chapter 10 provides overall conclusions and future directions.
- Appendix A provides explanations of the various acoustic representations used in this work.
- Appendices B - E present tutorial sections related to restricted Boltzman machines, support vector machines, Kalman filtering, and conditional random fields.

Chapter 2: Background

This chapter surveys the state of the art in content-based musical emotion recognition [15, 16]. The first section is devoted to constructing a theoretical foundation for the understanding of human emotions. Research in cognitive science and psychology is surveyed to develop a framework for how emotion is defined, how emotion is quantified, and what considerations must be made when collecting human emotion data. That section will also investigate psychology studies on music theory and emotion. This chapter next discusses developing a framework for music emotion recognition in general and the different components involved (e.g. human data collection, informative domains for classification). In seeking to develop systems to collect music emotion data, many current systems and approaches are surveyed with focus on online annotation games and Amazon’s Mechanical Turk. Finally, the focus is narrowed to content-based audio analysis, and the last section discusses state of the art work in content-based music emotion recognition.

2.1 Psychology Research on Emotion

Over the past half-century, there have been several important developments spanning multiple approaches for qualifying and quantifying emotions related to music. Such inquiry began well before the widespread availability of music recordings as a means of clinically repeatable musical stimuli (using musical scores), but recordings are the overwhelmingly dominant form of stimulus used in modern research studies of emotion. Although scores can provide a wealth of relevant information, score-reading ability is not universal, and the focus in this chapter and the overall work shall be limited to music experienced through audition.

2.1.1 Perceptual Considerations

When performing any measurement of emotion, from direct biophysical indicators to qualitative self-reports, one must also consider the source of emotion being measured. Many studies, using categorical or scalar/vector measurements, indicate the important distinction between one’s percep-

tion of the emotion(s) *expressed* by music and the emotion(s) *induced* by music [4, 17]. Both the emotional response and its report are subject to confound. Early studies of psychological response to environment, which considered the emotional weight of music both as a focal and distracting stimulus, found affective response to music can also be sensitive to the environment and contexts of listening [18]. Juslin and Luakka, in studying the distinctions between perceptions and inductions of emotion, have demonstrated that *both* can be subject to not only the social context of the listening experience (such as audience and venue), but also personal motivation (e.g. music used for relaxation, stimulation, etc.) [17]. The remainder of this chapter and the overall work will focus on systems that attempt to discern the emotion expressed, rather than induced, by music.

2.1.2 Perception of Emotion Across Cultures

Cross-cultural studies of musical power suggest that there may be universal psychophysical and emotional cues that transcend language and acculturation [19]. Comparisons of tonal characteristics between Western 12-tone and Indian 24-tone music suggest certain universal mood-targeted melodic cues [20]. In a recent ethnomusicology study of people with no exposure to Western music (or culture), Mafa natives of Cameroon, categorized music examples into three categories of emotion in the same way as Westerners [21].

2.1.3 Perception of Emotion and Music Theory

A musical piece is made up of a combination of different attributes such as key, mode, tempo, instrumentation, etc. While none of these attributes fully describes a piece of music, each one contributes to the listener’s perception of the piece. Much previous work has been dedicated to what compositional attributes significantly determine emotion and which parameters are less relevant [22, 23, 24]. Previous work has shown that while these parameters are not the sole contributors to the emotion of the music [25], they do undoubtedly contain some information about the underlying emotion. Information about which musical parameters are important in conveying specific emotions can potentially aid in selecting informative acoustic representations for content-based prediction. For

example, in Chapter 4 it will be investigated whether these compositional building blocks induce changes in the acoustic feature domain.

Mode and tempo have been shown to consistently elicit a change in perceived emotion in user studies. Mode is the selection of notes (scale) that form the basic tonal substance of a composition and tempo is the speed of a composition [26]. Research shows that major modes tend to elicit happier emotional responses, while the inverse is true for minor modes [24, 27, 28, 29]. Tempo also determines a user’s perception of music, with higher tempi generally inducing stronger positive emotions with higher emotional intensity [23, 24, 27, 28, 30].

2.1.4 Representations of Emotion

Music-IR systems tend to use either categorical descriptions or parametric models of emotion for classification or recognition. Each representation is supported by a large body of psychology research.

Categorical Psychometrics

Categorical approaches involve finding and organizing some set of emotional descriptors (tags) based on their relevance to some music in question. One of the earliest studies by Hevner, published in 1936, initially used 66 adjectives, which were then arranged into 8 groups [22]. While the adjectives used and their specific grouping and hierarchy have been scrutinized and even disputed, many categorical studies conducted since Hevner’s indicate such tagging can be intuitive and consistent, regardless of the listener’s musical training [31, 32].

In a recent sequence of music-listening studies, Zenter *et al.* reduced a set of 801 “general” emotional terms into a subset metric of 146 terms specific to music mood rating. Their studies, which involved rating music-specificity of words and testing words in lab and concert settings with casual and genre-aficionado listeners, revealed that the interpretation of these mood words varies between different genres of music [33].

The recent MIREX evaluations for automatic music mood classification have categorized songs into one of five mood clusters, shown in Table 2.1. The five categories were derived by performing

clustering on a co-occurrence matrix of mood labels for popular music from the All Music Guide [34, 35].

Clusters	Mood Adjectives
Cluster 1	passionate, rousing, confident, boisterous, rowdy
Cluster 2	rollicking, cheerful, fun, sweet, amiable/good natured
Cluster 3	literate, poignant, wistful, bitter-sweet, autumnal, brooding
Cluster 4	humorous, silly, campy, quirky, whimsical, witty, wry
Cluster 5	aggressive, fiery, tense/anxious, intense, volatile, visceral

Table 2.1: Mood adjectives used in the MIREX Audio Mood Classification task [35].

Scalar/Dimensional Psychometrics

Other research suggests that mood can be scaled and measured by a continuum of descriptors or simple multi-dimensional metrics. Seminal work by Russell and Thayer in studying dimensions of arousal established a foundation upon which sets of mood descriptors may be organized into low-dimensional models. Most noted is the two-dimensional *valence-arousal* (V-A) space (See Figure 2.1), where emotions exist on a plane along independent axes of arousal (intensity), ranging high-to-low, and valence (an appraisal of polarity), ranging positive-to-negative [4, 5]. The validity of this two-dimensional representation of emotions for a wide range of music has been confirmed in multiple studies [31, 36, 37].

Two very interesting studies have validated the use of V-A by constructing a parametric space via dimensionality reduction techniques from data that had been annotated with text-based labels. Laurier *et al.* investigated a set of 100 emotion specific tag words (e.g. “scary,” “anxious,” “depressing,” etc.) that had been applied to over 61,080 tracks [36]. A two-dimensional distance space was created by the application of self organizing maps and was found to have dimensions corresponding to valence and arousal. Furthermore, in a recent study Liebetrau *et. al* had listeners observe simple major-major and minor-minor chord changes [37]. Participants applied text-based emotion descriptions that they defined themselves, and a two-dimensional emotion space was constructed from this

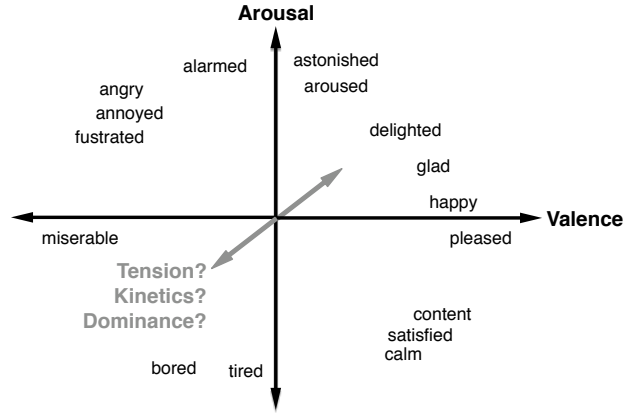


Figure 2.1: The Valence-Arousal space, labeled by Russell’s direct circular projection of adjectives [4]. Includes semantic of projected third affect dimensions: “tension” [38], “kinetics” [39], “dominance” [18].

data via multiple factor analysis. Again, the resulting dimensions were highly representative of V-A, with one dimension explaining the mode of the pair and the degree of harmony between the chords (valence), and the other explaining tempo (arousal).

Some studies have expanded this approach to develop three-dimensional spatial metrics for comparative analysis of musical excerpts, although the semantic nature of the third dimension is subject to speculation and disagreement [40]. Other investigations of the V-A model itself suggest evidence for separate channels of arousal (as originally proposed by Thayer) that are not elements of valence [41].

A related, but categorical, assessment tool for self-reported affect is the Positive and Negative Affect Schedule (PANAS), which asserts that all discrete emotions (and their associated labels) exist as incidences of positive or negative affect, similar to valence [42, 43]. In this case, however, positive and negative are treated as separate categories as opposed to the parametric approach of V-A.

2.2 Framework for Emotion Recognition

Emotion recognition can be viewed as a multiclass-multilabel classification or regression problem, where we try to annotate each music piece with a set of emotions. A *music piece* might be an entire

song, a section of a song (e.g. chorus, verse), a fixed-length clip (e.g. 30-second song snippet), or a short-term segment (e.g. 1 second).

Emotion is represented as either a single multi-dimensional vector or a time-series of vectors over a semantic space of emotions. That is, each dimension of a vector represents a single emotion (e.g. angry) or a bi-polar pair of emotions (e.g. positive/negative). The value of a dimension encodes the strength-of-semantic-association between the piece and the emotion. This is sometimes represented with a binary label to denote the presence or absence of the emotion, but more often represented as a real-valued score (e.g. Likert scale value, probability estimate). Emotion is represented as a time-series of vectors if, for example, the goal is to track changes in emotional content over the duration of a piece.

The values of the emotion vector for a music piece can be estimated in a number of ways using various forms of data (Figure 2.2). The direct method is to ask human listeners to evaluate the relevance of an emotion for a piece (see Section 2.3). This can be done, for example, using a survey, a social tagging mechanism, or an annotation game. It is also possible to analyze forms of contextual meta-data in text form. This may include text-mining web-documents (e.g. artist biographies, album reviews) or a large collection of social tags (referred to as a *tag cloud*), and analyzing lyrics using natural language processing (e.g. sentiment analysis) [44, 45]. Finally, it is also possible to analyze the audio content using both signal processing and supervised machine learning to automatically annotate music pieces with emotions (see Section 2.4.2). Content-based methods can also be used to analyze other related forms of multimedia data such as music videos and promotional photographs [46].

2.3 Human Annotation

A survey is a straightforward technique for collecting information about emotional content in music. All Music Guide has devoted considerable amounts of money, time and human resources to annotate their music databases with high-quality emotion tags. As such, they are unlikely to fully share this data with the Music-IR research community. To remedy this problem, Turnbull *et al.* collected the CAL500 data set of annotated music [47]. This data set contains one song from 500 unique

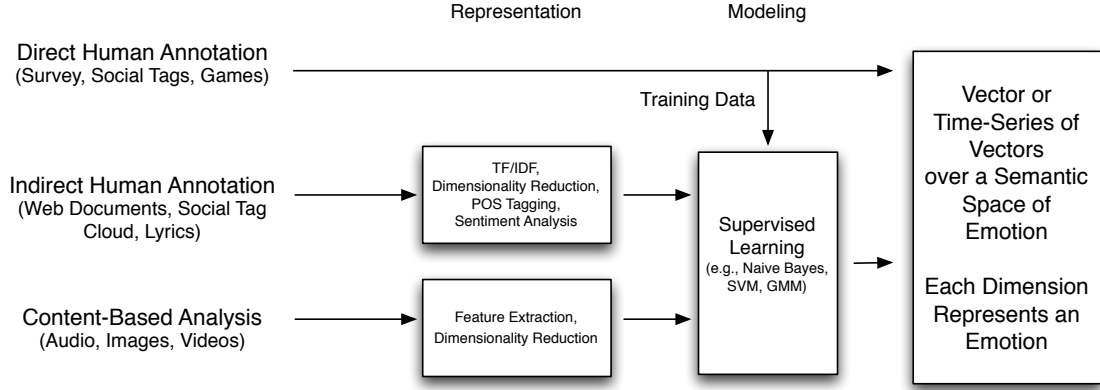


Figure 2.2: Overall model of emotion classification systems [15].

artists, each of which has been manually annotated by a minimum of three non-expert reviewers using a vocabulary of 174 tags, of which 18 relate to different emotions. Trohidis *et al.* have also created a publicly available data set consisting of 593 songs, each of which has been annotated using 6 emotions by 3 expert listeners [48].

A second approach to directly collecting emotion annotations from human listeners involves social tagging. For example, Last.fm is a music discovery website that allows users to contribute *social* tags through a text box in their audio player interface. By the beginning of 2007, their large base of 20 million monthly users had built up an unstructured vocabulary of 960,000 free-text tags and used it to annotate millions of songs [49]. Unlike AMG, Last.fm makes much of this data available to the public through their public APIs. While this data is a useful resource for the Music-IR community, Lamere and Celma point out that there are a number of problems with social tags: sparsity due to the cold-start problem and popularity bias, ad-hoc labeling techniques, multiple spellings of tags, malicious tagging, etc. [50].

2.3.1 Annotation Games

Traditional methods of data collection, such as the hiring of subjects, can be flawed, since labeling tasks are time-consuming, tedious, and expensive [51]. Recently, a significant amount of attention has been placed on the use of collaborative online games to collect such ground truth labels for difficult problems, so-called “Games With a Purpose.” Several such games have been proposed for

the collection of music data, such as *MajorMiner* [52], *Listen Game* [53], and *TagATune* [54]. These implementations have primarily focused on the collection of descriptive labels for a relatively short audio clip.

Herd It combines multiple types of music annotation games, including valence-arousal annotation of clips, descriptive labeling, and music trivia [55].

2.3.2 Mechanical Turk

Mechanical Turk (MTurk) is an online service provided by Amazon to collect human judgements. Using MTurk’s Human Intelligence Tasks (HITs), it is possible to obtain human judgements on almost any task for a small amount of compensation. These tasks are open to anyone on the web and therefore provide the ability to collect large amounts of data over a short period of time.

Who are MTurk Workers

A recent study on the demographics of MTurk workers has indicated that respondents tend to be reasonably well educated and located in the United States more than any other country [56]. The study found that 62.8% of respondents had a bachelor’s degree or higher. Respondents were located in 66 countries, with the highest percentages in the US at 46.8% and India at 34%. Among US respondents, the majority were women (70%), and 65% had a household income below \$60,000.

MTurk in Music-IR

The natural language processing (NLP) [57] and machine vision [58, 59] communities have utilized MTurk extensively, but machine listening and Music-IR have been slow to adopt its use. Lee found crowdsourcing music similarity judgments on MTurk to be less time-consuming than collecting data from experts in the research community [60]. The experiment cost \$130.90 and produced 6,732 similarity judgements, less than \$0.02 per rating. HITs were rejected if workers rated songs too quickly or failed to assign high similarity to identical songs. While nearly half of all HITs were rejected, the dataset was obtained an order of magnitude more quickly than their previous attempts. Comparing the datasets yields a Pearson’s correlation coefficient of 0.495, consistent with previous NLP work involving MTurk [57]. As the previous data collection was assembled for

MIREX, Lee returned the submitted systems using MTurk data as ground truth and found no significant alterations to the outcome, scoring a 5.7% difference on the Friedman test.

Mandel *et al.* employed MTurk for collecting free form tags to study relationships between audio tags and content [61]. The group collected 2,100 unique tags across 925 clips, for a reported cost of approximately \$100. To ensure data quality, they rejected a HIT if any tag had more than 25 characters, if less than 5 tags were provided, or if less than half of tags were contained in a dictionary of commonly applied tags (Last.fm). All HITs by a particular worker were rejected if the worker used too small a vocabulary, if they used more than 15% “stop words” (e.g., “music” or “nice”), or if half of their individual HITs were rejected for other reasons. The authors then trained a support vector machine (SVM) classifier for content-based autotagging. With smoothed labels, the MTurk version increased performance to 63.4% versus 63.09% with MajorMiner.

2.4 Content-Based Audio Analysis

Clearly, many human assessments of musical mood are derived from the audio itself; after all, tags are most often generated by people listening to the music. Contextual information for a music piece may be incomplete or missing entirely (e.g., for newly composed music). Given the rapid expansion of digital music libraries, including commercial databases with millions of songs, it is clear that manual annotation methods will not efficiently scale. Thus, the appeal of content-based systems is obvious, and the recognition of emotions from audio has been a longstanding goal for the Music-IR research community (the corresponding MIREX task focused on systems driven by music audio).

2.4.1 Acoustic Features

Emotions can be influenced by such attributes as tempo, timbre, harmony, and loudness (to name only a few), and much prior work in Music-IR has been directed towards the development of informative acoustic features. Although some research has focused on searching for the most informative features for emotion classification, no dominant single feature has emerged. An overview of the most common acoustic features used for mood recognition is given in Table 2.2.

Type	Features
Dynamics	RMS energy
Timbre	MFCCs, spectral shape, spectral contrast
Harmony	Roughness, harmonic change, key clarity, majoriness
Register	Chromagram, chroma centroid and deviation
Rhythm	Rhythm strength, regularity, tempo, beat histograms
Articulation	Event density, attack slope, attack time

Table 2.2: Common acoustic feature types for emotion classification.

In searching for the most informative emotion and expressive features to extract from audio, Mion and De Poli investigated a system for feature selection and demonstrated it on an initial set of single-dimensional features, including intensity and spectral shape, as well as several music-theoretic features [39]. Their system used sequential feature selection (SFS), followed by principal component analysis (PCA) on the subset to identify and remove redundant feature dimensions. The focus of their research, however, was monophonic instrument classification across nine classes spanning emotion and expression, as opposed to musical mixtures. Of the 17 tested features, the most informative overall were found to be roughness, notes per second, attack time, and peak sound level.

MacDorman *et al.* examined the ability of multiple acoustic features (sonogram, spectral histogram, periodicity histogram, fluctuation pattern, and mel-frequency cepstral coefficients–MFCCs [62, 63]) to predict pleasure and arousal ratings of music excerpts. They found all of these features to be better at predicting arousal than pleasure, and the best prediction results were obtained when all five features were used together [64].

Eerola *et al.*, the developers of the open-source feature extraction code, *MIRtoolbox* [65], have developed a specific subset of informative audio features for emotion. These features are aggregated from a wide range of domains including dynamics, timbre, harmony, register, rhythm, and articulation [38]. Other recent approaches have adopted a generalized approach to feature extraction,

compiling multiple feature sets (resulting in high dimensional spaces) and employing dimensionality reduction techniques [66, 67]. Regardless of feature fusion or dimensionality reduction methods, the most successful systems combine multiple acoustic feature types.

Feature Learning

An alternative method to extract features is to learn a basis from the collected data. Feature learning has only recently gained attention in the machine listening community and has yet to be applied to music emotion recognition. Lee *et al.* was the first to apply deep belief networks to acoustic signals, employing an unsupervised convolutional approach [68]. Their system employed PCA to provide a dimensionality reduced representation of the magnitude spectrum as the input to the DBN and showed slight improvement over MFCCs for speaker, gender, and phoneme detection.

Hamel and Eck applied deep belief networks (DBNs) to the problems of musical genre identification and autotagging [69]. Their approach used raw magnitude spectra as the input to their DBNs, which were constructed from three layers, employing fifty units at each layer. The system was trained using a greedy-wise pre-training and fine-tuned on a genre classification dataset, consisting of 1000 30-second clips. The system took 104 hours to train and as a result was not cross-validated. On the genre classification problem, the learned features achieved a classification accuracy of 0.843, which was an increase over MFCCs at 0.790. The learned model was also applied to inform an autotagging algorithm, which scored 0.73 in terms of mean accuracy, a slight improvement over MFCCs at 0.70.

2.4.2 Audio-Based Musical Mood Detection Systems

Audio content-based methods for emotion recognition use either a categorical or parametric model of emotion. The former results in a classification task and the latter in a regression task, with all recent systems employing one of these methods.

Categorical Emotion Classification

In one of the first publications on this topic, Li and Ogihara used acoustic features related to timbre, rhythm, and pitch to train support vector machines (SVMs) to classify music into one of 13 mood

categories [70]. Using a hand-labeled library of 499 music clips (30-seconds each) from a variety of genres spanning ambient, classical, fusion, and jazz, they achieved an accuracy of 45%.

Lu *et al.* pursued mood detection and tracking using a similar variety of acoustic features including intensity, timbre, and rhythm [66]. Their classifier used Gaussian Mixture Models (GMMs) for the four principal mood quadrants on the V-A representation. The system was trained using a set of 800 classical music clips (from a data set of 250 pieces), each 20 seconds in duration, hand labeled to one of the 4 quadrants. Their system achieved an overall accuracy of 85%, although it is also unclear how the multiple clips extracted from the same recording were distributed between training and testing sets.

Proposing a guided scheme for music recommendation, Mandel *et al.* developed active learning systems, an approach that can provide recommendations based upon any musical context defined by the user [71]. To perform a playlist retrieval, the user presents the system with a set of “seed songs,” or songs representing the class of playlist desired. The system uses this data, combined with verification data from the user, to construct a binary SVM classifier using MFCC features. When tested on 72 distinct moods from AMG labels, the system achieved a peak performance of 45.2%.

Skowronek *et al.* developed binary classifiers for each of 12 non-exclusive mood categories using a data set of 1059 song excerpts. Using features based on temporal modulation, tempo and rhythm, chroma and key information, and occurrences of percussive sound events, they trained quadratic discriminant functions for each mood, with accuracy ranging from 77% (carefree-playful) to 91% (calming-soothing) depending on the category [72].

Vaizman *et al.* investigated the use of the dynamic texture mixture (DTM) model for the representation of short-time audio features in an emotion classification problem [73]. In their work they consider each audio excerpt to contain a static emotion, which is chosen from one of four categories. The dataset consists of 72 audio excerpts, each of about 30 seconds. Using the DTM model, they consider their short-time features to be the output of a mixture of linear dynamical systems. By using such an approach they are able to take into account the dynamic evolution of the features required to produce a specific emotion. Their best performing approach obtains 0.8692

AROC, though it would be beneficial to see how their DTM representation compares in terms of performance to other methods such as standard Gaussian mixtures.

As mentioned in Chapter 1, MIREX first included audio music mood classification as a task in 2007 [35]. In 2007, Tzanetakis achieved the highest percentage correct (61.5%), using only MFCC, and spectral shape, centroid, and rolloff features with an SVM classifier [74]. The highest performing system in 2008 by Peeters demonstrated some improvement (63.7%) by introducing a much larger feature corpus including, MFCCs, Spectral Crest/Spectral Flatness, as well as a variety of chroma based measurements [75]. The system uses a GMM approach to classification, but first employs Inertia Ratio Maximization with Feature Space Projection (IRMFSP) to select the most informative 40 features for each task (in this case mood), and performs Linear Discriminant Analysis (LDA) for dimensionality reduction. In 2009, Cao and Li submitted a system that was a top performer in several categories, including mood classification (65.7%) [76]. Their system employs a “super vector” of low-level acoustic features and employs a Gaussian Super Vector followed by Support Vector Machine (GSV-SVM). It’s worth noting that the best performers in each of the three years of the evaluation were general systems designed to perform multiple MIREX tasks.

Parametric Emotion Regression

Recent work in music emotion prediction from audio has suggested that parametric regression approaches can outperform labeled classifications using equivalent features. Targeting the prediction of V-A coordinates from audio, Yang *et al.* introduced the use of regression for mapping high-dimensional acoustic features to the two-dimensional space [67]. Support vector regression (SVR) [77] and a variety of ensemble boosting algorithms, including AdaBoost.RT [78], were applied to the regression problem, and one ground-truth V-A label was collected for each of 195 music clips. As this work focused primarily on labeling and regression techniques, features were extracted using publicly available extraction tools, such as PsySound [79] and Marsyas [80], totaling 114 feature dimensions. To reduce the data to a tractable number of dimensions, PCA was applied prior to regression. This system achieves an R^2 (coefficient of determination) score of 0.58 for arousal and 0.28 for valence.

Han *et al.* began their investigation with a quantized representation of the V-A space and employed SVMs for classification [81]. Citing unsatisfactory results (obtaining 33% accuracy in an 11-class problem), they moved to regression-based approaches. Han reformulated the problem using regression, mapping the projected results into the original mood categories, employing SVR and Gaussian Mixture Model (GMM) regression methods. Using 11 quantized categories with GMM regression, they obtain a peak performance of 95% correct classification.

Eerola *et al.* introduced the use of a three-dimensional emotion model for labeling music, fully committing themselves to regression [38]. In their work, they investigated multiple regression approaches, including Partial Least-Squares (PLS) regression, an approach that considers correlation between label dimensions. They achieve R^2 performance of 0.72, 0.85, and 0.79 for valence, activity, and tension, respectively, using PLS and also report peak R^2 prediction rates for 5 basic emotion classes (angry, scary, happy, sad, and tender) as ranging from 0.58 to 0.74.

Madsen *et al.* propose a very interesting approach, developing a system that is trained on ranking data, but is capable of making V-A predictions in the testing phase. By operating on ranking data, they are able to perform highly controlled human data collection experiments. Their experiments simply ask subjects to rate pairs of songs as to which song is higher in terms of valence and arousal. The downside is that full evaluation of the system requires ranking on all pair combinations and thus the dataset is limited in size. Their current set contains 20 songs, and therefore 190 unique pairings. With the complete training set (90% of all data) they obtain valence and arousal error of 0.13 and 0.14, respectively [82].

2.4.3 Emotion Recognition Over Time

As few other Music-IR tasks are subject to dynamic (time varying) “ground truth,” it can be argued that accounting for the time varying nature of music is perhaps more important for emotion recognition than most other tasks. Because of this variation, systems relying on a single mood label to refer to an entire song or lengthy clip are subject to high classification uncertainty. Lu *et al.* pursued mood tracking across the four principal V-A quadrants, detecting mood changes at

1 second resolution. They report precision and recall for mood boundary detection at 84.1% and 81.5%, respectively, on a corpus of 9 movements from classical works [66].

Chapter 3: Data Collection

The ambiguous nature of musical emotion (mood) presents an interesting difficulty in collecting ground truth data. As discussed in Section 2.1.4, a variety of representations are used for mood-specific labels when collecting music corpora. For example, the two most ubiquitous collections use categorical representations curtailed by experts (i.e., Allmusic.com) and unstructured user-generated tags (i.e., Last.fm). While these approaches efficiently provide data for large collections, they are not well-suited for reflecting variations in the emotional content as the music changes. Throughout this work emotion will be represented using the parametric arousal-valence (A-V) orientation of the valence and arousal dimensions discussed in Section 2.1.4. Valence indicates positive versus negative emotion and arousal indicates emotional intensity [4]. This method potentially lends itself well to capturing labels dynamically (over time) to reflect emotion changes in synchrony with music and also provides quantitative labels that are well-suited to computational methods for parameter estimation.

This chapter will first present MoodSwings [15, 83], an online collaborative activity designed to collect second-by-second A-V labels on a large database of music. From the MoodSwings data, a corpus is constructed for supervised machine learning. This corpus, referred to as MoodSwings Lite [6], is a reduced set of the overall MoodSwings database. In forming MoodSwings Lite, this chapter will also discuss the aggregation of data from multiple subjects to form A-V “ground truth” representations. Next, the chapter turns to Amazon’s Mechanical Turk (MTurk) as another potential method for collecting new data (see Section 2.3.2). MTurk is investigated in seeking to address questions that have been raised about potential biases in the labels due to the collaborative nature of the game, and in looking for a method to collect data more quickly. The MoodSwings Turk [8] database is presented, which is reannotated to provide a denser data collection (i.e., more human annotations per clip) and to provide a means of analysis between the two data collection methods.

The last part of the chapter presents an entirely new corpus using instrumental data, which is hand selected to contain dynamic emotion transitions and is annotated with Mechanical Turk.

3.1 MoodSwings

Inspired by other successful Music-IR annotation games (Section 2.3.1), MoodSwings is a game for online collaborative emotion-based music annotation. The game board (Figure 3.1) is based on the two dimensional A-V model for human emotions. During gameplay, players are randomly paired across the internet to complete matches consisting of five 30-second music clips. The objective of the game is for players to achieve agreement of their valence-arousal labels over time, where closer distance between labels represents more agreement. The closer their labels are at any given time, the higher the points awarded, and their points will accumulate with agreement over the 30-second clip. Since agreement is necessary to score points, players are discouraged from making nonsensical labels. As an additional incentive for proactive and independent labeling, bonus points are awarded to the player who first reaches a particular location, making it impossible to outscore an opponent by simply following their cursor. This will ultimately provide labels at 1-second intervals of music across the entire database.

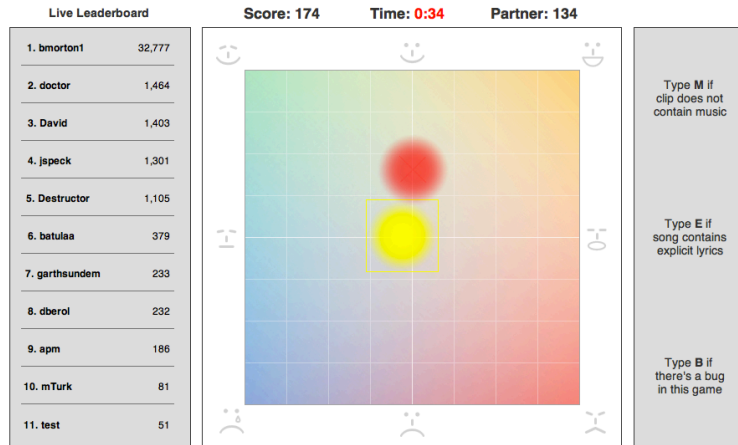


Figure 3.1: The MoodSwings gameboard.

3.1.1 Summary of Data Collection

The song clips used in MoodSwings are drawn from the “uspop2002” database [84], and overall the game has collected over 150,000 individual A-V labels spanning more than 1,000 songs. A contour map showing the distribution of the collected data can be seen in Figure 3.2. Since the database consists entirely of popular music, the labels collected to date display an expected bias towards high-valence and high-arousal values. Although inclusion of this bias could be useful for optimizing classification performance, it is not as helpful for learning a mapping from acoustic features that provides coverage of the entire emotion space, which is an issue that must be addressed to form a corpus for supervised machine learning.

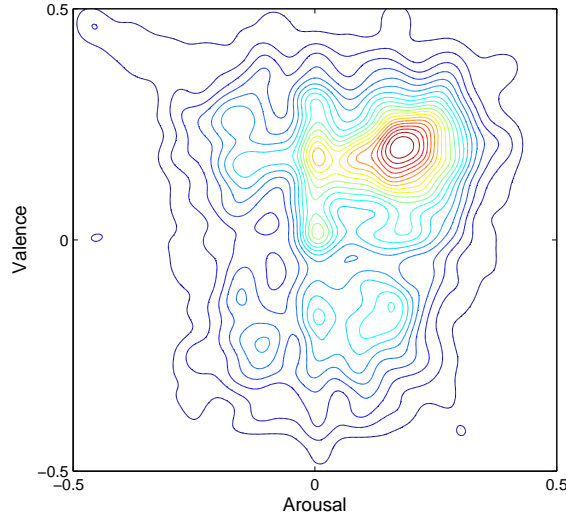


Figure 3.2: A contour plot showing the distribution of the MoodSwings labels [15].

Shown in Figure 3.3 is example song data from “American Pie” by Don McLean. In this transition between the first chorus and the second verse, there is a clear change in both intensity and emotion. In the song itself, this can be identified in the changes in instrumentation and increased tempo.

3.2 MoodSwings Lite Corpus

As a result of the high-valence, high-arousal trend, a reduced dataset consisting of 15-second music clips from 240 songs selected was developed. These clips were selected using the initial labels collected through the game to approximate an even distribution between the four primary quadrants of the

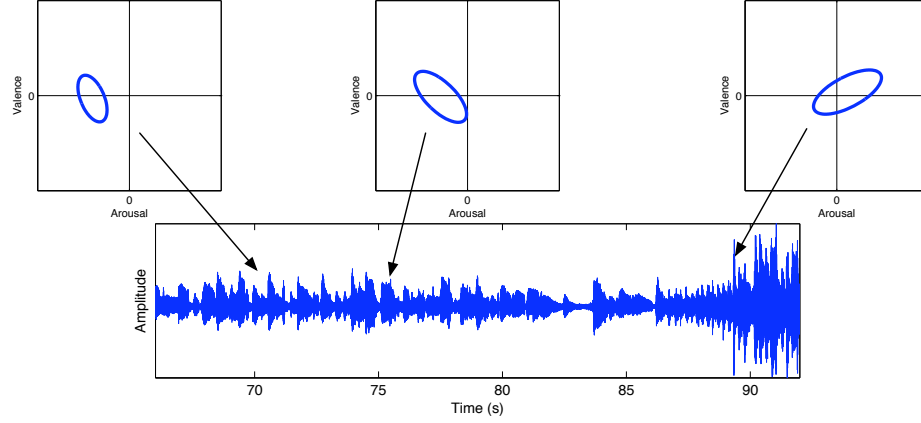


Figure 3.3: Progression of arousal-valence labels over an example clip. The ellipses represent the standard deviation across different players [15].

A-V space. These clips were then subjected to intense focus within the game in order to form a corpus, referred to here as MoodSwings Lite, with significantly more labels per song clip. The overall distribution of this label collection is shown as a contour map in Figure 3.4.

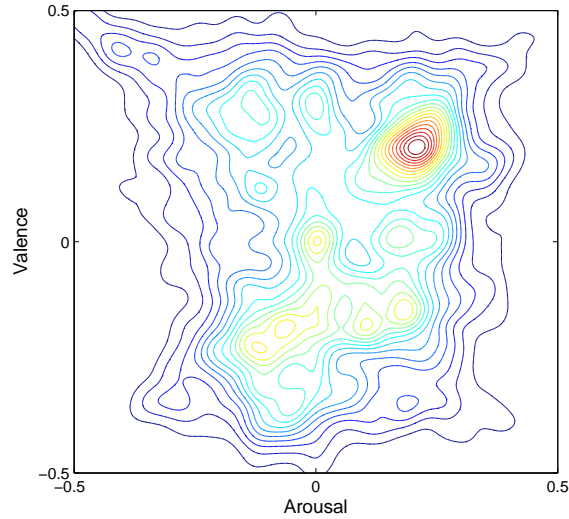


Figure 3.4: A contour plot showing the distribution of all MoodSwings Lite A-V labels.

Shown in Figure 3.5 is the data collected over a single 15-second clip. Each gray dot represents the annotation applied by a MoodSwings player. Clearly, all players agree on the general area of the A-V space that the clip lies in, but there is certainly a significant amount of variance or disagreement.

As all of these ratings are equally valid answers, it is very important to consider how the data is aggregated across multiple subjects and how disagreement could be represented.

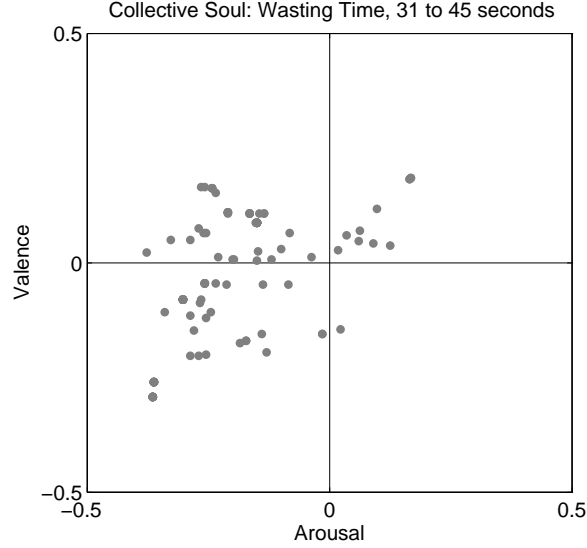


Figure 3.5: Data collected from MoodSwings over a 15-second clip.

3.2.1 Classification Corpus

The first representation investigated is for a simple 4-quadrant mood classification problem. In this representation, a single class label for each clip was obtained by quantizing the mean of all A-V labels collected across the duration of the clip. This simplification, while necessary for the classification task, will be seen as a potential source of error and is indicative of the inherent problems of focusing on singular mood classes, which is discussed in Chapter 7. Although the MoodSwings Lite corpus was used as the basis for classification, the original (uniform) distribution across the quadrants shifted slightly as more labels were collected for each individual clip. The final distribution of “ground truth” class labels is given in Table 3.1.

Class No.	Arousal	Valence	No. of Examples
1	high	high	70
2	low	high	51
3	low	low	59
4	high	low	64

Table 3.1: Quadrant-based class assignments of all MoodSwings Lite music clips.

3.2.2 Emotion Regression Corpus

Given the continuous nature of the collected A-V labels and the myriad problems produced by discrete emotional classes, this section discusses the development of a corpus for emotion regression. In emotion regression, which will be discussed in Chapter 7, multiple methods will be investigated for mapping acoustic features into the A-V space. Two versions of the corpus are constructed: one with static emotion labels, and the other with time varying sequences. The first version views the 15-second clip as a single example, just as was done with the classification corpus (Figure 3.6, left). The goal of the regression problem is to then predict one A-V coordinate for the clip. In the second version, the corpus is viewed as a sequence of 15 values, taking into account the variation in emotion over the clip (Figure 3.6, right). In those problems, the goal will be to analyze and predict the evolution of emotion over time.

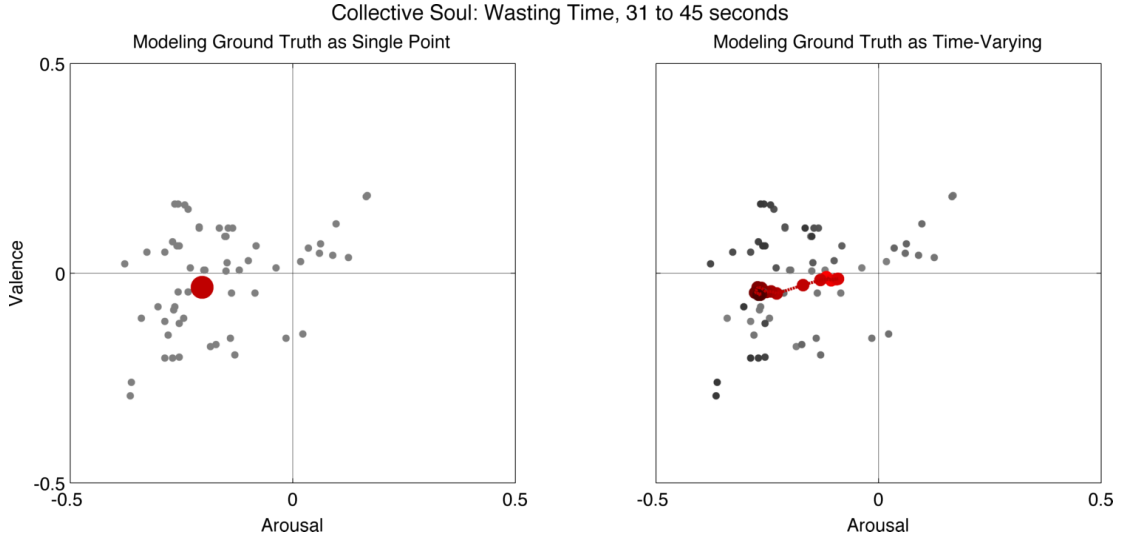


Figure 3.6: Modeling emotion space ground truth as a single A-V value. Left plot: modeling the clip as a single example. Right plot: modeling the clip as a sequence of fifteen examples (markers become darker as time advances). Both plots display the collected from MoodSwings (gray ●) and the ground truth representation (red ●).

3.2.3 Emotion Distribution Regression Corpus

While the regression corpus takes into account multiple opinions when determining the aggregated emotion label, disagreement by nature is a stochastic distribution. For these examples, instead of oversimplifying the ground truth into a single point, a single bivariate Gaussian distribution will

be used to represent the data. This assumption allows the representation of a reasonable degree of disagreement in emotion, and potentially provides a more realistic (and accurate) reflection of the perceived emotions conveyed by a song. Shown in Figure 3.7 is a graphical depiction of this representation. Just as in the single point corpus, here distributions are estimated on both the entire 15-second clip (Figure 3.7, left) and independently at each time step (Figure 3.7, right).

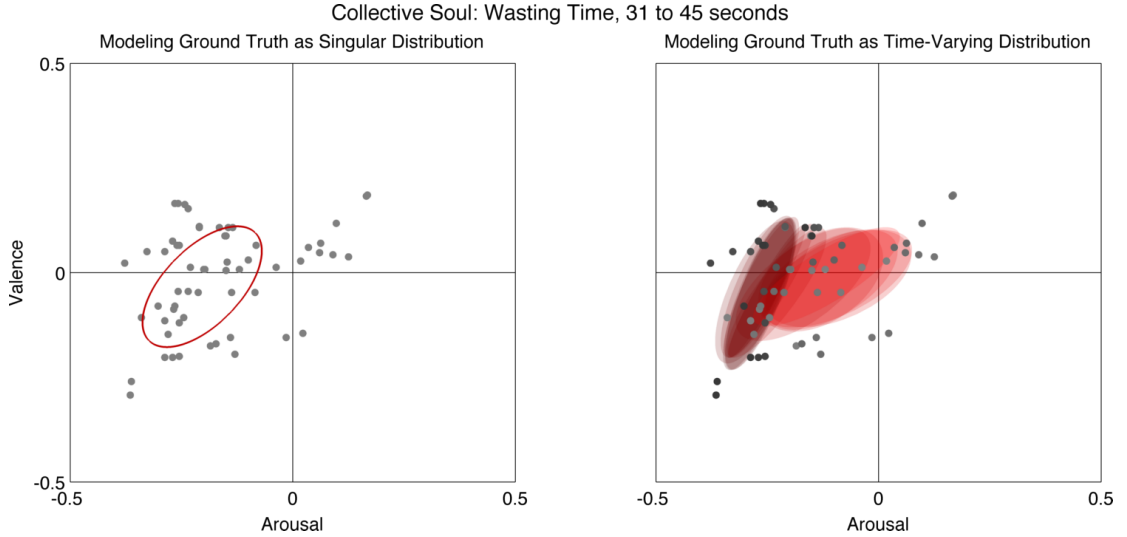


Figure 3.7: Modeling emotion space ground truth as a stochastic distribution. Left plot: modeling the clip as a single example. Right plot: modeling the clip as a sequence of fifteen examples (markers become darker as time advances). Both plots display the collected from MoodSwings (gray ●) and the ground truth representation (red).

3.3 MoodSwings Turk Corpus

Throughout the process of completing this work, many questions arose about the two-player structure of MoodSwings and whether or not it induced biases in the collected annotations as a result. The MoodSwings Turk dataset discussed here is a reannotation of the MoodSwings Lite dataset with Amazon’s Mechanical Turk (MTurk) service (see Section 2.3.2). The MTurk Human Intelligence Task (HIT) is completed by single annotators over the internet, who are compensated for their time. In addition to investigating biases in MoodSwings, it is also wished to explore MTurk as a means for quickly collecting data.

3.3.1 Mechanical Turk Data Collection

As in MoodSwings, the task collects per-second labels, but no partner is present and no points are awarded. Workers are given detailed instructions describing the A-V space before they begin. In the activity, they navigate to a website that hosts the task and label 11 randomly-chosen clips. The first clip is a practice round and is omitted from the analysis. The third and ninth are identical clips, randomly chosen from a set of 10 “verification clips,” which are evaluated to identify unsatisfactory work. Workers are given a 6-digit verification code to enter on the MTurk website as proof of completion which, if successful, will earn workers \$0.25 per HIT.

The system collected 4,064 label sequences after two stages of filtering: first evaluating verification clip labels and then removing labeling sessions of workers who kept the cursor at the origin for too long or consistently provided the same rating (e.g., consistently labeled all clips as angry throughout a game). It is assumed that the relatively small number of workers who did not move after 15 seconds misunderstood the task, and thus their data was filtered out. Table 3.2 shows statistics of the collected per-clip annotations in the dataset, before and after filtering.

Metric	Unfiltered Dataset	Verification Filtering	Stage 2 Filtering
Mean	49.79	18.20	16.93
St. Dev.	4.328	2.480	2.690
Max	72	24	23
Min	39	8	7

Table 3.2: Number of MTurk worker annotations for each clip before and after filtering.

3.3.2 MoodSwings Analysis with MTurk Data

We compute Pearson’s product-moment correlation between the datasets from MoodSwings and MTurk for each dimension. Pearson’s correlation between random variables X and Y is defined as:

$$\rho = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y}. \quad (3.1)$$

To account for discrepancies between the number of annotations for each clip, per-second sample means are treated as observations. The results, 0.712 for Arousal and 0.846 for Valence, show more

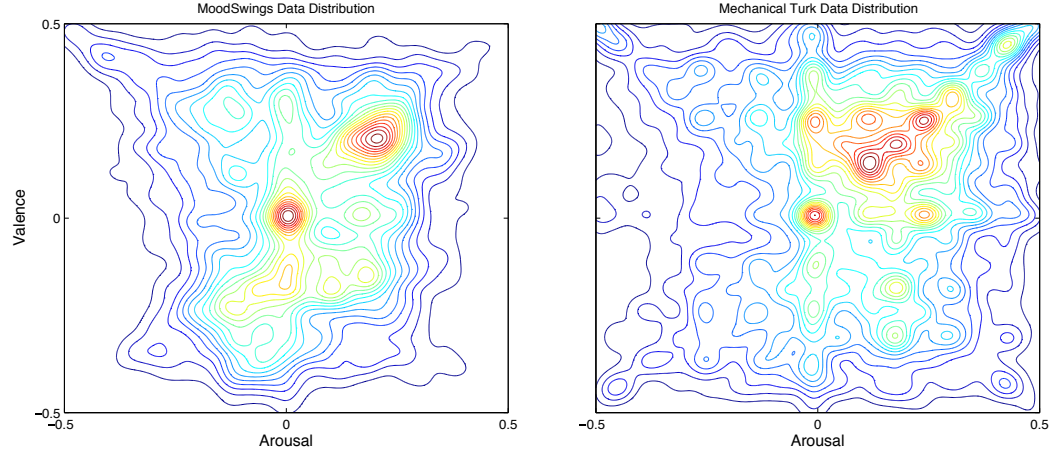


Figure 3.8: A-V distribution of data shown as a contour map. MoodSwings (left) and MTurk (right).

correlation between the two datasets than Lee’s comparison of a MTurk-collected dataset to similarly crowdsourced data [60]. High correlation provides evidence that annotators’ judgments are unlikely to be affected by collaborating with a partner during MoodSwings.

Figure 3.8 shows contour maps for the datasets collected with MoodSwings and MTurk. Both datasets have similar densities in the quadrant centers, though the MTurk dataset has higher densities along the space’s extremities, which could be attributed to a larger sample. The MTurk dataset also contains small peaks throughout the distribution, whereas the MoodSwings set has more consistent clusters. It is difficult to pinpoint a cause for this difference, but multiple small peaks in the MTurk distribution may suggest that workers remain indecisive about their mood ratings throughout the duration of a clip. In MoodSwings it was found that it takes 7-8 seconds on average for players to reach 85% of the total distance from the origin to their final mood labels [83]. By contrast, it took MTurk workers 10-12 seconds to reach the same distance percentage. Faster convergence towards a mood decision in the game could imply that collaboration encourages annotators to re-evaluate their ratings earlier in the clips, perhaps improving the quality of collected data.

3.3.3 Label Sequence Statistical Analysis

In looking to ultimately apply relational learning methods to this sequence data, those models will allow the ability to model statistical dependencies from one observation to the next. To verify that this data collection exhibits such dependencies, the correlation coefficients of the label sequences from one frame to the next, and from the first frame of each sequence to the last are computed. In these cases, the individual discretized user labels are treated as variables and each second as observations of those variables. Statistics of the squared correlation coefficients (r^2) are provided for the full dataset in Table 3.3.

Dimension	r^2 Frame-Frame	r^2 First-Last Frame
Arousal	0.944 ± 0.093	0.507 ± 0.242
Valence	0.951 ± 0.097	0.524 ± 0.235

Table 3.3: Statistics of ground truth squared correlation coefficient (r^2) from one second to the next and from the first second to the last.

Overall, the dataset shows high correlation from one frame to the next, and lower correlation between the first frame and last frame. In other words, the current emotion is highly dependent upon the emotion of the prior second, and on average each sequence exhibits a significant change in emotion from beginning to end. As a result, the dataset is a good match for graphical modeling techniques.

3.3.4 Emotion Space Heatmaps

In addition to regression and distribution regression, the MoodSwings Turk experiments (see Chapter 8) will also investigate the prediction of emotion space heatmaps. The previous methods for emotion distribution representation fit single bivariate Gaussian distribution to A-V data. This representation allows some variation in opinion, but it is limited to a single mode and all songs ratings do not necessarily follow the same distribution. Using emotion space heatmaps it is possible to represent arbitrary distributions with an arbitrary number of modes. An example of an A-V heatmap is shown in Figure 3.9.

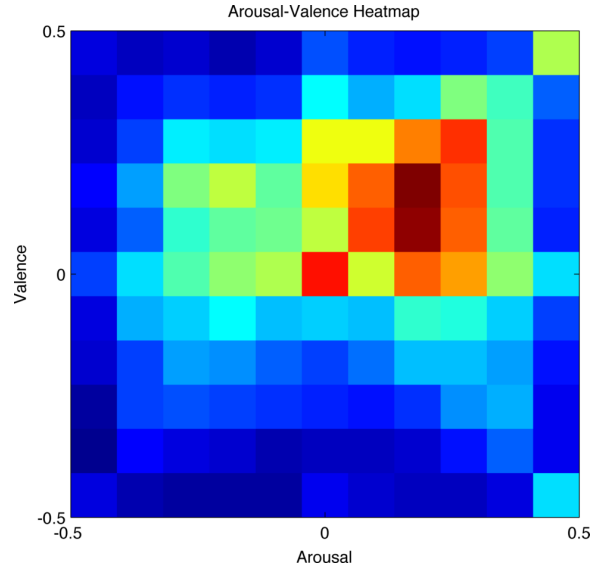


Figure 3.9: An emotion space heatmap.

3.3.5 MTurk Discussion

The strong positive correlation between data from MoodSwings and MTurk provides evidence that collaborating with a partner does not bias annotators' mood judgments any more than participating in a traditional labeling task. The logistics of each method are significant considerations, particularly the pace of data collection and time spent on quality control. Monetary incentives can attract annotators very quickly, but researchers must determine how to separate anonymous paid annotators (e.g., MTurk workers) with good intentions from those who wish to obtain payment for as little work as possible. With MoodSwings, however, the challenge is attracting annotators and the process of collecting data has been slow. The current MoodSwings Turk dataset contains annotations from an average of ~ 17 people for each song.

Given the success of MTurk discussed both here and in the experiments in Chapter 8, it is chosen as the method of annotation for the instrumental dataset, which will be discussed in the next section. While dealing with the noise in MTurk data is quite challenging, the speed at which the data can be collected makes it a more attractive method overall. While the approach of the game has perhaps been validated, the issue of attracting people to play it remains a much more difficult task.

3.4 Instrumental Data Corpus

The motivation in moving towards instrumental music is to remove any influence of lyrics and to gather data on music that potentially has a larger breadth of interpretations among annotators, and therefore more complex emotion space distributions. This corpus is largely centered around jazz and classical music and contains clips that are hand selected to lie in areas of the piece that contain large changes in emotion. This is a major difference from the MoodSwings Lite pop music dataset, where many of the clips lie within a single structural segment (e.g., verse, chorus). In addition, the dataset will consist of 30-second sequences, which are twice as long as the current set and are potentially better suited to train time-varying models. The final corpus contains 300 clips, and because they were hand picked the selection was a very time consuming process.

3.4.1 Data Collection

Just as with MoodSwings Turk, workers are given detailed instructions describing the A-V space before they begin. In the activity, they navigate to a website that hosts the task and label 11 randomly-chosen clips. The first clip is a practice round, omitted from the analysis. The third and ninth are identical clips, randomly chosen from a set of 10 “verification clips,” which are evaluated to identify unsatisfactory work. Workers are given a 6-digit verification code to enter on the MTurk website as proof of completion that, if successful, will earn workers \$0.35 per HIT, a 10 cent increase over MoodSwings Turk. The increase was given in order to compensate for the slightly longer clips, as well as to hopefully attract more annotators.

This data collection also provides additional filtering to ensure data quality. Just as in MoodSwings Turk, each HIT is manually compared to a set of predetermined “verification” songs that have been labeled by a set of experts. To ensure this process is done properly, each HIT is manually evaluated by three people, and the final decision of whether to include the HIT in the dataset is taken using majority voting. The final dataset contains annotations from an average of 15.52 ± 1.820 workers for each 30-second clip.

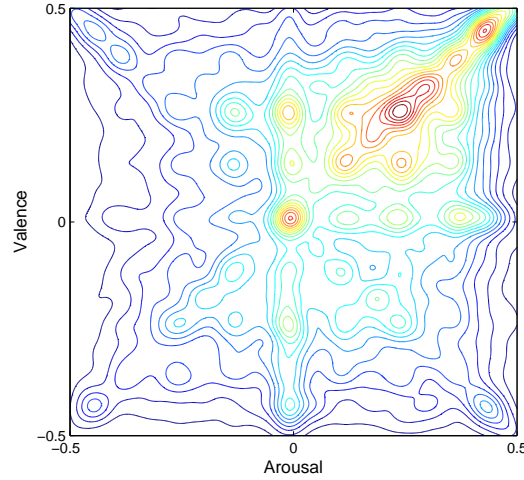


Figure 3.10: A contour plot showing the A-V distribution of the instrumental dataset.

3.4.2 Comparison to MoodSwings Pop Music Corpus

A contour plot of the Instrumental Dataset distribution is shown in Figure 3.10. Comparing to MoodSwings Lite and MoodSwings Turk (Figure 3.8), a bias towards the top right quadrant can also be seen. However, it also seems that this dataset is not as centered around the origin as previous sets. This is perhaps due to the emotion-space dynamics contained in the music pieces; annotators are constantly required to change their ratings, which generally requires the movement towards one extreme or the other.

Table 3.4 contains the aggregated squared correlation coefficients for individual user ratings from second to second (Frame-Frame) and from first second to last second (First-Last Frame). Comparing these values to MoodSwings Turk (Table 3.3), a similar second by second correlation can be seen, in that the emotion at the current second depends heavily on the previous second. However, a significant change can be seen in the first to last second comparisons, demonstrating that these clips have much larger emotion-space dynamics over the range of the clip.

Dimension	r^2 Frame-Frame	r^2 First-Last Frame
Arousal	0.926 ± 0.113	0.192 ± 0.186
Valence	0.946 ± 0.098	0.305 ± 0.223

Table 3.4: Statistics of ground truth squared correlation coefficient (r^2) from one second to the next and from the first second to the last.

Chapter 4: Acoustic Feature Selection

This chapter investigates the selection of acoustic features for music emotion recognition. Current approaches to feature selection (e.g., sequential feature selection [39, 85]) often focus on selecting features that provide the best separability within the space of a specific classification problem. These approaches work well when competing against other researchers to obtain the best classification performance on a given dataset, but what holds true on a specific dataset does not necessarily generalize to the global task (e.g., music emotion recognition). Given the difficulty of obtaining music content, datasets are often assembled using convenience sampling (i.e., using music that is readily available). Thus, while datasets can still be paramount in advancing the state of the art in certain fields, few if any can be thought of as an ideal sampling of the true distribution of music.

By developing controlled experiments, this chapter seeks to investigate feature selection for the general problem of music emotion recognition. The first experiments seek to perceptually evaluate two of the most commonly used features in Music-IR: mel-frequency cepstral coefficients (MFCCs) and the chromagram. MFCCs have been shown to be one of the most informative feature domains for music emotion recognition [6, 9, 7, 11], but as MFCCs were originally designed for speech recognition, it is unclear why they perform so well or how much information about emotion they actually contain. Conversely, the chromagram appears to be one of the most intuitive representations, as it provides information about the notes contained in the piece, which could potentially provide information about the key and mode. Thus far, chroma has shown little promise in informing this problem. In order to properly assess these features, a perceptual study is conducted using Amazon’s Mechanical Turk [86] (MTurk). The goal of the study is to analyze the relative emotion of two song clips, comparing human ratings of both the original audio to those of audio reconstructions from these features. By asking human subjects to analyze these reconstructions, it is perhaps possible to get a much more accurate depiction of how informative these features are to musical emotion.

Given such a data collection, this chapter also seeks to identify patterns and relationships between musical parameters (e.g. key, mode, tempo) and perceived emotion. By identifying variability in emotion related to these parameters, this chapter identifies existing features that respond with the highest variance to those that inform emotion, and the least variance in those that do not. In order to properly assess a large variety of features, these experiments utilize the features used the perceptual study reconstructions, as well as the other features used in this work, and 14 additional features from the MIR-toolbox [65].

4.1 Data Collection

In previous studies (such as [24]), several controlled variations of musical phrases are provided to each participant. Since this chapter focuses on studying the changes in the acoustic feature domain, it is necessary to have samples that can be easily manipulated in terms of mode and tempo and that provide a wide enough range to ensure that all possible variations in the feature space are accurately represented. To this end, this chapter uses a dataset all of its own, separate from those discussed in Chapter 3 and later used for modeling and prediction. This dataset consists of 50 Beatles MIDI files, attained online [87], spanning 5 albums (Sgt. Peppers, Revolver, Let It Be, Rubber Soul, Magical Mystery Tour). In order to remove the effect of instrumentation, each song was synthesized as a piano reduction and a random twenty second clip of each song was used for the labeling task.

4.1.1 Song Clip Pair Selection

Labeling the entire 1225 possible pairs from the 50 songs would be prohibitive, so the focus is instead narrowed to a subset of 160 pairs. Since the Beatles dataset contains 35 songs in the major (Ionian) mode and only 9 in the minor (Aolean) mode (with 6 additional pieces in alternate modes), it is necessary to ensure that major-major pairings do not completely dominate our task. Some songs are represented one extra time in order to generate 160 pairs but no song is repeated more than once. Out of these 160 pairs, there are 81 major-major pairings, 33 major-minor pairings, and 7 minor-minor pairings.

For each song, a piano reduction of the MIDI file is rendered for the 20 second clip, and then MFCCs and chroma features are computed on the audio. Audio examples for reconstructions are synthesized from the resulting features. Chromagram features are extracted and reconstructed using Dan Ellis’ chroma features analysis and synthesis code [88] and MFCCs using his rastamat [89] library. The MFCC reconstructions sound like a pitched noise source, and while the chroma reconstructions have an ethereal ‘warbly’ quality to them, they sound more like the original audio than the MFCC reconstructions.

4.1.2 Mechanical Turk Annotation Task

In order to annotate the clip pairs, Amazon’s online crowd-sourcing engine Mechanical Turk is employed, providing input from a wide variety of subjects. In the MTurk Human Intelligence Task (HIT), participants are asked to label four uniformly selected song pairs from each of the three categories: original MIDI rendering, MFCC reconstructions, and chromagram reconstructions. For each pair of clips, participants are asked to label which one exhibits more positive emotion and which clip is more intense. The three categories of audio sources are presented on three separate pages. The participants are always comparing chroma reconstructions to chroma reconstructions, MFCC reconstructions to MFCC reconstructions, or MIDI renderings to MIDI renderings. Subjects never compare a reconstruction to the original audio. For each round, a clip is randomly selected to be repeated as a means of verification. If a user labels the duplicated verification clip differently during the round with the original audio, their data is removed from the dataset.

4.2 Experiments and Results

The first set of experiments investigate the emotional information retained in some of the most common acoustic features used in Music-IR, MFCCs and chromagrams. As described above, users listen to a pair of clips that was reconstructed from features (MFCC or chroma) and rate which is more positive and which has more emotional intensity. It is desired to quantify how much information about musical emotion is retained in these acoustic features by how strongly emotion ratings of the reconstructions correlate with that of the originals. First, the user ratings are related to musical

tempo and mode, and then it is explored which features exhibit high variance with changes in tempo and mode or are invariant to altering these musical qualities.

4.2.1 Perceptual Evaluation of Acoustic Features

Running the task for three days, a total of 3661 completed HITs were collected, and a total of 1,426 were accepted for an approval rating of 39%, which is similar to previous work annotating music data with MTurk [8, 60, 61]. The final dataset contains 17,112 individual song pair annotations, distributed among 457 unique Turkers, with each Turker completing on average ~ 2.5 HITs. With a total of 160 pairs, this equates to ~ 35.65 ratings per pair. HITs are rejected for completing the task too quickly (less than 5 minutes), failing to label the repeated verification pairs the same for the original versions, and failing too many previous HITs. While repeated clips were presented for both reconstruction pairs and originals, requiring identical ratings on the reconstructions ultimately proved to be too stringent, due to the nature of the reconstructed clips. For the original clips, the repeated pair was required to have the same ratings for both the higher valence and higher arousal clips, and the A/B presentation of the clips was reversed to ensure Turk users were not just selecting song A or song B for every pair to speed through the task.

For each pair and for each audio type, the percentage of subjects that rated clip A as more positive (valence) and the percentage that labeled clip A as more intense (arousal) is computed,

$$p_v = \frac{1}{N} \sum_{n=1}^N \mathbb{1}\{A_n = \text{HigherValence}\}, \quad (4.1)$$

$$p_a = \frac{1}{N} \sum_{n=1}^N \mathbb{1}\{A_n = \text{HigherArousal}\}, \quad (4.2)$$

where N is the total number of annotations for a given clip, p_v is the percentage of annotators that labeled clip A as higher valence, and p_a is the percentage of annotators that labeled clip A as higher arousal. For each song pair, the percentage of Turkers who rated song A as more positive in the original audio is compared to those who rated song A more positive in the reconstructions, yielding the normalized difference error for all songs.

Audio Source	Normalized Difference Error	
	Valence	Arousal
MFCC Reconstructions	0.133 ± 0.094	0.104 ± 0.080
Chroma Reconstructions	0.120 ± 0.095	0.121 ± 0.082

Table 4.1: Normalized difference error between the arousal/valence ratings for the reconstructions and the originals.

In Table 4.1, the error statistics are shown for the deviation between the two groups. The paired ratings of each type are also verified with a paired Student’s t-test to verify that they do not fall under the alternative hypothesis that there is a significant change, but as it is desired to have proof that there is no change, average error remains the best indicator. Figure 4.1 shows histograms of these error values over all examples. It can be seen that in general MFCCs are less uniform for Arousal, and Chroma is less uniform for Valence, indicating that those features contain more information about those emotion parameters.

4.2.2 Relationships Between Musical Attributes and Emotional Affect

The next experiments analyze the data for trends relating major/minor modes and tempo to valence and arousal. In Section 2.1.3, the general trend of major tonality being associated with positive emotional affect was discussed, as well as higher tempo corresponding to an increase in arousal or valence.

A subset of the dataset S is taken $M \subset S$ that consists of pairs that contain one major mode song and one minor mode song, as well as a subset $T \subset S$ in which pairs differ in tempo by more than 10 beats per minute (bpm). For subset M , the percentage of users that labeled the major song as more positive and the percentage of users that labeled the major song as more intense are calculated. For subset T , it is determined whether the faster song is more intense and whether the faster song is happier according to the users. Looking at Table 4.2, it is concluded that the results are commensurate with the findings from the various psychology studies referenced in Section 2.1.3, namely that major songs are happier and faster songs are more intense.

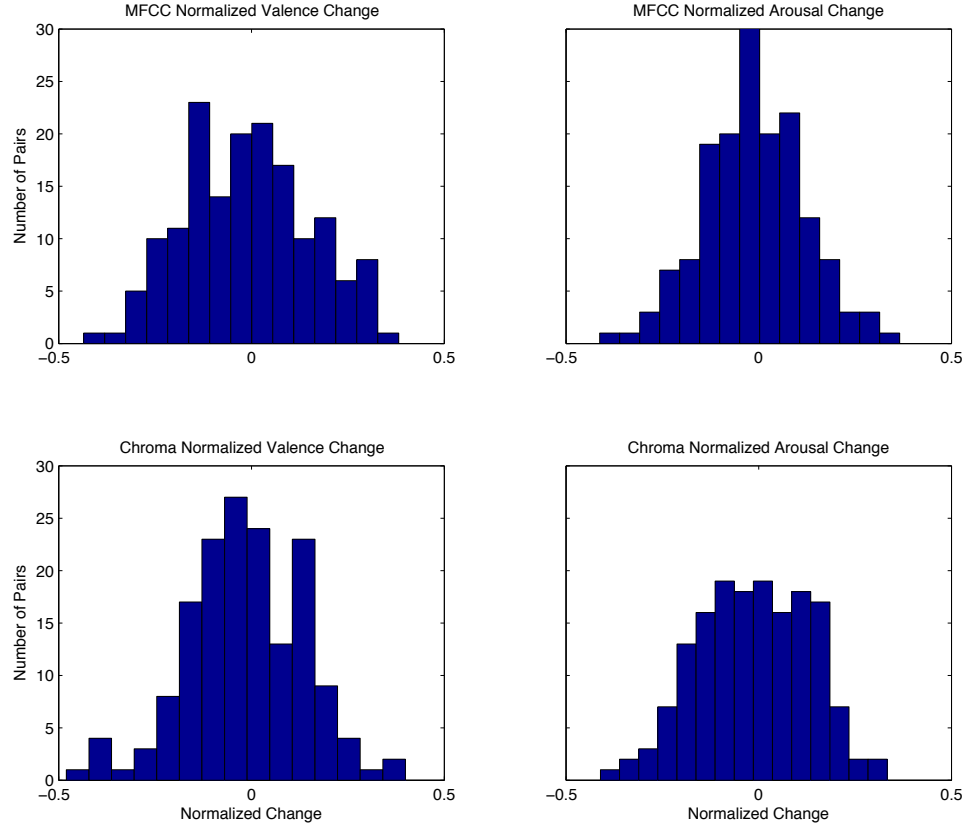


Figure 4.1: Histograms of normalized difference error between the arousal/valence ratings for the reconstructions and the originals.

Null Hypothesis	Agreement Ratio
Major Key Labeled as More Positive Valence	0.667
Faster Tempo Labeled as More Positive Valence	0.570
Major Key Labeled as More Positive Arousal	0.528
Faster Tempo Labeled as More Positive Arousal	0.498

Table 4.2: Percentage of paired comparisons that yielded the desired perceptual result for mode and tempo.

One area where larger agreement was expected is the relationship between tempo and intensity. Only the beats per minute for each song are available, and the faster song is labeled as the one with a higher bpm. The note lengths and emphasis in relation to the tempo are disregarded in this analysis and may be a source of uncertainty in the result. Depending upon the predominant note

value (quarter/eighth/sixteenth), a slower tempo can sound faster than a song with a higher number of beats per minute. These are two different compositions, not the same clip at two different tempos.

4.2.3 Identifying Informative Feature Domains

When using features to understand certain perceptual qualities of music, it is important to know how those features relate to changes in the perceptual qualities being studied. This section looks to find appropriate variances and invariances as they relate to a perceptual quality. For example, supposing emotion is invariant to key, if the key changes, the features should also be invariant to that key change. It is desired to have correlation in variance as well. If the emotion of the audio changes, the desired features are the ones that describe it to change in response. In order to investigate these variances and invariances, the feature sets used in other experiments in this work are used, as well as a set of features from the MIR-toolbox. Using the Beatles' clips, changes in key, tempo, and mode are generated to investigate possible corresponding differences in features. For key, the original was compared with transposed versions a 5th above and below. For tempo, the original was compared with versions at 75% and 133% of the original tempo. For mode, all the minor songs were shifted to major and all the major songs to natural minor and compared the full dataset in major vs. the full dataset in minor.

Because the features contain different dimensions and have different ranges, looking at differences in their direct results does not allow for proper comparison between them. In order to draw proper comparisons, the features are normalized over dimension and range.

Given 2 feature vectors over time $F_1 \in \mathbb{R}^{M \times N}$ and $F_2 \in \mathbb{R}^{M \times N}$, the content is normalized over the vectors' shared range,

$$F'_1 = \frac{F_1 - \min(F_1 \cup F_2)}{\max(F_1 \cup F_2)}, \quad (4.3)$$

$$F'_2 = \frac{F_2 - \min(F_1 \cup F_2)}{\max(F_1 \cup F_2)}. \quad (4.4)$$

The mean for each dimension is calculated, creating mean vectors $\mu_1 \in \mathbb{R}^{N \times 1}$ and $\mu_2 \in \mathbb{R}^{N \times 1}$. The average feature change across all dimensions is then computed,

$$FeatureChange = \frac{1}{N} \sum_{n=1}^N |\mu_1(n) - \mu_2(n)|. \quad (4.5)$$

If this *FeatureChange* value is low, it means that the feature is invariant to the musical change being presented. In Table 4.3 it is observed that features that exhibit higher variance to the specified change (tempo up/down, key up/down, and mode shift) should be more effective in computational models that are sensitive to these parameters. Several intuitive features, including onsets, RMS energy, and beat spectrum, emerge as the most variant features to tempo. Conversely, it is intuitive that features like mode and tonal center do not vary much with tempo.

Tempo Up		Tempo Down		Key Up		Key Down		Mode Shift	
Feature Domain	Feature Change	Feature Domain	Feature Change	Feature Domain	Feature Change	Feature Domain	Feature Change	Feature Domain	Feature Change
Onsets	0.127	Onsets	0.126	Key	0.142	Key	0.145	Mode	0.142
Beat Spec.	0.081	Beat Spec.	0.078	Beat Spec.	0.134	Beat Spec.	0.131	Tonal Cent.	0.114
RMS Energy	0.049	RMS	0.050	Tonal Cent.	0.105	Tonal Cent.	0.102	Beat Spec.	0.103
HCDF	0.024	HCDF	0.022	MFCC	0.084	MFCC	0.178	Key	0.063
xChroma	0.024	xChroma	0.021	Zerocross	0.081	Zerocross	0.064	Chroma	0.047
Roughness	0.023	Roughness	0.019	Chroma	0.055	Chroma	0.051	MFCC	0.030
Zerocross	0.022	SSD	0.017	Contrast	0.054	Contrast	0.049	Brightness	0.019
Brightness	0.021	MFCC	0.016	Regularity	0.050	xChroma	0.048	Onsets	0.015
SSD	0.021	Brightness	0.015	xChroma	0.038	Regularity	0.045	Attack time	0.014
MFCC	0.017	Zerocross	0.015	Mode	0.038	SSD	0.041	Regularity	0.013
Chroma	0.014	Chroma	0.014	Brightness	0.037	Brightness	0.041	Zerocross	0.012
Key	0.013	Key	0.014	SSD	0.036	Mode	0.040	Contrast	0.011
S. Contrast	0.012	Regularity	0.011	Attack time	0.030	Attack time	0.026	xChroma	0.011
Regularity	0.012	Contrast	0.010	RMS	0.021	Roughness	0.023	SSD	0.010
Fluctuation	0.011	Fluctuation	0.009	Roughness	0.021	Onsets	0.020	RMS	0.009
Attack time	0.010	Mode	0.007	Onsets	0.017	RMS	0.017	Attack Slope	0.008
Mode	0.009	Attack time	0.007	Attack Slope	0.015	HCDF	0.015	Roughness	0.007
Tonal Cent.	0.007	Tonal Cent.	0.006	HCDF	0.012	Attack Slope	0.009	HCDF	0.006
Attack Slope	0.006	Attack Slope	0.005	Fluctuation	0.008	Fluctuation	0.008	Fluctuation	0.002

Table 4.3: Normalized feature change with respect to musical mode and tempo alterations.

4.3 Discussion

In this chapter, perceptual evaluation of emotional content in audio reconstructions from acoustic features has been provided. In addition, these findings have been related to those from previous work, showing correlation between major keys and increased positive emotion, as well as increased tempo and increased positive emotion and activity. For tempo, mode and key, this chapter has provided a variational analysis for a large number of acoustic features. The findings presented here

should be informative for future computational investigations in modeling emotions in music using content based methods.

Chapter 5: Acoustic Feature Learning

While there has been much progress in machine learning systems for estimating human emotional response to music, little progress has been made in terms of intuitive feature representations. As discussed in Chapter 2, current methods generally focus on combining several feature domains (e.g. loudness, timbre, harmony, rhythm) and performing dimensionality reduction techniques to extract the most relevant information. In many cases, these methods have failed to provide enhanced classification performance, and they leave much to be desired in terms of understanding the complex relationship between emotional associations and acoustic content.

The Music Information Retrieval Evaluation eXchange (MIREX) [90] audio mood classification task provides an excellent illustration of this [35]. Shown in Figure 5.1 is the performance of MIREX submissions for each year. The first year MIREX ran the task, it received 9 submissions, and the best performing system achieved 61.50% performance on the 6-class problem using a feature space spanning 16-dimensions [74]. Each year the task has received a larger number of submissions, with exponentially larger feature libraries, but it has failed to produce significant performance gains. Most recently, in 2010 the task received 36 submissions with the best system mining a 70-dimensional feature space, but achieved only 64.17% [91]. These results perhaps indicate that the data necessary for informing systems for this problem is not present in any current feature set.

Instead of collecting sets of existing features, this chapter will discuss methods to potentially learn representations of music audio that are specifically optimized for the prediction of emotion. One consideration in feature learning is obtaining sparse representations. Sparse representations are particularly desirable in the application of feature learning for forming compact representations of high-dimensional spectral data, and previous work in sparse methods has also shown that such representations lead to basis functions that closely model mammalian auditory filters in the auditory cortex [92]. As a result, such representations can potentially lead to particularly informative feature domains.

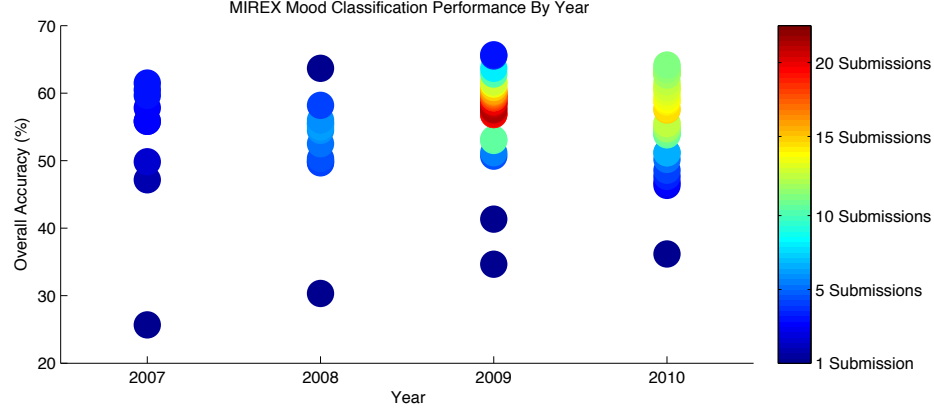


Figure 5.1: MIREX mood classification task performance by year.

The ambiguous nature of musical emotion makes it an especially interesting problem for the application of feature learning. Using deep belief networks (DBNs) [93], this chapter will develop methods for the learning of emotion-based acoustic representations directly from magnitude spectra. The topology of a trained DBN is identical to that of a multi-layer perceptron (MLP) or neural network, but DBNs employ a far superior training procedure involving a secondary topology. DBN training begins with an unsupervised pre-training approach using greedily-trained restricted Boltzmann machines (RBMs) [94, 95]. The general approach is to attach a logistic regression layer for classification after pre-training, and to then use gradient descent to perform the fine-tuning. In this approach, the DBN is implemented to learn feature detectors for a regression problem and instead a linear regression layer is attached.

In general, deep belief networks have received very little attention in the application of regression problems, and at the time of writing no other work is known that has applied them to any regression-based feature learning problem. While this model is applied specifically for music emotion recognition, the topology presented here could easily be applied to a wide variety of regression-based feature learning problems.

5.1 Deep Belief Networks

A fully trained deep belief network (DBN) [93, 94, 95, 96] shares an identical topology to a multi-layer perceptron (MLP) or neural network, though they offer a far-superior training procedure,

beginning with an unsupervised pretraining that models the hidden layers as restricted Boltzman machines (RBMs) [93, 94, 95, 96, 97]. The general goal of pretraining is to initialize the network to the distribution of the input data. Finetuning, as will be discussed, consists of gradient descent and relies heavily on starting conditions, and properly pretraining the network is therefore crucial.

All DBN development for this work was done using Theano [98], a Python-based package for symbolic math compilation. Theano is an extremely powerful tool for machine learning problems because it combines the simplicity of Python with the power of compiled C, which can target the CPU or GPU.

5.1.1 Restricted Boltzman Machines

A restricted Boltzman machine is a type of log-linear Markov random field with hidden units. A graphical depiction of an RBM is shown in Figure 5.2. An RBM is a generative model that contains only a single hidden layer, and in very simplistic terms it can be thought of as a set of basis vectors that serve to both reduce the dimensionality of the data as well as reconstruct it. These models are restricted in the sense that they contain only connections between the visible and hidden layer (i.e., no visible-visible or hidden-hidden connections).

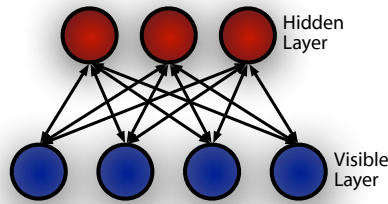


Figure 5.2: Restricted Boltzman machine topology.

These generative models express their probability distribution via an energy function, which is shown here as a joint configuration between an observed part \mathbf{v} and hidden part \mathbf{h} [99],

$$E(\mathbf{v}, \mathbf{h}) = - \sum_{i \in \text{visible}} b_i v_i - \sum_{j \in \text{hidden}} a_j h_j - \sum_{i,j} v_i h_j w_{ij}. \quad (5.1)$$

In the experiments discussed in this work (Chapter 7 - 8), the visible layer is a spectrogram $\mathbf{v} \in \mathbb{R}^{I \times 1}$ and the hidden layer $\mathbf{h} \in \mathbb{R}^{J \times 1}$. The model has parameters $W \in \mathbb{R}^{I \times J}$, with biases $a \in \mathbb{R}^{J \times 1}$ and $b \in \mathbb{R}^{I \times 1}$. A more detailed RBM tutorial is available in Appendix B.

5.1.2 Constructing a Multi-Layer Network

When constructing deep architectures, the standard method is via “greedily” stacking restricted Boltzman machines [95]. In this method RBMs are trained one at a time starting from the bottom. After the first layer RBM is trained using the collected data, only the forward weights are retained. Then samples of that RBM’s hidden layer (Equation B.8) are used as the visible layer of the next RBM. Another way to think about this is that the outputs of the first layer are used as training data for another RBM. Shown in Figure 5.3 is a three layer DBN.

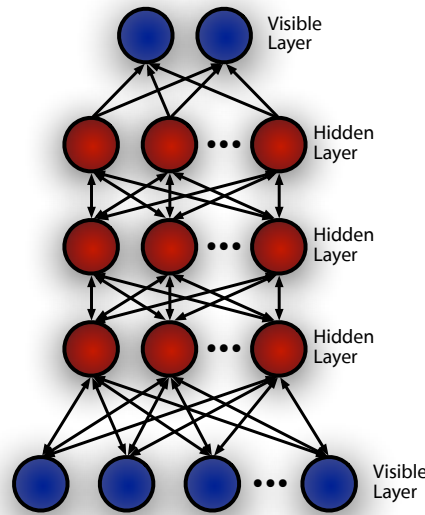


Figure 5.3: Greedy stacking of restricted Boltzman machines.

Starting from the bottom, the first layer is trained as an RBM, and subsequent layers are trained using samples from the previous ones. Note that the first hidden layer will become the visible layer of the next RBM. Figure 5.3 shows two way connections at each level, though the weights are untied after RBM training and only the forward weights are retained. Layer sizes are not noted, though in some approaches, layers become smaller when moving up in the network to fully take advantage of the dimensionality reduction power of RBMs. The last step is to add an additional visible layer

for finetuning the network. This generally involves a logistic regression layer to train the network for classification, but in this work a linear regression layer will be attached due to the continuous nature of the label data.

5.1.3 Supervised Finetuning

The model resulting from stacking RBMs is referred to as a multi-layer perceptron (MLP), and it is then finetuned to predict some set of outputs. The general approach is to use some form of gradient descent, which operates on the back propagation of cost or error and optimizing gradients of the model parameters with respect to error [100],

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \epsilon \Delta E(\mathbf{w}^{(t)}), \quad (5.2)$$

where \mathbf{w} is the model parameters, t is the current time, and ϵ represents the “learning rate,” which limits how much the parameters will be allowed to change on each iteration. $E(\mathbf{w}^{(t)})$ is a function describing the error of the model at the current time. This work uses the built-in conjugate-gradient optimizer in Python’s SciPy package [101].

Generally speaking, these algorithms relate the changes in model parameters to changes in error and modify those parameters in the direction of minimizing error. The limitation of these methods is that they rely on having a starting position to *descend* from, which is near to the global minima. As these algorithms are very likely to get caught in local minima, it is crucial that the MLP is initialized as well as possible.

5.2 Generating Features with DBNs

A trained DBN can be used to serve several purposes. The most obvious is to use the model to make predictions on unseen data, but the model can also be used in other ways. For one, analyzing the model parameters can potentially shed new light on the relationship between acoustic content and human emotions. Furthermore, the model can also be used as a set of basis functions that are specifically tuned for human emotion. In these cases, the individual layers of the DBN are used as features in an alternate machine learning algorithm.

Shown in Figure 5.4 is an example of how this process may work. Moving from left to right in the figure, magnitude spectra is used as input to the DBN. The experiments in Chapter 7 use only a single frame of spectra as inputs, and the A-V coordinates are resampled to match the 20 msec rate of the features. In the experiments in Chapter 8, spectra is aggregated at multiple window sizes and concatenated to form DBN inputs. Shown in Figure 5.4, is the later method, which aggregates spectra. Furthermore, once the model is trained, the individual layers are used in an alternate machine learning algorithm. Shown here is a simple multiple linear regression algorithm to predict parametrized emotion distributions in the A-V space. While MLR is another very simple algorithm, the application of much more sophisticated algorithms, such as support vector regression and conditional random fields will be discussed.

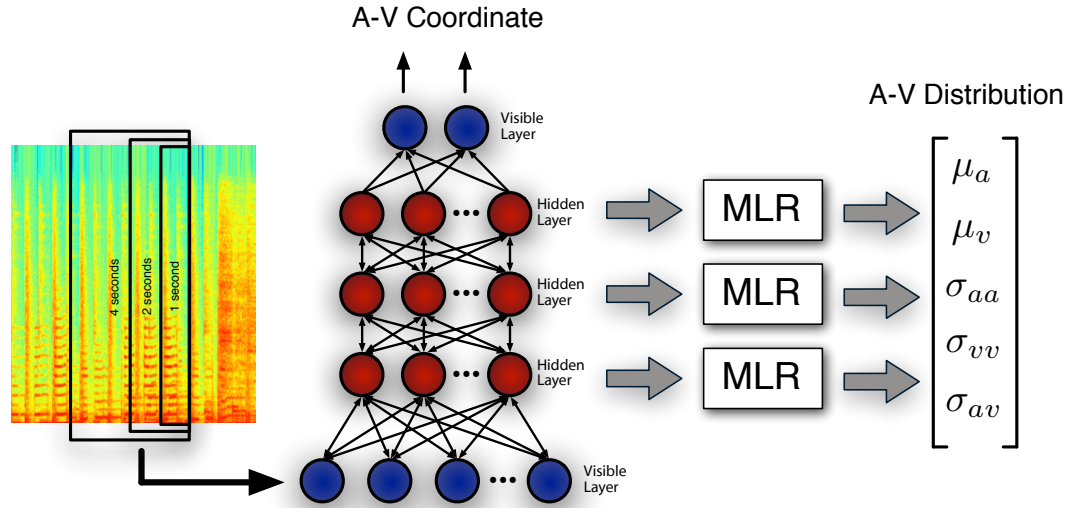


Figure 5.4: Feature learning system architecture showing the temporal aggregation, deep belief network and subsequent training of linear regressors to predict multi-dimensional A-V distributions.

Chapter 6: Machine Learning

This chapter will focus on machine learning methods for the development of a functional mapping from high-dimensional acoustic features to low-dimensional Arousal-Valence (A-V) representations. This chapter will first touch on classification systems and discuss a system using support vector machines (SVMs) for mapping audio into discrete emotion classes. As a result of the continuous nature of the space, the remainder of the methods presented are heavily biased towards regression, including support vector regression which will be shown to follow a training procedure similar to SVMs. In seeking to combine multiple feature domains to boost overall performance, the methods discussed here will focus on decision-level fusion, as well as multi-level classification and regression schemes.

Furthermore, given the continuous nature of musical emotion, it is also necessary to develop systems to predict how emotion *evolves* over time. In the sections following classification and regression, first a systems based approach will be discussed that uses a linear dynamical system (LDS) to model the evolution of emotion space parameters, and employs Kalman smoothing to estimate a sequence of emotion space parameters for an unknown testing example. Lastly, a method using conditional random fields (CRFs) is presented, which models emotion trajectories of label sequences as a progression through a discretized emotion grid. The trained CRF is capable of predicting emotion space heatmaps that can take on arbitrary distributions with an arbitrary number of modes.

6.1 Classification Methods

Support vector machines (SVMs) are the primary classification method discussed in this work because of their successful application to similar music classification tasks (e.g., artist and genre classification) [71]. Using kernel methods, SVMs can be used to construct non-linear decision boundaries, and they have proven to be very robust to noise [102, 103, 100].

6.1.1 Support Vector Machines

Motivated by the formulation of the perceptron, an SVM attempts to construct a classification hyperplane of maximal separation between two classes of data [102, 103, 100]. Figure 6.1 shows an example of a maximum margin hyperplane, shown as the solid line. The two parallel transport functions sit up against the data on either side of the hyperplane and are explained in more detail in the SVM tutorial found in Appendix C.

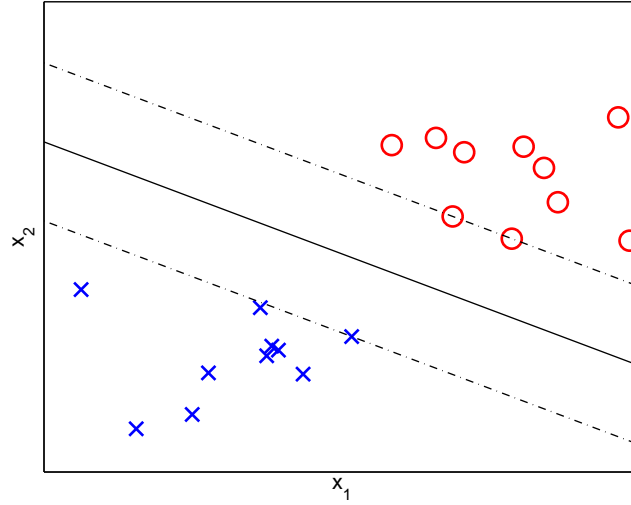


Figure 6.1: An example of a maximum margin classifier. In this example x_1 and x_2 are the dimensions of the input data, each blue X represents an example in the negative class and each red O represents an example in the positive class.

Kernel Methods

The objective when using kernel functions is to project the data into a space where it is linearly separable, which often involves using a non-linear projection function, resulting in a space of much higher dimension. There has been significant investigation of kernel methods with audio data, specifically to account for common acoustic features [71]. In all classification experiments discussed in this work, a distance-based radial basis function (RBF) kernel is used,

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2), \gamma > 0. \quad (6.1)$$

One property of the RBF kernel is that the Euclidean distance can be replaced by another distance metric, such as relative entropy between features represented as stochastic distributions [71]. These methods were investigated initially in this work but were not used due to decreased performance over standard RBF kernels. Additional details about kernel methods are available in the SVM tutorial in Appendix C.

6.1.2 Classifier Decision Level Fusion

Given the various feature domains described in Appendix A, it is also desired to have machine learning methods capable of combining them. The goal of decision level fusion is to combine decisions contributed by individual SVM classifiers on different feature types with the objective of obtaining better predictions overall (Figure 6.2). This section describes the three approaches used in the experiments in Chapter 7. For experiments containing more than two classes, multiple binary one-versus-all classifiers are used, combining these results to obtain a class estimate. Given a trained SVM decision function $f(\mathbf{x})$ (see Appendix C) estimates the class y for example i as follows:

$$\hat{y}_i = \arg \max_{c \in Y} f_c(\mathbf{x}_i), \quad (6.2)$$

where Y is the set of possible outcomes associated with y , and f_c is a function that assigns the likelihood of observation \mathbf{x}_i belonging to class c .

Maximum Distance

The first approach is to simply use the hypothesis of the classifier and feature domain that produces the greatest distance from the separation boundary, i.e., the one reporting the highest confidence.

Sum Rule

The next approach is to combine hypotheses by simply summing the distances (confidences) reported by each of the classifiers. The class receiving the highest ensemble confidence is considered to be the most likely source of the observed data.

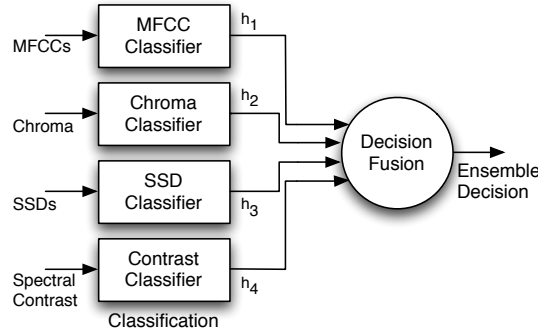


Figure 6.2: Ensemble-based decision-fusion system of differing acoustic feature types.

Decision Templates

Decision templates are a method of multi-layer (sequential) classification, specific to multi-class decision fusion. Each initial feature-based classifier is both trained and tested on the training data. A template matrix is created for each class, where the number of rows in each template corresponds to the number of classifiers, and the columns are the respective one-versus-all confidence values. In the testing phase, the testing hypotheses are mapped to a given decision template using Euclidean distance [104].

6.2 Regression Methods

Since the eventual goal of this work is to predict continuous A-V values, as opposed to emotion classes, this chapter also investigates multiple regression methods for mapping acoustic features into the A-V space. Initially, these experiments employ least-squares regression to determine the optimal mapping. Furthermore, also investigated is a mapping using support vector regression (SVR), in order to utilize non-linear regression functions, that may be more robust to noise than the linear functions used in least-squares.

6.2.1 Least-Squares Regression

Creating a linear projection from acoustic features to A-V coordinates is a straightforward example of multivariate least-squares regression. This section will first discuss singlevariate linear regression and then show how it can be easily translated to multiple parameters by making independence

assumptions. The vector of input data for a single example is denoted as $\mathbf{x}_i \in \mathbb{R}^{1 \times D}$, and the full training set as a matrix $\mathbf{X} \in \mathbb{R}^{N \times D}$, where N is the total number of examples and D is the dimensionality of the input space. Each scalar label value is denoted $y_i \in \mathbb{R}$, and the vector of all training labels $\mathbf{y} \in \mathbb{R}^{N \times 1}$. The goal of linear regression is simply to learn a mapping \mathbf{w} ,

$$y_i = \mathbf{x}_i \mathbf{w} + \mathcal{N}(0, \sigma^2), \quad (6.3)$$

where error is modeled as a zero mean Gaussian distribution. Using the full training set (\mathbf{X}, \mathbf{y}) , the residual error vector is denoted as \mathbf{r} ,

$$\mathbf{r} = \mathbf{y} - \mathbf{X}\mathbf{w}. \quad (6.4)$$

It is possible to find the maximum likelihood solution by taking the partial derivatives of the residual error with respect to the parameters, setting it to zero, and solving for \mathbf{w} [105]:

$$\frac{\partial \mathbf{r}^T \mathbf{r}}{\partial \mathbf{w}^2} = -\mathbf{X}^T (\mathbf{y} - \mathbf{X}\mathbf{w}) = 0, \quad (6.5)$$

$$\mathbf{w}_{LS} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}. \quad (6.6)$$

This same method can be applied to multivariate regression by assuming independence between the desired output dimensions. The output variable \mathbf{y} then becomes $\mathbf{Y} \in \mathbb{R}^{N \times K}$, where K is the number of output dimensions.

6.2.2 Support Vector Regression

Support vector regression (SVR) borrows much from its classification counterpart. Again, the hyperplane is expressed through its normal equation, where \mathbf{x} is the data, and \mathbf{w} is the normal vector with some offset b (see Equation C.1). Now, the norm of \mathbf{w} is minimized to ensure maximal flatness in the projection, while imposing an additional constraint that values projected from \mathbf{x}_i are within some range ϵ of the actual values y_i [77]:

$$\begin{aligned}
& \text{minimize: } \frac{1}{2} \|\mathbf{w}\|^2, \\
& \text{subject to: } \begin{cases} y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - b \leq \epsilon \\ \langle \mathbf{w}, \mathbf{x}_i \rangle + b - y_i \leq \epsilon \end{cases}.
\end{aligned} \tag{6.7}$$

Other than the addition of another regularization parameter, there is little change in the optimization problem, which is discussed in detail in Appendix C. In the implementations discussed in this work, these problems are optimized in the Lagrange dual form, as with SVMs (Equation C.9), using quadratic programming [106].

6.2.3 Regression Decision-Level Fusion

Just as with classification, it will be shown that combining information from multiple domains (in a more informed manner than simply concatenating the features) can potentially lead to more accurate projections.

Direct Fusion Methods

The first approach to fused regression consists of a system that simply combines the outputs of the individual feature regression systems. In the unweighted approach, outputs are simply averaged for each output dimension (i.e., emotion space dimension/parameter) from each individual feature regressor. This approach treats all features as equally informative for each output dimension, which is a naive assumption, and a weighted approach is therefore also investigated where each individual feature regressor is weighted by its ability to predict a particular emotion space dimension, determined through leave-one-out cross-validation.

Multi-level Regression

The next approach is a two-level regression scheme that feeds the outputs of individual regressors, each trained using distinct features, into a second-stage regressor to determine the final prediction. Two topologies are investigated, shown in Figure 6.3: in one case the secondary arousal and valence

regressors receive only arousal and valence estimates, respectively; in the other case, the secondary arousal and valence regressors receive both arousal and valence estimates from the first-stage. These two topologies will be referred to as *multi-level separate* and *multi-level combined*.

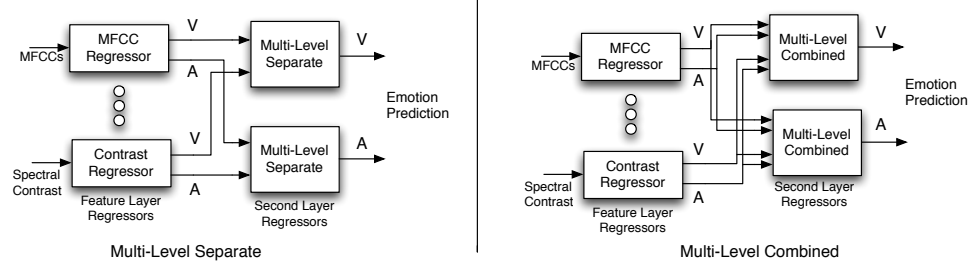


Figure 6.3: Multi-level regression topologies.

In all cases, the second-level regressors employ linear least-squares and are trained using a leave-one-out method (for every iteration, the first-stage regressors are trained by withholding one example, and the combined first-stage estimates from the withheld examples are used to train the second stage).

6.3 A Kalman Filtering Approach

When applying the previously discussed approaches to the problem of time-varying emotion prediction, emotion space parameters are predicted independently at each emotion space time analysis window, without considering the other time steps. In this approach, which was originally presented as part of this work in [10], musical emotion is modeled using a systems approach, employing a linear dynamical system (LDS). Using an LDS, it is possible to model both the mapping from features to emotion, as well as how the emotion parameters evolve over time. It is noted that this section will follow conventional systems notation. As the unknown parameter in LDS prediction and inference problems is the hidden state \mathbf{x} , and the output \mathbf{y} is known, this section will denote the emotion space parameters as \mathbf{x} and features as \mathbf{y} .

6.3.1 Estimating model parameters

In investigating time-varying models, this section will start with the simplest approach: to simply learn the time dependence of emotion labels \mathbf{x} and use that to restrict the way the emotion predictions

evolve. The goal is to learn a simple dynamics model A such that,

$$\mathbf{x}_t = A\mathbf{x}_{t-1}, \quad (6.8)$$

where $\mathbf{x}_t \in \mathbb{R}^{K \times 1}$ is a vector of emotion space parameters with dynamics matrix $A \in \mathbb{R}^{K \times K}$. Given this model, an emotion space mapping could be represented as a linear dynamical system as follows,

$$\mathbf{x}_t = A\mathbf{x}_{t-1} + w_t, \quad (6.9)$$

$$\mathbf{y}_t = C\mathbf{x}_t + v_t, \quad (6.10)$$

where the acoustic features are defined as $\mathbf{x}_t \in \mathbb{R}^{D \times 1}$ with matrix $C \in \mathbb{R}^{D \times K}$, which relates emotion space parameters to acoustic features. It is possible to estimate C through least-squares, just as was done in MLR (Equation 6.3), but it must function such that,

$$\mathbf{y}_t = C\mathbf{x}_t + v_t, \quad (6.11)$$

and driving noise w and observation noise v are defined as a zero mean Gaussian,

$$w \sim \mathcal{N}(0, Q), \quad (6.12)$$

$$v \sim \mathcal{N}(0, R). \quad (6.13)$$

The values for Q and R are estimated directly from the residuals of A and C .

In estimating the dynamics matrix A , initial experiments were performed using least-squares regression in the experiments in Chapter 7, but a constraint generation approach was used in the final approach. Using that approach, slightly better results were achieved, which is likely attributed to the fact that it guarantees a stable solution for A . Using constraint generation, the problem is formed as a convex optimization one and efficiently solved through quadratic programming [107].

Finally, because the LDS models a zero mean Gaussian process, the means of the labels and the features are subtracted prior to training, notated as $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$, respectively. These values are stored, such that $\bar{\mathbf{y}}$ can be removed from unlabeled features \mathbf{y} in the testing phase, and so $\bar{\mathbf{x}}$ can be used to apply the proper bias to the estimate of the labels \mathbf{x} .

6.3.2 Making Predictions using Kalman Smoothing

Given an unknown testing example \mathbf{y} , the known bias $\bar{\mathbf{y}}$ is removed, and an estimate of its emotion space distribution is formed by Kalman/RTS forward-backward inference [108, 109] (see Appendix D for equations).

6.4 Conditional Random Fields

This section will discuss the application of conditional random fields (CRFs) to the modeling of time-varying musical emotion. CRFs are powerful graphical models that are trained to predict the conditional probability $p(\mathbf{y}|\mathbf{x})$ for a sequence of labels \mathbf{y} given a sequence of features \mathbf{x} [110, 111]. Treating our features as deterministic, we retain the rich local subtleties present in the data, which is especially promising in content-based audio analysis where there is no shortage of rich data. Furthermore, the system provides a model of both the relationships between acoustic data and emotion space parameters, and also how those relationships evolve over time.

6.4.1 Definition

Conditional random fields are a type of log-linear model, similar to logistic regression, that are defined using feature functions that produce binary outcomes relating observations \mathbf{x} to states y . These functions are weighted by parameter λ_K , where K is the total number of binary features, each defined by its own feature function f_k . In this work, the linear-chain case of CRFs is investigated, which restricts features to apply only to the current and previous position of the labels, and thus the distribution of the CRF takes the form:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp \left\{ \sum_{k=1}^K \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\}. \quad (6.14)$$

The constant $Z(\mathbf{x})$ is referred to as the partition function and is a normalization function for a specific instance or sequence:

$$Z(\mathbf{x}) = \sum_{\mathbf{y}} \exp \left\{ \sum_{k=1}^K \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\}. \quad (6.15)$$

The CRF tutorial in Appendix E offers further detail on the training of CRFs.

6.4.2 Arousal-Valence Heatmaps

In applying CRFs to the problem of predicting emotion in music, instead of modeling the ambiguity of emotion *a-priori* and representing the distribution of our emotion space parameters as the ground truth, the training algorithm is presented with the individual user label sequences, thus allowing the model to learn the range of emotion responses to a given piece. In this application of the CRF it is also necessary to assign emotion space meanings to the states of the model, and in doing so each label in each sequence is discretized to an 11×11 grid. While this is a significant simplification, the findings in this work indicate that it provides sufficient granularity. As it is possible to compute the conditional probability of every state in the model, those “transitions” can be shown as an emotion-space heatmap. These heatmaps can model arbitrary modes and distributions, in contrast to the previous A-V distribution approach, which constructed uni-modal Gaussian A-V predictions. An example A-V heatmap is shown in Figure 6.4.

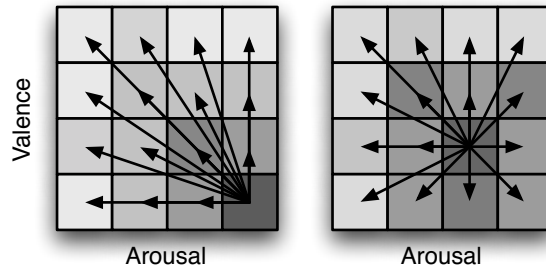


Figure 6.4: Heatmap visualization of CRF transition probabilities. Actual discretization is 11×11.

Chapter 7: MoodSwings Lite Experiments

This chapter will investigate a framework for content-based prediction of musical emotion using the MoodSwings Lite database, which was discussed in Chapter 3. As discussed in that chapter, the dataset consists of 240, 15-second music clips annotated with MoodSwings, a collaborative online game developed for music mood labeling [15]. The MoodSwings game board is based on the Arousal-Valence (A-V) space, where the valence dimension represents positive versus negative emotions, and the arousal dimension represents high versus low energy [4]. The game provides annotations at one-second intervals, reflecting the time-varying nature of musical mood, and delivers a distribution of labels across multiple players for a given song or even a moment within a song.

The first experiments briefly investigate quantizing the collected data into discrete emotion classes. These experiments will develop classification methods to relate acoustic data to the four quadrants of the A-V space. In order to take full advantage of the A-V space, the experiments thereafter formulate the problem as a regression one, developing a functional mapping from high-dimensional acoustic features to emotion space coordinates. This mapping is first implemented as a least-squares regression and later improved using support vector regression (SVR). The performance of the system in tracking the emotional content of music over short time windows is also demonstrated and later implemented in simplified form to be used as a simulated player AI for single-player MoodSwings games [6].

Perceptions of the emotional content of a given song or musical excerpt are bound to vary and reflect some degree of disagreement between listeners. Instead of viewing musical mood as a singular label or value, the next experiments investigate the modeling of emotional ground-truth as a *probability distribution*, potentially providing a more realistic (and accurate) reflection of the perceived emotions conveyed by a song. To perform the mapping from acoustic features to the A-V mood space, these experiments explore parameter prediction using multiple linear regression (MLR), partial least-squares (PLS) regression, and support vector regression (SVR). The data is modeled as

a two-dimensional Gaussian, and the goal is to be able to predict the A-V distribution parameters $\mathcal{N}(\mu, \Sigma)$ from the acoustic content. First, the effectiveness of the system is evaluated in predicting emotion distributions for 15 second clips, and as in the previous experiments, the analysis window length is then shorted to demonstrate its ability to follow changes in A-V label distributions over time [9].

While the experiments discussed thus far will be shown to have promising results, the goal of the next experiments is to improve time-varying musical emotion predictions using a systems approach. In these experiments, a linear dynamical system (LDS) is used, enabling the approach to model both the mapping from features to emotion space parameters, and the evolution of the emotion space parameters over time. As it will be found that the emotion space time-series distribution parameters do not evolve in the same way across the entire music corpus, a mixture approach is also developed, partitioning the data into different A-V clusters and using the oracle bound to prove that this will enhance overall performance [10].

The final experiments take on a different focus, turning to the topic of acoustic feature learning. Using deep belief networks (DBNs), these methods seek to learn emotion-specific features directly from short-time magnitude spectra [7]. Using a trained network, features are extracted as discussed in Chapter 5, where the intermediary layers of the network are used as features for a separate machine learning algorithm. The features are then compared, in terms of performance, to those features discussed thus far on the MoodSwings Lite database.

For both the classification and regression experiments, the MoodSwings Lite corpus is divided 70%-30% between training and testing samples. In the case of DBN training, a validation set is necessary to determine the stopping condition in training, and it is therefore split 50% training, 20% validation, and 30% testing. However, once the DBN is trained, both the training set and validation are combined. To avoid the “album-effect” [112, 113], any songs recorded on the same album are placed entirely in either the training or testing set. Additionally, each experiment is subject to multiple cross-validations, varying the distribution of training and testing data sets. For

initial experiments, 50 cross-validations are done, but due to the computational complexity of DBN training, that number is reduced to 5 for those experiments.

7.1 Music Emotion Classification

In the first experiments, classifying music into discrete emotion categories is briefly investigated [6]. The goal of this approach is the classification of each 15-second music clip into one of four discrete emotion classes, where each of the four classes corresponds to a quadrant of the A-V space. To perform the classification, four one-versus-all support vector machines (SVMs) are used, and in testing, the binary class resulting in the highest confidence is chosen as the correct one. The SVMs employ radial basis kernel functions in all experiments. The first experiments use singular acoustic features, and later experiments combine predictions based on multiple features using decision fusion methods.

7.1.1 Single Feature Classification

Shown in Table 7.1 are the 4-way classification results using the individual acoustic feature sets. As there is no expectation to find any single dominant feature for emotion classification, the next methods investigate combining the predictions of these individual feature classifiers to potentially obtain higher performance.

Feature Type	Accuracy
MFCC	$51.85 \pm 4.29\%$
Chroma	$41.05 \pm 5.05\%$
SSD	$37.89 \pm 4.76\%$
Spectral Contrast	$49.14 \pm 5.52\%$

Table 7.1: Single feature classification.

7.1.2 Decision Fusion Classification

In the decision fusion experiments, multiple methods are investigated for combining the hypotheses of our distinct classifiers, as discussed in Section 6.1; here each classifier is trained using a different acoustic feature set. Table 7.2 compares feature fusion using the methods of maximum distance, sum rule, and decision templates against the collective combination of all features (stacked). A $\sim 5\%$

classification accuracy increase is observed between the simple “stacked” feature combination and the sum rule method, the highest performing feature fusion method.

Using All Features	
Fusion Method	Accuracy
All Features Stacked	$42.27 \pm 4.59\%$
Max Distance	$45.70 \pm 5.64\%$
Sum Rule	$47.79 \pm 6.27\%$
Decision Templates	$44.77 \pm 5.57\%$
Using Only MFCCs & Spectral Contrast	
MFCC & S. Contrast Stacked	$51.03 \pm 4.76\%$
Max Distance	$52.90 \pm 4.53\%$
Sum Rule	$53.72 \pm 4.24\%$
Decision Templates	$53.07 \pm 5.49\%$

Table 7.2: Classifier fusion results.

Although all of the fusion methods easily outperform the “stacked” combination of features, their performance still lags behind that of the MFCC and spectral contrast (single feature set) classifiers. Considering the large gap in individual feature classification performance between the high performers (MFCCs and spectral contrast) and the low performers (chroma and spectral shape), the feature fusion experiments are reduced to include only the high performers. The lower-half of Table 7.2 shows that the removal of low performing acoustic features and combination of the remaining classifiers using the sum rule leads to a modest performance gain.

7.2 Music Emotion Regression

In dividing the data into discrete classes, clips for which A-V labels are in fact quite similar may be categorized into completely different classes. Such severe quantization of essentially continuous label data is likely the primary reason for the generally poor 4-way classification performance. Because of this, the next experiments focus on combining the same set of acoustic features with multiple regression methods to provide an alternative indication of musical mood [6]. In these experiments, for each input example \mathbf{x}_i , the model is trained to produce the emotion space parameter vector \mathbf{y}_i ,

$$\mathbf{y}_i = [\mu_a, \mu_v], \quad (7.1)$$

where μ_a and μ_v are the mean arousal and valence of the collected data, respectively.

Each training set was used to estimate regression parameters using both least-squares and SVR to create optimal projections from the mean acoustic features to A-V coordinates (averaging the labels across each 15-second music clip). There are many possible methods for evaluating regression performance. The first method looked at is the standard error of the estimate for each emotion space dimension (i.e., arousal and valence), where the estimate is denoted as \hat{y}_i , given the collected data y_i . $S_{y \cdot x}$ becomes an unbiased estimator of the true variance by dividing by the number of samples minus two [114],

$$S_{y \cdot x} = \sqrt{\frac{\sum_{i=1}^N [y_i - \hat{y}_i]^2}{N - 2}}, \quad (7.2)$$

where N is the number of testing examples.

The preferred performance metric, however, is the average Euclidean distance between the projected coordinates and the collected A-V labels, as a normalized percentage of the A-V space. To benchmark the significance of the regression results, these projections are compared to those of an essentially random baseline. Given a trained regressor and a set of labeled testing examples, an A-V prediction for each sample is first determined. The resulting distance to the corresponding A-V label was compared to that of another randomly selected A-V label from the test set. Comparing these cases over 50 cross-validations, a Student's T-test for paired samples is computed to demonstrate the statistical significance of the results. A lack of correlation (larger t values) between predicted and random values provides evidence of the statistical significance of the projected labels (beyond random guessing). In the results (Table 7.3), average distances are given from the projected coordinates to both the matching and randomly selected A-V labels.

7.2.1 Single Feature Regression

Without the need for forcing the data into discrete classes, qualitatively higher performance is achieved. For single feature sets, regression using least-squares regression and spectral contrast features results in the smallest average deviation (15.10%) from the mean labels of the test samples.

Feature/ Topology	Regression Method	Arousal Standard Error	Valence Standard Error	Average A-V Error	Average A-V Random Error	T-test
MFCC	LS	0.134 ± 0.009	0.125 ± 0.007	0.160 ± 0.008	0.241 ± 0.015	4.606
Chroma	LS	0.174 ± 0.016	0.124 ± 0.007	0.185 ± 0.008	0.223 ± 0.014	2.363
SSD	LS	0.139 ± 0.009	0.122 ± 0.007	0.165 ± 0.007	0.228 ± 0.013	3.774
S. Contrast	LS	0.117 ± 0.007	0.128 ± 0.014	0.151 ± 0.009	0.243 ± 0.014	5.309
MFCC	SVR	0.113 ± 0.008	0.113 ± 0.007	0.139 ± 0.007	0.243 ± 0.016	5.743
Chroma	SVR	0.163 ± 0.009	0.129 ± 0.008	0.184 ± 0.007	0.219 ± 0.011	2.033
SSD	SVR	0.143 ± 0.009	0.138 ± 0.009	0.175 ± 0.008	0.235 ± 0.012	3.435
S. Contrast	SVR	0.121 ± 0.007	0.120 ± 0.009	0.148 ± 0.007	0.234 ± 0.014	4.926
Stacked Features	LS	0.123 ± 0.009	0.122 ± 0.011	0.153 ± 0.008	0.257 ± 0.017	5.818
Fusion Unweighted	LS	0.120 ± 0.006	0.113 ± 0.006	0.148 ± 0.005	0.224 ± 0.012	4.956
Fusion Weighted	LS	0.119 ± 0.006	0.113 ± 0.006	0.147 ± 0.005	0.224 ± 0.010	4.915
M.L. Separate	LS-LS	0.114 ± 0.007	0.114 ± 0.006	0.144 ± 0.006	0.237 ± 0.014	5.573
M.L. Combined	LS-LS	0.113 ± 0.008	0.116 ± 0.007	0.144 ± 0.006	0.242 ± 0.017	5.707
Stacked Features	SVR	0.139 ± 0.013	0.128 ± 0.007	0.162 ± 0.008	0.241 ± 0.014	4.309
Fusion Unweighted	SVR	0.118 ± 0.007	0.116 ± 0.006	0.148 ± 0.006	0.221 ± 0.012	4.672
Fusion Weighted	SVR	0.117 ± 0.007	0.116 ± 0.006	0.147 ± 0.006	0.225 ± 0.012	4.967
M.L. Separate	SVR-LS	0.112 ± 0.007	0.114 ± 0.008	0.140 ± 0.007	0.238 ± 0.014	5.616
M.L. Combined	SVR-LS	0.112 ± 0.007	0.113 ± 0.007	0.139 ± 0.006	0.241 ± 0.016	5.786

Table 7.3: Regression results for fifteen second clips.

The corresponding T-test value for this case (5.309), given the degrees of freedom (72 test data samples), provides >99.99% confidence of statistical significance. Support vector regression using single acoustic features results in a smaller average deviation in almost all cases, with the highest performance (lowest average distance) achieved using MFCCs. Again, the T-test indicates very high confidence of statistical significance.

7.2.2 Regressor Fusion

In the fusion results, the performance for simply stacking features into one large feature vector is supplied to provide a comparison baseline for the other fusion methods proposed in Section 6.2.3. The direct fusion methods consist of unweighted and weighted methods for linearly combining the outputs of distinct regressors trained on separate feature sets. Both approaches outperform the concatenation of all features, as well as any of the individual feature sets for a given regressor (least-squares or SVR).

While the direct fusion methods provide some improvement, the multi-level regression methods offer greater performance gains when using least-squares regression for both stages. Conversely, the multi-level fusion results using SVR followed by least-squares do not offer any significant im-

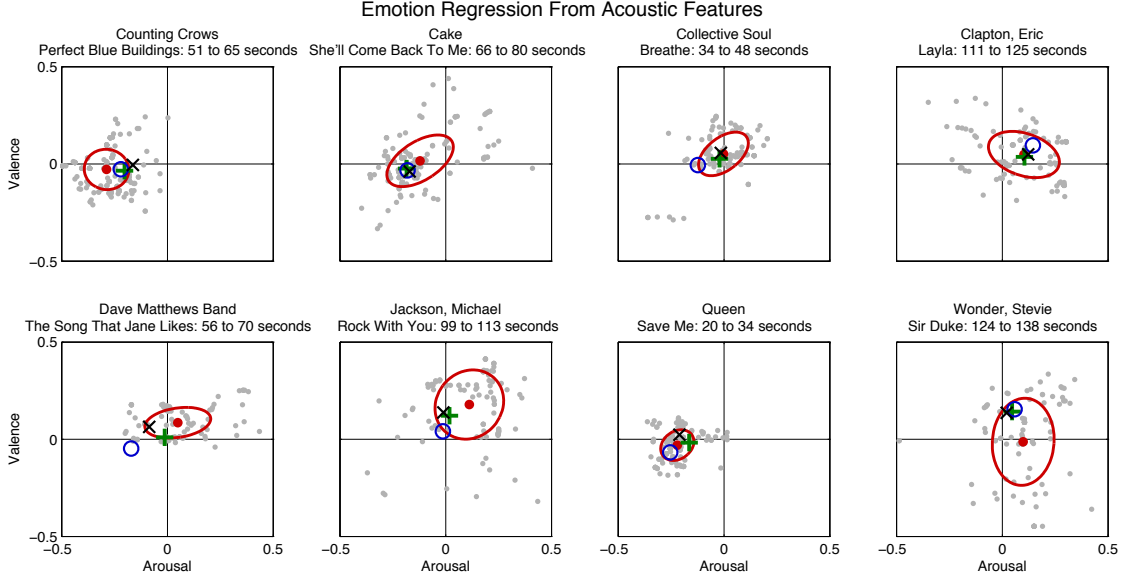


Figure 7.1: Collected A-V labels and projections resulting from regression analysis. A-V labels: second-by-second labels per song (gray \bullet), μ of collected labels (red \bullet), σ of collected labels (red ellipse). Projections: least-squares spectral contrast projection (green +), SVR MFCC projection (blue O), least-squares multi-level combined (black X).

provement over the best single-feature SVR. The LS-LS results approach the best case using SVR at substantially lower computational complexity, which is important for a potential real-time realization of the system (presented in Section 7.3.2).

7.2.3 Test Data Projections

The primary advantage of regression over classification is highlighted in the case of A-V labels close to an axis, where a highly accurate regressor can still lead to a misclassification according to the discrete class labels. Shown in Figure 7.1 are eight 15-second clips projected into the A-V space using multiple regression methods and acoustic features. Second-by-second A-V labels collected over the duration of the clip are indicated as a cloud of gray points. The performance of the regression can be evaluated both in terms of the distance from the mean of the collected labels, and also whether or not the regression label falls within the first standard deviation of the labels (shown as an ellipse).

7.3 Emotion Regression Over Time

For the time-varying approach, regressors are developed to predict the emotion for a given time (at one second intervals) using only current and past audio data [6]. In terms of the data collection, this implies that 15 examples are present for each 15-second music clip (for 240 clips this yields a total of 3600 examples). It is also desired to consider the optimal analysis window length (duration of past audio) for each acoustic feature set.

In Figure 7.2, regression analysis for each window length is performed from 1 to 45 seconds (in increments of one second) and average normalized emotion space distance is plotted from the projections to the collected labels. As in previous experiments, the training/testing data is split 70%/30% and is cross-validated 50 times. From the window length analysis in Figure 7.2, it can be seen that the optimal window length is not the same for all feature domains. For MFCCs, the most accurate prediction is obtained using 10 seconds of past feature data, 13 seconds for SSDs, 9s for spectral contrast, and 32s for chroma. These window lengths are used for the respective features in the regression analysis to follow.

In moving to per-second predictions, it can be seen from Table 7.4 that the overall error has increased slightly, as expected. The T-test values have increased as well, which is attributed to the overall increase in examples (from 240 to 3600). Considering the short-time degrees of freedom

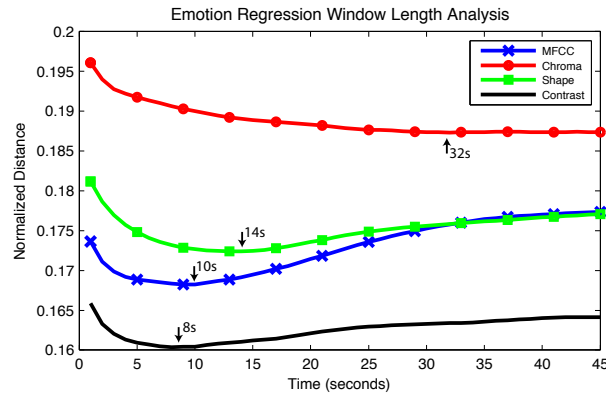


Figure 7.2: Window length analysis for different acoustic features.

Feature/ Topology	Arousal Standard Error	Valence Standard Error	Average A-V Error	Average A-V Random Error	T-test
MFCC	0.138 ± 0.007	0.131 ± 0.009	0.169 ± 0.007	0.245 ± 0.009	16.07
Chroma	0.177 ± 0.016	0.129 ± 0.009	0.192 ± 0.010	0.230 ± 0.014	8.606
S. Shape	0.146 ± 0.009	0.128 ± 0.007	0.173 ± 0.008	0.236 ± 0.009	13.68
S. Contrast	0.124 ± 0.006	0.127 ± 0.008	0.159 ± 0.006	0.248 ± 0.010	19.15
Stacked Features	0.121 ± 0.006	0.124 ± 0.007	0.154 ± 0.007	0.259 ± 0.010	21.85
Fusion Unweighted	0.123 ± 0.008	0.121 ± 0.007	0.155 ± 0.008	0.227 ± 0.009	17.18
Fusion Weighted	0.122 ± 0.008	0.121 ± 0.007	0.155 ± 0.008	0.227 ± 0.008	17.19
M.L. Separate	0.118 ± 0.008	0.122 ± 0.008	0.152 ± 0.008	0.248 ± 0.009	20.73
M.L. Combined	0.117 ± 0.008	0.122 ± 0.008	0.152 ± 0.008	0.249 ± 0.010	20.88

Table 7.4: Time-varying regression results.

(1080 testing examples), the lowest T value (8.606) produces confidence of statistical significance (vs. randomly selected projections) higher than 99.99%.

7.3.1 Test Data Projections

To visualize the projections over time, eight clips are chosen that display a clear shift in emotional content. In Figure 7.3, both the collected and projected emotion space labels are plotted for a 15-second selection for each clip at one second intervals. In these examples, the multi-level combined least-squares system is used, the highest performing system indicated in Table 7.4. The symbols for both the collected labels and the projected coordinates become darker over time, clearly depicting the common motion of both values.

7.3.2 MoodSwings “AI”

To further demonstrate the system and to allow for additional human feedback, the multi-level least-squares regression has been incorporated into a version of MoodSwings to compensate for a major issue with the game in that oftentimes no human partners are available online. Single-player games can be played using “partner” labels recorded from a prior game, but this eliminates songs for which annotations have not been previously recorded. The previous solution was to intersperse use of an “AI” partner that generates random labels centered around the player’s position, which is easy to detect and highly frustrating for a player. The low-complexity of the multi-level least-squares projection enables real-time implementation as the new game “AI,” where the system generates

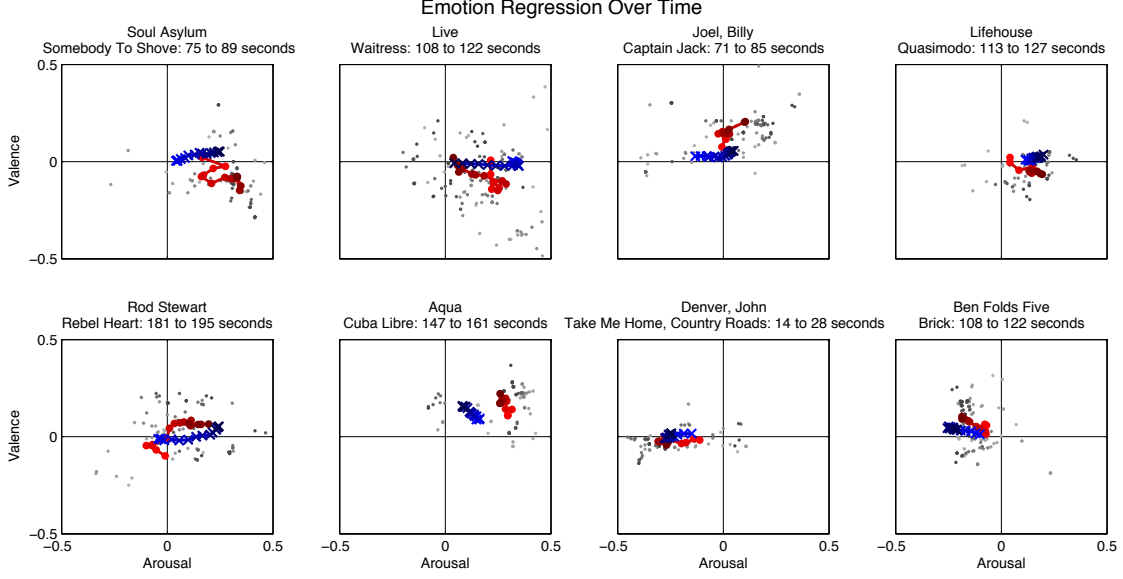


Figure 7.3: A-V labels and projections over time for eight example 15-second music clips (markers become darker as time advances): second-by-second labels per song (gray ●), mean of the collected labels over 1-second intervals (red ●), and projection from acoustic features in 1-second intervals (blue X).

the “partner” annotations at one-second intervals. This version, MoodSwings Single Player, is also available online [115].

7.4 Emotion Distribution Prediction

Human judgements are necessary for deriving emotion labels and associations, but perceptions of the emotional content of a given song or musical excerpt are bound to vary and reflect some degree of disagreement between listeners. In developing computational systems for recognizing musical affect, this lack of specificity presents significant challenges for the traditional approach of using supervised machine learning systems for classification. Instead of viewing musical mood as a singular label or value, the modeling of emotional “ground-truth” as a *probability distribution* potentially provides a more realistic (and accurate) reflection of the perceived emotions conveyed by a song [9].

These experiments will develop regressors to predict the parameterization vector \mathbf{y}_i of a two-dimensional Gaussian distribution in A-V space,

$$\mathbf{y}_i = [\mu_a, \mu_v, \sigma_{aa}^2, \sigma_{vv}^2, \sigma_{av}^2], \quad (7.3)$$

which encapsulates a fully parametrized Gaussian representation of the collected arousal and valence data, denoted as a and v , respectively.

7.4.1 Single Feature Emotion Distribution Prediction

Given the continuous nature of the problem, the prediction of a bivariate Gaussian within the A-V space, several methods are explored for multi-variate parameter regression. In these experiments, multiple linear regression (MLR), partial least-squares (PLS), and support vector regression (SVR) are used to create optimal projections from the acoustic feature sets described in Appendix A. For the initial distribution regression experiments, feature dimensions are averaged across all frames of a given 15-second music clip, thus representing each clip with a single vector of features. Preliminary experiments were performed using second- and higher-order statistics with the 15-second clips, but in all cases the inclusions of such data failed to show any significant performance gains.

There are many possible methods for evaluating the performance of the system. Kullback-Liebler (KL) divergence (relative entropy) is commonly used to compare probability distributions. Since the regression problem targets known distributions, our primary performance metric is the non-symmetrized (one-way) KL divergence (from the projected distribution to that of the collected A-V labels),

$$\text{KL}(p||q) \equiv \int p(x) \log \frac{p(x)}{q(x)} dx = E_p \left\{ \log \frac{p(x)}{q(x)} \right\}. \quad (7.4)$$

Since the comparison is between two emotion space Gaussian distributions, the KL divergence has closed-form,

$$\text{KL}(p||q) = \log \frac{|\Sigma_q|}{|\Sigma_p|} + \text{tr}(\Sigma_q^{-1} \Sigma_p^{-1}) + (\mu_q - \mu_p)^T \Sigma_q^{-1} (\mu_q - \mu_p) - d. \quad (7.5)$$

To provide an additional qualitative metric, the Euclidean distance between the projected means as a normalized percentage of the A-V space is also supplied. However, to provide context to KL values and to benchmark the significance of the regression results, the projections are again

Feature/ Topology	Regression Method	Average Mean Distance	Average KL Divergence	Average Randomized KL Divergence	T-test
MFCC	MLR	0.161 ± 0.008	4.098 ± 0.513	8.516 ± 1.566	5.306
Chroma	MLR	0.185 ± 0.010	5.617 ± 0.707	7.765 ± 2.135	5.659
S. Shape	MLR	0.167 ± 0.009	4.183 ± 0.656	7.691 ± 1.573	5.582
S. Contrast	MLR	0.151 ± 0.008	3.696 ± 0.657	8.601 ± 1.467	5.192
MFCC	PLS	0.155 ± 0.008	3.863 ± 0.56712	8.306 ± 1.389	5.540
Chroma	PLS	0.183 ± 0.010	5.286 ± 0.96019	7.146 ± 1.665	5.565
S. Shape	PLS	0.151 ± 0.008	3.770 ± 0.84026	8.278 ± 1.527	4.951
S. Contrast	PLS	0.151 ± 0.008	3.684 ± 0.644	8.700 ± 1.831	5.171
MFCC	SVR	0.140 ± 0.008	3.186 ± 0.597	7.744 ± 1.252	5.176
Chroma	SVR	0.186 ± 0.008	4.831 ± 0.737	6.466 ± 0.935	5.655
S. Shape	SVR	0.176 ± 0.008	4.611 ± 0.841	7.348 ± 1.025	5.251
S. Contrast	SVR	0.150 ± 0.008	3.357 ± 0.500	7.356 ± 1.341	5.301
Stacked Features	MLR	0.152 ± 0.007	3.917 ± 0.496	9.355 ± 1.879	5.737
Fusion Unweighted	MLR	0.149 ± 0.007	3.333 ± 0.433	6.785 ± 0.996	5.879
Fusion Weighted	MLR	0.147 ± 0.007	3.280 ± 0.423	6.803 ± 1.309	5.980
M.L. Seperate	MLR	0.147 ± 0.007	3.399 ± 0.478	8.235 ± 1.598	5.598
M.L. Combined	MLR	0.145 ± 0.007	3.198 ± 0.454	7.637 ± 1.389	5.551
Stacked Features	PLS	0.145 ± 0.006	3.403 ± 0.467	8.407 ± 1.635	5.543
Fusion Unweighted	PLS	0.145 ± 0.007	3.332 ± 0.508	7.123 ± 1.461	5.681
Fusion Weighted	PLS	0.145 ± 0.006	3.309 ± 0.501	7.160 ± 1.373	5.619
M.L. Seperate	PLS	0.145 ± 0.008	3.465 ± 0.577	8.426 ± 1.705	5.433
M.L. Combined	PLS	0.144 ± 0.007	3.206 ± 0.515	7.889 ± 1.656	5.485

Table 7.5: Distribution regression results for fifteen second clips.

compared to those of an essentially random baseline. Given a trained regressor and a set of labeled testing examples, an A-V distribution is estimated for each sample. The resulting KL divergence to the corresponding A-V distribution was compared to that of another randomly selected A-V distribution from the test set. Comparing these cases over 50 cross-validations, a Student's T-test for paired samples is computed to verify the statistical significance of the results.

From Table 7.5, it can be seen that the best performing single feature system is SVR with MFCC features at an average KL of 3.186. However, in both the MLR and PLS system, the highest performing single feature is spectral contrast with 3.696 and 3.684, respectively. As the main advantage of PLS over MLR is that it observes any correlation between dimensions in the multivariate regression, it is surprising that the performance difference between the two is nearly negligible. Given the degrees of freedom (72 test samples), even the lowest T-test value (5.171) produces confidence of statistical significance greater than 99.999%.

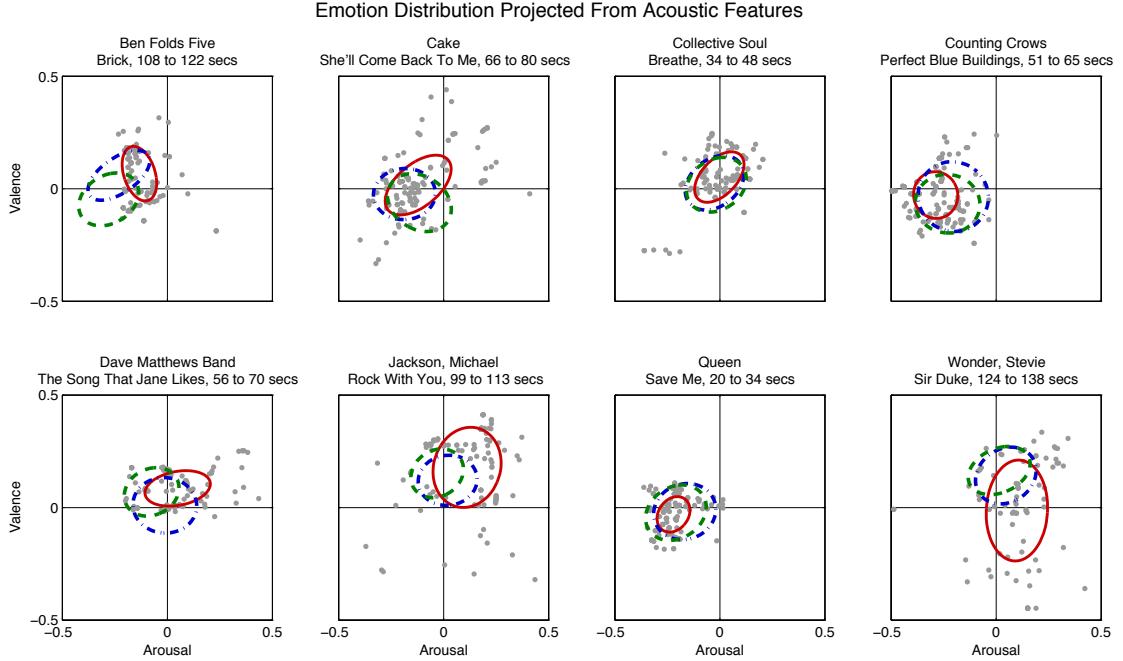


Figure 7.4: Collected A-V labels and distribution projections resulting from regression analysis. A-V labels: second-by-second labels per song (gray ●), Σ of collected labels (solid red ellipse), Σ of MLR projection from spectral contrast features (dash-dot blue ellipse), Σ of MLR Multi-Level combined projection (dashed green ellipse).

Shown in Figure 7.4 is the projection of six 15-second clips into the A-V space resulting from multiple regression methods and acoustic features. The standard deviation of the ground truth, and of each projection is shown as an ellipse. The performance of the regression can be evaluated in terms of the total amount of overlap between a projection and its ground truth.

7.4.2 Emotion Distribution Feature Fusion

While most individual features perform reasonably in mapping to A-V coordinates, a method for combining information from these domains (more informed than simply concatenating the features) could potentially lead to higher performance. In this section, the feature fusion approaches discussed in Section 6.2.3 are investigated for emotion distributions. Given the very small performance gains and high computational overhead of SVR, the focus is narrowed to MLR and PLS for these experiments. As the ultimate system will require many predictions over time in order to reflect emotional changes, the costs of SVR outweigh the benefits.

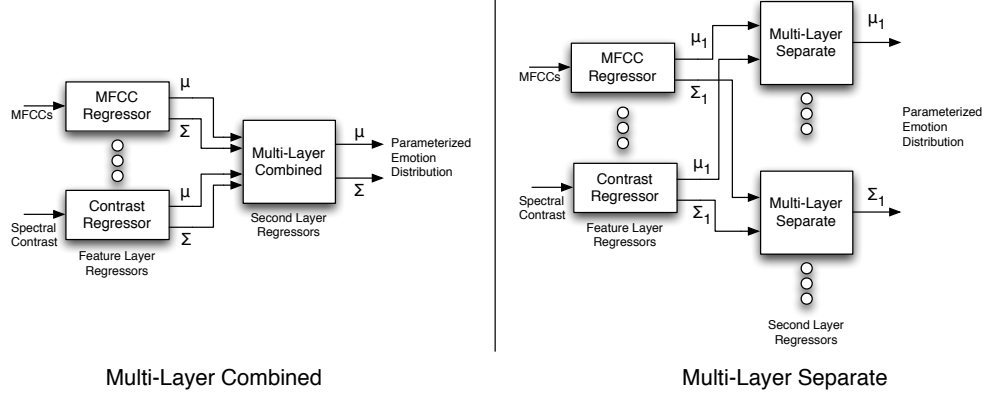


Figure 7.5: Multi-layer regression topologies.

In the fusion results, the performance for simply stacking features into one large feature vector is provided to give context to the other fusion methods. The more simple approach consists of a fusion system that is a combination of the outputs from the individual feature regression systems. In the unweighted approach, the parameter outputs are simply averaged from each individual feature regressor, and in the weighted approach each individual feature regressor is weighted by its ability to predict a particular parameter, which is determined by leave-one-out cross-validation.

In addition, a two-level regression scheme is developed by feeding the outputs of individual regressors, each trained using distinct features, into a second-stage regressor determining the final prediction. Two topologies are investigated (Figure 7.5): in one case, the secondary arousal and valence regressors receive only arousal and valence estimates, respectively; in the second case, the secondary arousal and valence regressors receive both arousal and valence estimates from the first-stage. Just as in the prior experiments, these two topologies are referred to as multi-layer separate and multi-layer combined. In all cases, the secondary regressors are trained using a leave-one-out method (on each iteration the first-stage regressors are trained by leaving one example out and using the estimates of that example from the first stage to train the second stage). The results for both cases are shown in Table 7.5.

7.5 Time-Varying Emotion Distribution Prediction

In attempting to predict the emotion distribution over time, the label observation rate is shortened to once per second and these experiments attempt to regress multiple feature windows from each song [9]. Just as in the previous time-varying experiments, this means that for each 15-second clip there are 15 examples, increasing the total corpus to 3600 examples. Of course, for any experiment, multiple examples from the same song must be either all in the training or testing set. In addition, as it is clear that some past data may be necessary to accurately determine the current emotional content, past features are included and the optimal feature window length is investigated.

Given the similar performance of MLR and PLS in fusion methods, for the short time analysis the experiments will be restricted to only the MLR methods. The similarity in performance is likely due to the fact that in the multi-layer combined system, both MLR and PLS are able to account for the correlation between label dimensions.

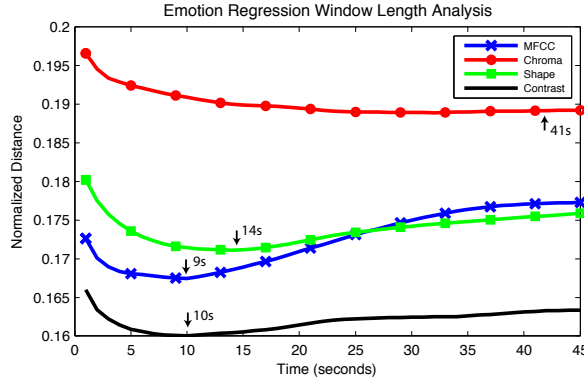


Figure 7.6: Window length analysis for different acoustic features.

For the time-varying approach, regressors are developed to predict the emotion for a single second using only current and past audio data. In terms of the data collection, this implies that there are 15 distributions for each 15-second music clip (for 240 clips this yields a total of 3600 distributions). Just as in the prior experiments with A-V means, the optimal analysis window length for regression is investigated for each acoustic feature set. In Figure 7.6, a regression analysis is performed for each window length from 1 to 45 seconds (in increments of one second). It can be seen that the optimal window length is not the same for all feature domains. For MFCCs, the most accurate prediction

Feature/ Topology	Average Mean Distance	Average KL Divergence	Average Randomized KL Divergence	T-test
MFCC	0.169 ± 0.007	14.61 ± 3.751	27.00 ± 10.33	10.77
Chroma	0.190 ± 0.007	18.71 ± 6.819	22.53 ± 6.984	9.403
S. Shape	0.173 ± 0.007	15.46 ± 6.402	24.61 ± 9.220	11.06
S. Contrast	0.160 ± 0.006	13.61 ± 5.007	27.29 ± 9.861	10.23
M.L. Combined	0.154 ± 0.006	13.10 ± 5.359	28.39 ± 10.35	10.08

Table 7.6: Distribution regression results for short-time (one-second) A-V labels.

is obtained using 13 seconds of past feature data, also 13 seconds for SSDs, 15 seconds for spectral contrast, and 41 seconds for chroma. This has changed slightly from the previous experiments (Figure 7.2) because additional label data was collected between the time the experiments were conducted. These feature window lengths are used in the regression analysis to follow.

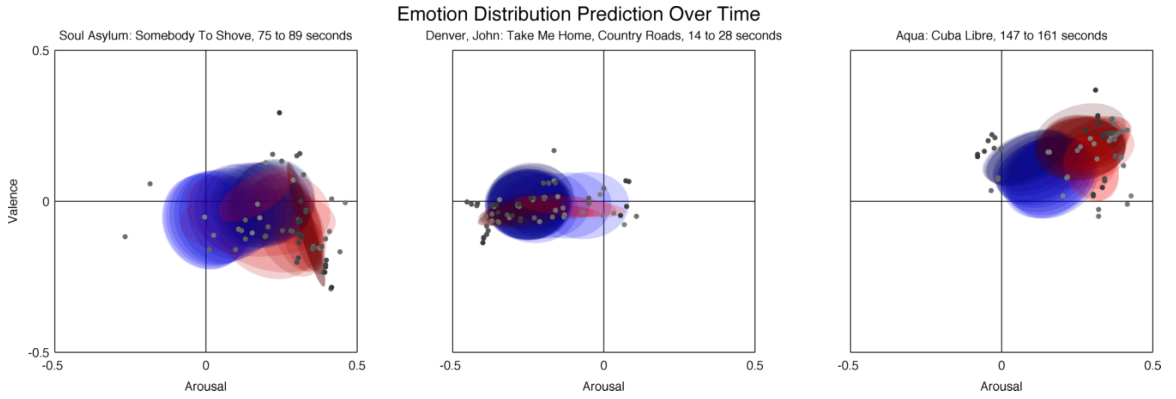


Figure 7.7: Time-varying emotion distribution regression results for three example 15-second music clips (markers become darker as time advances): second-by-second labels per song (gray \bullet), Σ of the collected labels over 1-second intervals (red ellipse), and Σ of the distribution projected from acoustic features in 1-second intervals (blue ellipse).

In moving to short-time labels, it can be seen from Table 7.6 that the overall KL has increased, but the average distance ratings have mostly remained the same. This is most likely attributed to the fact that the underlying label covariance is less consistent due to the smaller quantity of collected A-V labels. The T-test values have increased as well, which can be attributed to the overall increase in examples (from 240 to 3600). Considering the short-time degrees of freedom (1080 testing examples), the lowest T value (9.403) produces confidence of statistical significance (vs. randomly selected projections) higher than 99.999%. To visualize emotion regression over time,

three clips are shown that display a clear shift in emotion distribution, plotting both the collected and projected distributions at one second intervals (Figure 7.7).

7.6 A Kalman Filtering Approach

The systems used in the time-varying experiments thus far made predictions at each time slice independently. In this section, a linear dynamical systems (LDS) approach is used to estimate the temporal evolution of emotion space distributions with Kalman/Rauch-Tung-Striebel (RTS) smoothing. In the first set of experiments, a single LDS model is used, and in later experiments mixture systems are investigated to establish an “oracle bound” for the performance upper limit [10]. Furthermore, as this section will focus on the relative performance of different models, these experiments will be constrained to only a single feature domain, which is chosen to be spectral contrast as it achieved the highest performance on most prior experiments.

7.6.1 Kalman Data Preprocessing

In the previous sections, the parameters of the ground truth distributions were estimated at each time instance independently, computing the sample mean and minimum variance unbiased estimator of the covariance [9]. While the assumption of temporal independence allows estimation methods that are computationally efficient, it ignores the time-varying nature of music and undoubtedly introduces noise. Furthermore, the temporal starting point for a MoodSwings match is randomly selected for each game, so a particular labeler does not necessarily have the opportunity to weigh in on all 15 time intervals in the clip. As a result, there are often different sample sizes at each interval. For example, see the left-hand plot of Figure 7.8: shown in gray are the individual second-by-second labels collected from the MoodSwings game, and in red ellipses are the estimates of the distribution; both become darker as time progresses. While the general drift of the distribution is captured, there is a heavy amount of noise in the covariance ellipses.

To address this issue and form a better approximation of the ground truth distribution, a pre-processing step is applied using a Kalman/RTS smoother [108, 109]. Using this approach the labels

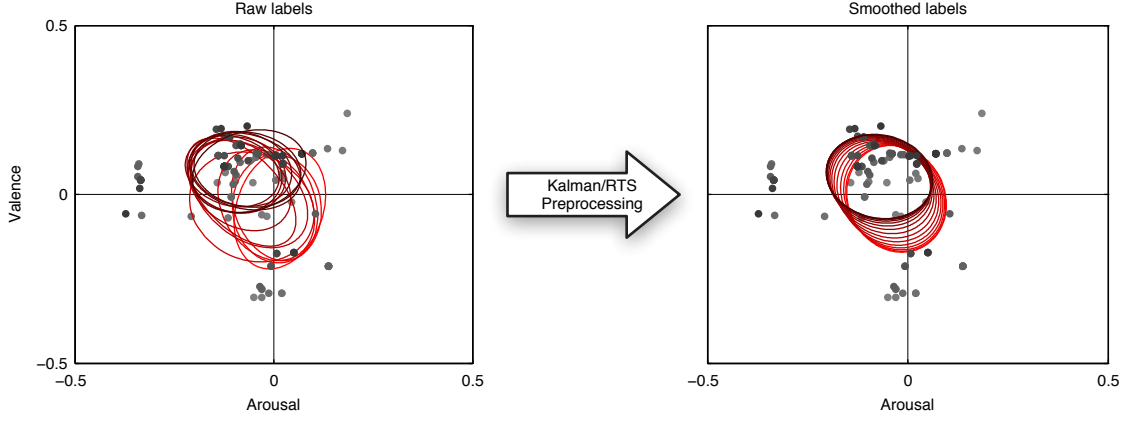


Figure 7.8: Emotion label preprocessing: gray dots indicate individual second-by-second labels collected from the MoodSwings game. Red ellipses are the estimates of the distribution; both become darker as time progresses.

\mathbf{y} are modeled as noisy observations of the true distribution of \mathbf{x} ,

$$\mathbf{x}_t = \mathbf{x}_{t-1} + w_t, \quad (7.6)$$

$$\mathbf{y}_t = \mathbf{x}_t + v_t. \quad (7.7)$$

Gaussian noise sources w and v are parametrized experimentally, representing the interfering noise corrupting the labels, which are assumed to be zero mean,

$$w \sim \mathcal{N}(0, Q), \quad (7.8)$$

$$v \sim \mathcal{N}(0, R). \quad (7.9)$$

Thus, the values of Q and R are chosen experimentally to provide the desired amount of smoothing. Defining the initial conditions as the mean and covariance of the noisy labels, Kalman smoothing is performed to estimate the clean distribution \mathbf{x} of the noisy process \mathbf{y} . The Kalman smoothing equations are discussed in detail in Appendix D. It is important to note that in this special case of the Kalman filter, because the observation matrix C and dynamics matrix A have been omitted, they simply reduce to identity matrices in the standard Kalman/RTS equations.

Feature/ Topology	Average Mean Distance	Average KL Divergence	Average Randomized KL Divergence	T-test
MLR Noisy [9]	0.169 ± 0.007	14.61 ± 3.751	27.00 ± 10.33	10.77
MLR	0.160 ± 0.007	4.576 ± 0.642	9.531 ± 1.856	18.35
Kalman	0.160 ± 0.007	4.650 ± 0.652	8.870 ± 1.633	16.85
MLR Mixture	0.109 ± 0.007	3.179 ± 0.539	12.00 ± 1.899	19.68
Kalman Mixture	0.109 ± 0.007	2.881 ± 0.568	12.34 ± 1.973	19.63

Table 7.7: Results for emotion distribution prediction over time.

The right half of Figure 7.8 shows the smoothed version of the estimate, which provides a much more reasonable representation of the true distribution. It can be seen that the distribution moves consistently forward, and the unrealistic movement in the covariance ellipses has been removed.

7.6.2 Multiple Linear Regression

MLR results on the smoothed data are provided for benchmarking. Shown in Table 7.7 are the results, where MLR has an average error in the mean prediction of 0.160 in the normalized space. This is a significant improvement over previous approaches, with an error of 0.169 on average (Table 7.6). This improvement can be directly attributed to the fact that there was no label preprocessing. However, the largest improvement can be seen in the KL-divergence values, which have been reduced from 14.61 (Table 7.6) to 4.576 on average.

7.6.3 Kalman Filtering

Applying the Kalman filtering approach presented in Section 6.3 provides quite reasonable results as well. The total error is similar to MLR, but this result is not surprising due to the fact that the evolution model has been restricted to a single A matrix, which is a significant simplification. While the average distance remains identical, the increase from 4.576 to 4.650 in KL is a nearly negligible change. As this demonstrates that all of our clips do not fit easily into exactly the same dynamics model, it makes a strong case for mixture systems.

Emotion Distribution Prediction Using Kalman Mixtures

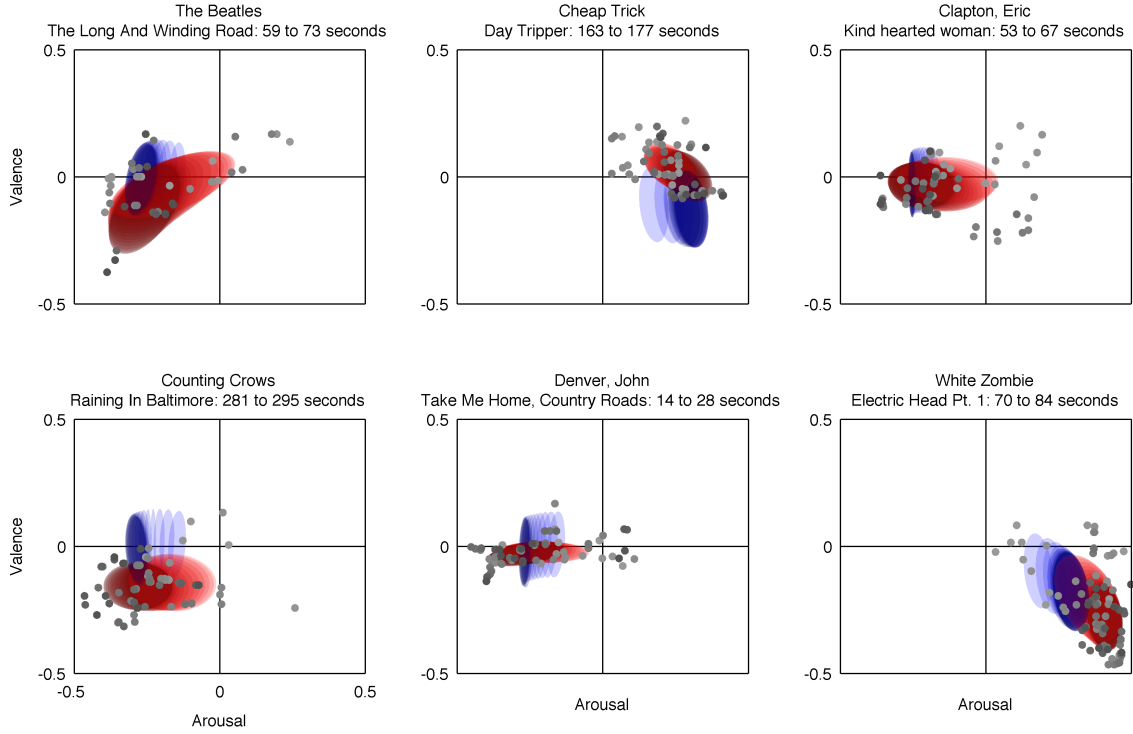


Figure 7.9: Emotion label preprocessing: gray dots indicate individual second-by-second labels collected from the MoodSwings game, red ellipses indicate the estimates of the distribution, and blue ellipses indicate the predictions using Kalman filter mixtures; all three become darker as time progresses.

7.6.4 Kalman Filter Mixtures

While a single LDS makes for an interesting approach to model the temporal dependence in music emotion prediction, it is not expected that the dynamics in all 240 clips can be represented with a single A matrix, and it may be possible to achieve better results with more than one C . In this approach, multiple systems are trained and an “oracle bound” is established for the prediction of musical emotion using mixtures, which is compared to using MLR on the same clusters. An assignment variable z is now added,

$$\mathbf{x}_t = A_z \mathbf{x}_{t-1} + w_t, \quad (7.10)$$

$$\mathbf{y}_t = C_z \mathbf{x}_t + v_t. \quad (7.11)$$

To form the mixtures, k-means clustering is performed on a three dimensional vector for each sequence: the average arousal, valence, and flux. The flux is computed between each time instance vector in the label sequences, which as previously discussed is made up of the arousal and valence means and covariance at one second intervals. This allows clustering to occur both spatially and dynamically.

As in the previous experiments, the feature means $\bar{\mathbf{y}}$ are computed across all clusters and saved for removal before both training and testing. In the case of the labels, a specific mean is computed for each cluster $\bar{\mathbf{x}}_z$, which is removed before training and saved to bias testing estimates for the corresponding cluster.

The Kalman mixture provides the best result of any system; using only four clusters an average KL of 2.881 is achieved, which is a significant improvement over both the MLR system at 4.576 and the MLR mixture at 3.179. In terms of mean error, however, Kalman and MLR mixtures produce nearly identical results, with normalized distances of about 0.109.

Shown in Figure 7.9 are the emotion space predictions of six fifteen second clips using this method. Shown in gray are the individual A-V ratings collected from the MoodSwings game, in red are the distribution estimates, and in blue are the predictions from acoustic features; all three become darker as time progresses.

7.7 Emotion Regression with Learned Features

Using the MoodSwings Lite dataset, this section presents a preliminary investigation into the use of deep belief networks (DBNs) feature learning [7]. An explanation of DBN training and application to feature learning can be found in Chapter 5. In these experiments, learned feature representations are compared to other state-of-the-art feature representations investigated in prior experiments, with the addition of the Echo Nest Timbre features. DBN hidden layer outputs are used as features to predict the training labels using a separate linear regression model. The features generated by the second layer of the DBN are shown to outperform all other features in terms of mean A-V error, and the topology is especially promising in providing insight into the relationship between acoustic data and emotional associations via a physical structure. The final model was demonstrated on five

training/verification/testing cross-validation permutations, which is something that had not been investigated in prior approaches due to the time required for training.

7.7.1 DBN Training

In the following experiments, a regression-based deep belief network is investigated for feature learning on magnitude spectra. Magnitude spectra is computed at approximately 20 msec window lengths, and the labels are up-sampled to match that same rate. All experiments use 3 hidden layers, each containing 50 nodes. Restricted Boltzman machine (RBM) pre-training is run for 50 epochs with a learning rate of 0.001. Conjugate gradient fine-tuning is used, and during that stage an additional multiple linear regression (MLR) layer is attached to the output of the DBN. As this stage is supervised, for each input example \mathbf{x}_i , the model is trained to produce the emotion space parameter vector \mathbf{y}_i ,

$$\mathbf{y}_i = [\mu_a, \mu_v]. \quad (7.12)$$

For each cross-validation, the MoodSwings Lite dataset was split such that 50% of the data was used for training, 20% for verification, and 30% for testing. To speed up the computation of our DBNs, the training was run on StarCluster [116], a Python-based tool for creating and managing computing clusters on Amazon’s Elastic Compute Cloud (EC2) [117]. Using StarCluster, several hundred parameterizations of the network were tried before settling on the ones described above.

7.7.2 DBN Features in Emotion Prediction

From Table 7.8, it can be seen that the DBN features performed very well overall in music emotion recognition. For each cross-validation in the experiment, the DBN was trained on the training set, using the verification set for fine-tuning. After model training, individual layer outputs were computed for the combined training and verification sets and then used for the training of a separate MLR regression system. Overall, Layer 2 of the DBN performs best in terms of mean error, validating the use of a deep representation.

KL-divergence for the Gaussian ground-truth representation used in prior work is also shown [9, 10], where regressors are developed to predict the parameterization vector \mathbf{y}_i of a two-dimensional

Feature/ Topology	Average Mean Distance	Average KL Divergence
DBN Layer 1	0.168 ± 0.006	5.73 ± 1.09
DBN Layer 2	0.158 ± 0.004	5.41 ± 1.12
DBN Layer 3	0.158 ± 0.005	5.45 ± 1.20
Spectral Contrast	0.161 ± 0.003	5.04 ± 0.688
MFCCs	0.167 ± 0.004	5.32 ± 0.737
Chroma	0.195 ± 0.007	7.89 ± 1.25
SSDs	0.180 ± 0.006	6.09 ± 1.02
ENTs	0.174 ± 0.004	5.69 ± 0.742

Table 7.8: Results for emotion distribution prediction over time.

Gaussian in A-V space,

$$\mathbf{y}_i = [\mu_a, \mu_v, \sigma_{aa}^2, \sigma_{vv}^2, \sigma_{av}^2]. \quad (7.13)$$

It is noted that, for these cases, spectral contrast performs better, but it is maintained that mean error in the arousal-valence space is significantly more important than KL divergence. For instance, KL-divergence may punish an incorrectly rotated covariance equal to a large mean distance, but in terms of emotion prediction the covariance rotation is much less important than the bias in the mean.

7.7.3 Relating DBN Features to Frequency Content

This deep learning approach is promising not just for the features it generates, but because it offers a mathematical model that represents the relationship between raw frequency content and human emotion. In this section, the learned mappings are investigated in terms of sparsity and shape, as well as the projection of the learned feature representations back to the magnitude spectra domain. The input to the DBN for these examples is a magnitude spectrogram from the classification experiments (Figure 7.10), reduced to 200 frames. Note that the log-magnitude of the spectrum is taken here for illustration purposes, but as with all previous experiments, only the magnitude was used as the DBN input.

The learned feature representations for the individual DBN layers are shown below in Figure 7.11. For the given sample, highly sparse features are observed at the first layer, and it is expected

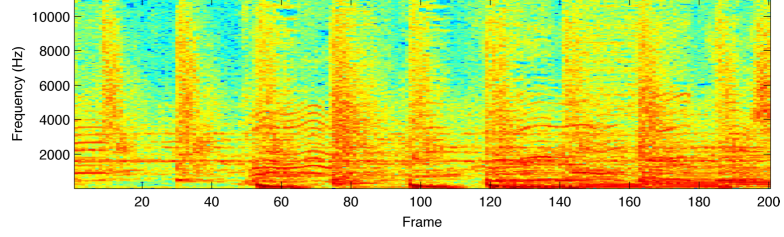


Figure 7.10: Log view of DBN magnitude spectra input.

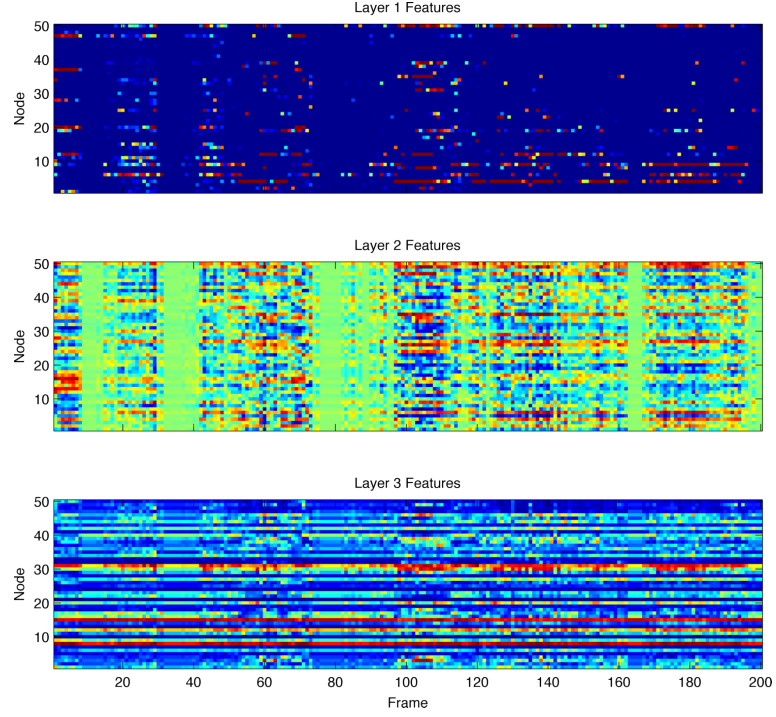


Figure 7.11: DBN hidden layer outputs.

that the learned mappings from magnitude spectra are heavily dependent on sparse frequency bands. Layers two and three show much more detail, which is expected due to their higher performance in the emotion prediction experiments.

Next, the reconstruction of the original spectrogram from the layer 1 output is investigated. The output y of the first hidden layer is simply the projected data with a logistic function applied,

$$\mathbf{y}_{\text{layer1}_i} = 1/(1 + \exp(-\mathbf{x}_i \mathbf{w} - \mathbf{b})), \quad (7.14)$$

and thus the original spectral data x can easily be reconstructed as follows,

$$\mathbf{x}_{\text{reconstructed}_i} = \left(-\ln \left(\frac{1}{\mathbf{y}_{\text{layer1}_i}} - 1 \right) - \mathbf{b} \right) \mathbf{w}^{-1}, \quad (7.15)$$

where $1/\mathbf{y}_{\text{layer1}_i}$ is element-wise division.

Shown in Figure 7.12 is the heatmap for the reconstructed \mathbf{x} . In looking back at frequency content, we see an interesting sparse representation. The reconstruction reveals that the learned representations are heavily dependent upon energy in specific frequency bands.

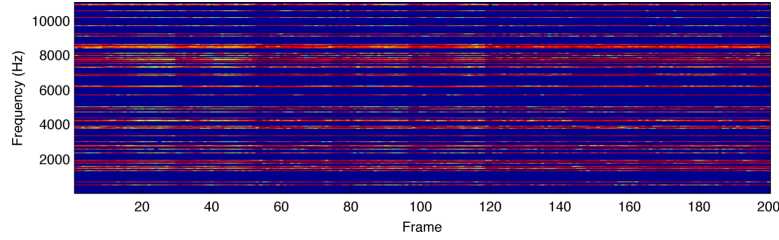


Figure 7.12: Log view of DBN magnitude spectra reconstruction.

In observing the layer 1 outputs in Figure 7.11, and their spectral reconstruction in Figure 7.12, it is impressive that the DBN is able to learn informative features from such a sparse representation of the spectrum.

7.8 Discussion

In working with a continuously labeled space such as A-V, it is clear that regression provides a more informative result than classification and is less sensitive to small variations (e.g., near the quadrant boundaries). In examining acoustic features for classification and regression, spectral contrast and MFCCs consistently provided the best performance across the classification and regression tasks, while spectral shape and chroma did not perform to expectations. MFCCs and spectral contrast capture different aspects of frequency-domain variation, so it is somewhat surprising that the combination of the two features improved classification but not regression performance. This is likely due to the “curse of dimensionality” (exponentially more data is required to fill a volume in feature space as more dimensions are added). Thus, adding feature dimensions, rather than leading to more informed decisions, could have hindered overall performance. The relative scaling of different

features also presents problems, since variations in magnitudes may lead to artificially inflated or reduced contributions from particular features.

The regression results are quite promising, even using the most elementary techniques (least-squares). In working with highly subjective emotional labels, where there is not necessarily a singular rating for even the smallest time slice, it is clear that it is possible to construct a more accurate system (in terms of predicting actual human labels) by representing the ground truth as a distribution. While accounting for potential dependence between the distribution parameters in the A-V emotion space seemed to be of high importance, some of the best performing techniques assumed total independence of parameters. In particular, combining MLR in multiple stages produces results comparable to more computationally complex methods.

In estimating the Gaussian distribution parameters from the collected data, it is clear that much more robust estimates of the distribution and how it evolves over time can be achieved with the Kalman approach. Given a limited number of samples at each time step, it is necessary to take into account samples from past (or future) observations if available. In looking to improve the collected data distribution, future work should explore beyond these Gaussian assumptions. While the Gaussian assumption provides an interesting representation of the data, it is in fact a naive assumption, as there is no reason to expect A-V ratings to follow a bivariate Gaussian for every second of every song. The true distribution may vary in terms of family or number of modes across different songs or even different moments within the same song.

In terms of automatically parameterizing these time-varying distributions from acoustic content, the Kalman approach has shown to be somewhat promising, yielding the ability to model and track emotion space distribution changes over time. Through the “oracle bound,” it has been shown that using mixtures, it is possible to more accurately model both the different spacial clusters as well as dynamics in the A-V space. In the future, it would be necessary to develop methods to accurately predict the cluster of an unknown testing example. This problem could be solved through traditional classification approaches, though added difficulty is presented in terms of clusters that are often of unequal sizes (especially in terms of dynamics clustering), and making decisions based on

the prior negates the benefit of those added clusters. It is possible that, given the limited dynamics representation of a linear dynamical system, it may be necessary to move to a more complex graphical model to obtain a better representation of this data.

The deep belief network is a powerful topology for music emotion recognition for both learning informative feature domains, as well as providing insight into the direct relationship between emotion and acoustic content. The overall performance could potentially be increased by more advanced regression algorithms that are more robust to the high dimensionality of the data. Furthermore, other optimization methods for fine-tuning the deep structure could be investigated, as well as alternate error metrics. This approach, which models the output of the regression layer as a single point in the A-V space, could be expanded to a metric that provides better knowledge of the emotion space distribution.

Chapter 8: MoodSwings Turk Experiments

This chapter will discuss experiments using the MoodSwings Turk database in music emotion recognition. The MoodSwings Turk database was collected with Mechanical Turk, an online crowdsourcing service provided by Amazon [86]. The initial purpose of investigating MTurk was to provide a dataset collected through more traditional means to assess the effectiveness of the game, specifically to determine any biases created through collaborative labeling. Working with Mechanical Turk presented myriad challenges in the area of data filtering, as the inclusion of monetary incentive attracted a great number of individuals looking to game the system. Chapter 3 discussed the many rules that were put in place to identify these workers. However, even after filtering out a good portion of the collected data, MTurk still attracted a much larger number of participants than MoodSwings, with the final dataset containing 4,064 label sequences in total, 16.93 ± 2.690 ratings per song. As a result of this dense labeling, the MoodSwings Turk database became the primary label set for future experiments. Furthermore, this dataset has also been made publicly available to the research community [118].

The first experiments discussed in this chapter are essentially a repeat of the MoodSwings Lite time-varying emotion distribution prediction experiments (Section 7.5). The goal of these experiments is to verify similar performance to that of the original MoodSwings Lite dataset. It is desired that the different features achieve similar performance, and also rank in terms of that performance in a similar order [8].

In the next set of experiments, this chapter will examine application of conditional random fields (CRFs) to the modeling of time-varying musical emotion. CRFs are powerful graphical models that are trained to predict the conditional probability $p(\mathbf{y}|\mathbf{x})$ for a sequence of labels \mathbf{y} given a sequence of observed data (e.g., acoustic features) \mathbf{x} . As CRFs are trained on the data of individual users, the fact that complete sequences are available from every annotator and the dense annotation of MoodSwings Turk makes it a good candidate for evaluating these methods [11].

This chapter will next investigate feature learning methods with the MoodSwings Turk database. The first feature learning experiments will essentially be a repetition of those performed on the MoodSwings Lite dataset (Section 7.7), and similar performance will be confirmed. Next, in looking to improve learned feature performance, the next feature learning methods will be modified to learn features from multiple time steps. The first feature learning experiments relied on a single ~ 20 msec frame of audio in order to provide comparison to standard features (e.g., MFCCs), but humans certainly cannot perceive emotion in such a short clip, and therefore extraction of such information should be performed over a much larger window. In these experiments, aggregations of past spectral data are included as input to the DBN. Furthermore, in looking to improve pretraining performance, a universal background model (UBM) approach is discussed, where pretraining occurs on a much larger set of unlabeled data (~ 8000 songs) than is available in MoodSwings Turk [13, 14].

Finally, this chapter will examine the use of learned features with conditional random fields. Both the features from the single frame approach as well as the multi-frame approach with UBM pretraining will be evaluated using CRFs. Overall it will be shown that these features work well in CRF training, despite their overwhelmingly large dimensionality.

8.1 Emotion Prediction with MTurk Data

These experiments seek to further establish correlation between the MoodSwings Lite and MoodSwings Turk datasets. The MoodSwings Turk label set is used in this section for the problem of time-varying emotion prediction experiments that was discussed in Section 7.5. The desired outcome is to identify similar performance overall, as well as relative performance in the ranking of the individual features.

Just as in the prior experiments, the systems utilize supervised machine learning algorithms to map A-V labels to content-based audio features (e.g., MFCCs). Prediction performance for each feature, as well as combined performance using a multi-layer regression method for late-feature fusion, using multiple linear regression (MLR) is shown in Table 8.1.

These results can be compared to Table 7.6, which contains the results using the MoodSwings Lite labels on the same tasks. The results are similar: all features rank in the same order, and in terms of overall mean distance there is only slight improvement for the MTurk dataset. In terms of

Feature/ Topology	Average Mean Distance	Average KL Divergence	Average Randomized KL Divergence	T-test
MFCC	0.143 ± 0.007	1.501 ± 0.148	2.801 ± 0.294	20.68
Chroma	0.181 ± 0.008	3.555 ± 0.302	3.897 ± 0.313	21.08
S. Shape	0.158 ± 0.007	1.733 ± 0.172	2.501 ± 0.246	23.51
S. Contrast	0.141 ± 0.007	1.486 ± 0.158	2.821 ± 0.297	21.17
M.L. Combined	0.130 ± 0.006	1.308 ± 0.132	2.928 ± 0.310	20.52

Table 8.1: MLR results for short-time (one-second) A-V labels, repeating the experiments of Table 7.6 with labels collected via MTurk.

KL-divergence, the MTurk system performs significantly better. However, high KL values in Table 7.6 were attributed to noisy distribution estimates at one-second intervals, taken independently from other time slices [10]. Increased performance on the MTurk set can be similarly attributed to the larger per-second sample sizes. Improvements based on the *quantity* of data collected are unrelated to the question of whether or not collaborative labeling biases the annotators’ judgments.

8.2 Emotion Prediction with Conditional Random Fields

This section will discuss the application of conditional random fields (CRFs) to the modeling of time-varying musical emotion. As discussed, CRFs are powerful graphical models that are trained to predict the conditional probability $p(\mathbf{y}|\mathbf{x})$ for a sequence of labels \mathbf{y} given a sequence of features \mathbf{x} [110, 111]. Treating acoustic features as deterministic, it is possible to retain the rich local subtleties present in the data, which is especially promising in content-based audio analysis where there is no shortage of rich data. Furthermore, the system provides a model of both the relationships between acoustic data and emotion space parameters, and also how those relationships evolve over time.

In applying CRFs to the problem of predicting emotion in music, instead of modeling the ambiguity of emotion *a-priori* and representing the emotion space distribution parameters as the ground truth, the training algorithm is instead presented with the individual user label sequences, thus allowing the model to learn the range of emotion responses to a given piece. In applying the CRF, it is also necessary to assign emotion space meanings to the states of the model, and in doing so each label in the sequences is discretized to an 11×11 grid. While this is a significant simplification, the

findings shown here indicate that it provides sufficient granularity. Furthermore, the trained models are fully connected and can be used to model complex distributions in emotion as an A-V heatmap. These heatmaps can model arbitrary modes and distributions, in contrast to the previous approach, which constructed uni-modal Gaussian A-V predictions.

8.2.1 A CRF Model for A-V Emotion Prediction

This section discusses the application of a linear-chain CRF to musical emotion prediction. In general, linear-chain CRFs place the assumption that each observation is only dependent upon the previous (i.e., first order Markov), and relate computability between observations and labels using feature functions. In these experiments, all state transitions are learned directly from the labels and are not conditioned on the data.

In mapping acoustic data to states of the model, two types of features are used, which will be referred to node and edge features. The node features map a single binary feature to a single node of the model. Edge features, in the case of these experiments, take into account the state change in the observations from the previous to current frame, but depend only upon the current state of the labels.

CRFs are trained on sequences, and in the process of learning them the system is presented with the individual user ratings (as opposed to statistics of all users) recorded in the MTurk task. Using a fully connected model, it is possible to learn a set of transition probabilities from each class to all others. During inference, CRFs allow the computation of the conditional probability $p(\mathbf{y}|\mathbf{x})$ for every state at each time step. Therefore, at each stage in a testing sequence, it is possible to display the transition probabilities in the form of a heatmap as shown in Figure 8.1.

Acoustic Feature Representation

All features are initially computed using short-time analysis windows at a much higher rate than the 1-second emotion label windows. In order to reduce their frame rate to that of the labels, spectral contrast and MFCCs are simply re-windowed via averaging from their original analysis rate (~ 23 msec). This section will also investigate the ENT features investigated in the feature learning

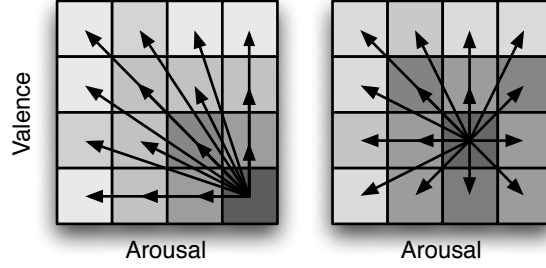


Figure 8.1: Heatmap visualization of CRF transition probabilities. Actual discretization is 11×11 .

experiments in Section 7.7; they are re-windowed following their non-linear analysis frame start times to take into account their beat-synchronous nature.

Additionally, as conditional random fields are highly optimized to operate on binary features, this approach will discuss converting acoustic features into such a representation. In doing so, each feature dimension is quantized using 10 equal energy bins, which for the 14-dimensional case of spectral contrast yields 140 binary features. In early experiments, the use of higher discretization levels was investigated as well as combining representations from multiple discretization levels (e.g., 5, 10, 20), but overall 10 levels were found to offer the best performance.

Training Label Sequence Jittering

In discretizing the original label sequences to the 11×11 grid representation, the CRF models are trained on vectorized version of that space by assigning 121 classes. As a result, the neighborhood relationship of the heatmap grid-cells is lost in the vector-wise representation, and it therefore is necessary to investigate how to improve the model’s ability to learn such relationships.

In order to ensure that the CRF learns the spatial relationships of each class, it is trained on additional “jittered” versions of each label sequence. This has two benefits: it increases the overall size of the dataset, and it helps the model to learn the spatial relationships between the different classes. In applying the jitter, the size of the dataset is increased by a factor of 10, creating 9 additional sequences for each sequence in the dataset. Each jittered sequence is created by adding a small amount of zero mean Gaussian noise, biasing the whole sequence by a single point. In initial

experiments, the number of jittered sequences was investigated at multiple levels between 0 and 50, but 10 was found to offer the best performance.

8.2.2 Applying the Model to MoodSwings Turk

As a baseline for comparing performance of the CRF in modeling the time-dependencies of the data, the performance for the CRF when trained on independent observations as opposed to sequences is also provided. Furthermore, to provide a baseline for comparison to the prior work discussed in Chapter 7, the prediction accuracy of multiple linear regression (MLR) is provided. To compute the heatmap representations for MLR, the mean and covariance of an emotion-space Gaussian density is first predicted using multivariate regression, and then the predicted probability density function is integrated under each square of the heatmap.

As in previous experiments, to avoid the well-known “album-effect,” any songs that were recorded on the same album are either placed entirely in the training or testing set. Additionally, each experiment is subject to 5 cross-validations, varying the distribution of training and testing data sets, which are split 70%/30%, respectively.

CRF Parameterization

The CRF optimization code used for these experiments is built on top of CRF++ [119], a highly efficient general purpose CRF toolkit written in C++. The training of CRFs requires the selection of feature functions. In these experiments, three different types of features are used: a simple unigram node feature for each acoustic feature dimension, a unigram edge feature that models the change in each feature dimension between nodes, and a simple bigram (first order) feature that models the transition between states of the model based upon the label data. The total number of binary CRF features for a selected training set is described in Table 8.2.

Additionally, in the case of the CRF trained on independent observations, only node features are used, so as to avoid an artificial decrease in performance. When presenting the training algorithm with independent examples instead of sequences, feature functions that encode time dependencies that cannot be modeled lead to large decreases in performance.

The training of graphical models such as CRFs tends to have a very high computational cost. These experiments were run on Amazon’s Elastic Compute Cloud (EC2) using High-CPU Extra Large Instances (c1.xlarge), which provide access to a 64-bit platform with 8 virtual cores. Shown in Table 8.2 is the computation time for each feature domain the CRF was trained on, as well as the number of binary features created using the specified feature functions.

Feature	# CRF Features	Compute Time (hrs)
Contrast	210,782	11.49 ± 1.245
MFCC	300,927	11.81 ± 1.515
ENT	185,009	12.04 ± 0.461

Table 8.2: Computing time analysis for CRF training on each cross-validation set.

A-V Mean Prediction Results

The first set of experiments uses CRFs to predict a singular A-V point at each second in the sequences. These MLR experiments vary slightly from those discussed in the previous section since only the past one second of feature data is used to provide an appropriate comparison to the CRF approach. These predictions are taken as the means of the CRF heatmaps, which are compared to the means of the MLR Gaussian distributions. The heatmap mean is computed as the sum of the weighted A-V coordinate values of each bin center. For each two-dimensional heatmap this is computed as,

$$\mu_a = \sum_{y_a, y_v} P(y_a, y_v | \mathbf{x}) y_a, \quad (8.1)$$

$$\mu_v = \sum_{y_a, y_v} P(y_a, y_v | \mathbf{x}) y_v, \quad (8.2)$$

where y_a and y_v are the arousal and valence coordinates of each bin center. The mean values for the ground truth distribution are computed directly in the continuous A-V space. These results are available in the third column of Table 8.3. Overall, the best performance (minimum mean ℓ^2 error) is 0.122 and is obtained using the CRF with MFCCs, a significant improvement over the best result with MLR, which is 0.140 using spectral contrast.

Acoustic Feature	Prediction Method	A-V Mean ℓ^2 Error	Heatmap Earth Mover's Distance	Heatmap Error Unsmoothed ($\times 10^{-2}$)	Heatmap Error Smoothed G.T. ($\times 10^{-2}$)	Heatmap Error Smoothed ($\times 10^{-2}$)
Contrast	CRF	0.130 ± 0.007	0.180 ± 0.007	1.300 ± 0.007	0.539 ± 0.002	0.342 ± 0.0142
MFCC	CRF	0.122 ± 0.004	0.173 ± 0.004	1.300 ± 0.000	0.541 ± 0.010	0.326 ± 0.008
ENT	CRF	0.130 ± 0.004	0.179 ± 0.003	1.300 ± 0.009	0.510 ± 0.010	0.337 ± 0.009
Contrast	CRF-I	0.138 ± 0.006	0.188 ± 0.005	1.323 ± 0.007	0.452 ± 0.012	0.355 ± 0.011
MFCC	CRF-I	0.135 ± 0.004	0.186 ± 0.003	1.319 ± 0.006	0.459 ± 0.007	0.350 ± 0.008
ENT	CRF-I	0.144 ± 0.005	0.194 ± 0.004	1.331 ± 0.005	0.446 ± 0.007	0.367 ± 0.009
Contrast	MLR	0.140 ± 0.005	0.213 ± 0.009	1.082 ± 0.010	0.580 ± 0.018	0.460 ± 0.018
MFCC	MLR	0.141 ± 0.005	0.208 ± 0.008	1.076 ± 0.009	0.570 ± 0.021	0.448 ± 0.021
ENT	MLR	0.153 ± 0.005	0.204 ± 0.007	1.068 ± 0.009	0.560 ± 0.018	0.440 ± 0.018

Table 8.3: Emotion prediction results for conditional random fields (CRF) trained on sequence examples as well as independent examples (CRF-I). Multiple linear regression (MLR) results are provided as a baseline.

Heatmap Prediction Results

As previously stated, with the CRF it is possible to compute the conditional probabilities for each step in a sequence for all states, which can be used to form an A-V heatmap; however, to analyze the prediction accuracy on testing data, ground truth heatmaps must be estimated directly, which is a difficult task. In traditional generative estimation (e.g., maximum likelihood estimation), the goal is to fit a probabilistic model to data and derive a smooth function, even with a small dataset. As heatmap estimation requires histogram techniques, a small amount of data can lead to sparse, blocky estimates, and a massive amount of data is needed to achieve the true smooth distribution.

As a result of this, the earth mover's distance (EMD) [120] has been chosen to be the primary metric for comparing these histograms, which can be thought of as the minimum cost of transforming one heatmap into the other. Using this metric, it is possible to take into account the weight of adjacent bins, which overall provides a more accurate comparison of the two heatmaps. These results are in the fourth column of Table 8.3, where the best performer is the CRF with an EMD of 0.173, which is significantly better than the CRF trained on independent samples at 0.186 and MLR at 0.213.

The absolute pixel error between the predicted and ground truth heatmaps is also investigated. These results are shown in the fifth column of Table 8.3, and it is found that MLR appears to be performing slightly better than the CRF. This result is not surprising given the sparsity that the

ground truth heatmaps exhibit, which is to be expected with 121 histogram bins computed from an average of 16.93 ratings. The MLR method, which predicts Gaussian distributions, guarantees a smooth distribution, producing a lower pixel error if the ground truth is sparse or blocky than the CRF, which takes arbitrary shapes. However, it can be easily demonstrated that the CRF is more accurate by applying a simple smoothing function to the ground truth.

To smooth out the blocking artifacts from sparsity, a simple 2-d Gaussian filter is applied. This process applies a light smoothing without altering the mean of the data. These results are shown in the sixth column of Table 8.3. Here the CRF performs slightly better, and the performance similarity is most likely because the CRF is producing rough edges compared to the smooth MLR predictions that are computed from the Gaussian PDF. An interesting result is that the independently learned CRFs perform the best here. This is most likely because they produce more uniform transition probabilities due to their training method.

To compensate for blocking artifacts in the CRF predictions, a smoothing filter is applied to the predicted heatmaps as well. Initial experiments showed that applying the same filter to the MLR heatmaps improved performance there also, so to keep the analysis consistent the filter is applied to them as well. The differences in heatmaps are examined using mean absolute error, and these results are shown in the seventh column of Table 8.3. In these results, it is seen again that the CRF is performing significantly better than MLR.

Visualizing the Results

Shown in Figure 8.2 are the CRF heatmap predictions for eight seconds of the song “Something About You,” by Boston. The colormap of these heatmaps assigns red to areas of high density, blue to low, and uses the color spectrum to assign colors in between. This clip was selected because of the large change in emotion that occurs at second 29, where the song transitions from a low-energy, negative-emotion introduction into a high-energy, positive-emotion hard-rock verse. The system tracks the transition very accurately, showing a brief amount of uncertainty at second 30 in terms of positive or negative emotion, and finally settling on positive emotion at second 31.

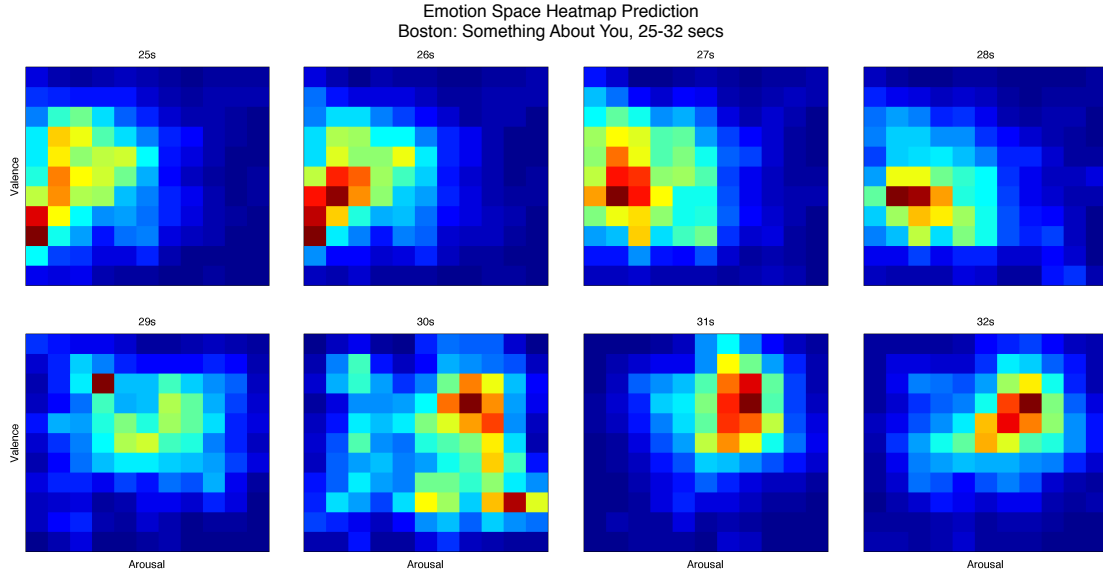


Figure 8.2: Emotion space heatmap prediction using conditional random fields. Shown is the predicted emotion from the beginning of the song “Something About You,” by Boston. These figures demonstrate the system tracking the emotion through the low-energy, negative-emotion introduction, and through the transition at second 29 into a high-energy, positive emotion rock verse. In these figures, red indicates the highest density and blue is the lowest.

8.3 Feature Learning Experiments

The following experiments investigate the use of deep belief networks (DBNs) for emotion-based acoustic feature learning [13, 14]. In all experiments, the model training is cross-validated 5 times, dividing the dataset into 50% training, 20% verification, and 30% testing. All learned features are then evaluated in the context of multiple linear regression (MLR), as has been investigated in prior experiments. MLR provides extremely high computational efficiency, making it ideal for discriminating between relative usefulness of many feature domains.

8.3.1 Short-Time Feature Learning

The first set of experiments will investigate learning features directly from short-time magnitude spectra. This approach was investigated in the previous chapter on the MoodSwings Lite set (Section 7.7), and here it is investigated to compare performance with the Turk dataset and to provide a baseline for further work. As with the previous approach, 3 hidden layers are used in all experiments, each containing 50 nodes. Furthermore, pre-training is run for 50 epochs with a learning rate of

0.001. During the conjugate gradient fine-tuning stage, an additional multiple linear regression (MLR) layer is attached to the output of the DBN. As this stage is supervised, for each input example \mathbf{x}_i , the model is trained to produce the emotion space parameter vector \mathbf{y}_i ,

$$\mathbf{y}_i = [\mu_a, \mu_v]. \quad (8.3)$$

Results for standard acoustic features (e.g., MFCC) are again presented, as these experiments differ slightly from the previous ones. All features are aggregated at one-second intervals just as in the CRF experiments, but here, since the focus is A-V Gaussian prediction, the Kalman approach for estimating ground truth parameters is used, which was developed in Section 7.6.

Shown in Table 8.4 are the results for employing the learned features for multiple linear regression. Features are first extracted on 20msec intervals and then appropriately aggregated to match the one second intervals of the labels. Results for these features that are learned from single frames are shown as DBN-SF. Additionally, the KL-divergence is shown for the Gaussian ground-truth representation used in prior work approaches. In that approach, regressors are developed to predict the parameterization vector \mathbf{y}_i of a two-dimensional Gaussian in A-V space,

$$\mathbf{y}_i = [\mu_a, \mu_v, \sigma_{aa}^2, \sigma_{vv}^2, \sigma_{av}^2]. \quad (8.4)$$

8.3.2 Multi-Frame Feature Learning

While future work on more sophisticated finetuning approaches or better stochastic models in pre-training may improve performance, the largest issue is the inherent limitation in using a single short-time window. Human emotional associations necessarily require more than a ~ 20 msec short-time window, and thus future approaches must take into account the variation of acoustic data over a larger period of time. In these experiments, the development of models that incorporate multiple spectral windows is investigated to derive musical emotion. Taking spectral aggregations of the past one second, past two seconds, and past four seconds, the resulting vectors are concatenated as inputs

Feature Type	A-V Mean ℓ^2 Error	Average KL Divergence
MFCC	0.140 ± 0.005	1.28 ± 0.157
Chroma	0.182 ± 0.006	3.33 ± 0.294
Spectral Shape	0.153 ± 0.006	1.51 ± 0.160
Spectral Contrast	0.138 ± 0.005	1.29 ± 0.160
ENT	0.151 ± 0.006	1.41 ± 0.175
DBN-SF Model Error	0.203 ± 0.009	-
DBN-SF Layer 1	0.138 ± 0.005	1.25 ± 0.142
DBN-SF Layer 2	0.133 ± 0.004	1.19 ± 0.129
DBN-SF Layer 3	0.133 ± 0.002	1.21 ± 0.180
DBN-MF Model Error	0.194 ± 0.032	-
DBN-MF Layer 1	0.131 ± 0.006	1.15 ± 0.106
DBN-MF Layer 2	0.131 ± 0.004	1.14 ± 0.107
DBN-MF Layer 3	0.129 ± 0.004	1.12 ± 0.114
DBN-UBM Model Error	0.140 ± 0.015	-
DBN-UBM Layer 1	0.129 ± 0.006	1.12 ± 0.091
DBN-UBM Layer 2	0.128 ± 0.004	1.13 ± 0.097
DBN-UBM Layer 3	0.128 ± 0.004	1.11 ± 0.090

Table 8.4: Emotion regression results for fifteen second clips. DBN-SF are features learned from single frames (SF), DBN-MF are features learned from multi-frame (MF) aggregations, and DBN-UBM are features learned with a universal background model (UBM) approach to DBN pretraining. KL-divergence is not applicable to model error.

to the system. As each spectrum frame is a 257-dimensional vector, the total DBN input is now 771 dimensions. Results for multi-frame (MF) feature learning can be found in Table 8.4 labeled as DBN-MF.

For this new approach, visualizations of the learned features are provided. Figure 8.3 shows the input spectrogram in log-magnitude for proper visualization, though log is not taken for the actual model input.

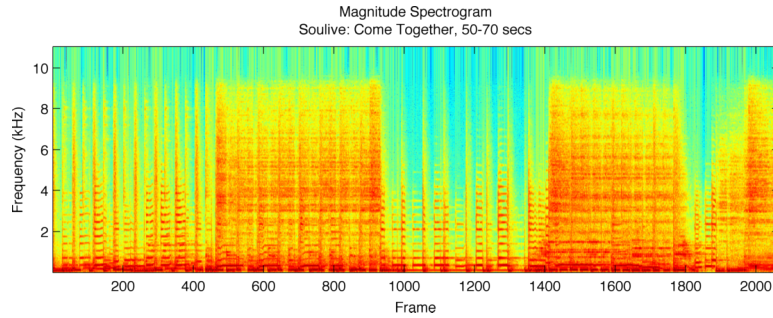


Figure 8.3: Log-magnitude spectrogram of input audio.

In the original spectrogram (Figure 8.3), the verse transition into the first chorus is seen for the Soulive rendition of the Beatles song *Come Together*, starting at around frame 500. A similar pattern is seen in the spectrogram between frames 1488-1800, which is the only other part of the clip where the percussion includes cymbals. Shown in Figure 8.4 are the resulting features from the intermediary layer outputs. Note that the structural information in the spectrogram is retained in the hidden layer outputs rendered in Figure 8.4.

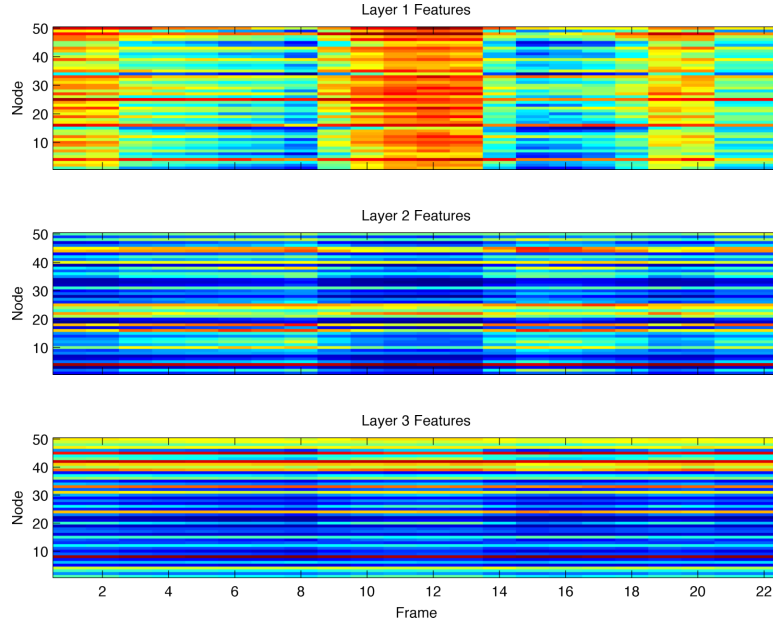


Figure 8.4: DBN hidden layer outputs using the aggregate spectral frames as input.

Also investigated is the reconstruction of the original input aggregated spectrogram from the hidden layer outputs. Figure 8.5 depicts this reconstruction, which was generated using the method outlined in Section 7.7. Due to the concatenations of multiple time scale aggregations, the y-axis is adjusted to display the correct frequency values for each. The top contains the last one second aggregations, below that is the aggregation from the last two seconds, and the last four seconds are at the bottom.

8.3.3 Universal Background Model Feature Learning

In order to improve the results with the multi-frame approach, it is desired to harness the power of a much larger unlabeled music dataset. As DBN training relies on a two step training process, the

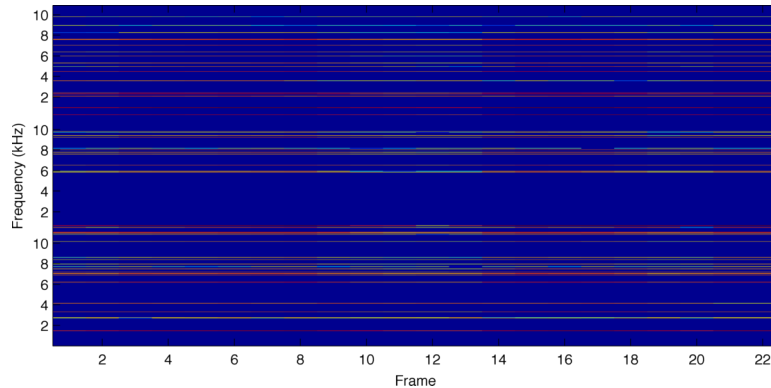


Figure 8.5: Reconstruction of the original aggregated spectrogram used as the DBN input. Top is last one second aggregates, middle is last 2 seconds, bottom is last 4 seconds.

first of which is unsupervised, there is no reason not to use every piece of available data. In training the RBMs with the larger dataset, a much more accurate portrayal of the overall distribution of music is provided, and therefore creates a much more accurate music model, which can then be finetuned for musical emotion or any other supervised machine learning problem. As this model is a general music model, it will be referred to as a universal background model (UBM). For the larger dataset, the uspop2002 dataset [84] is used in its entirety, which contains nearly 8000 songs. Even after aggregating spectra at one-second intervals, this adds up to ~ 26 GB of training data. Results for the universal background model approach are shown in Table 8.4 as DBN-UBM.

8.4 Conditional Random Fields with Learned Features

This set of experiments will investigate applying the feature representations learned in the previous section to CRFs for time-varying emotion prediction. The results here can be compared directly to those of the previous CRF experiments in Section 8.2, Table 8.3. Overall, the CRF experimental setup is identical to Section 8.2.

8.4.1 DBN Single-Frame Features

The first set of features investigated are the single-frame features (DBN-SF) from the prior section (Section 8.3). Results for both average mean distance (single A-V point prediction) as well as earth movers distance are shown in Table 8.5. Overall the CRF performs very well on DBN features.

Feature Type	A-V Mean ℓ^2 Error	Earth Mover's Distance
DBN-SF Layer 1	0.135 ± 0.004	0.182 ± 0.004
DBN-SF Layer 2	0.127 ± 0.007	0.176 ± 0.007
DBN-SF Layer 3	0.129 ± 0.004	0.178 ± 0.004
DBN-UBM Layer 1	0.140 ± 0.007	0.187 ± 0.005
DBN-UBM Layer 2	0.133 ± 0.006	0.181 ± 0.004
DBN-UBM Layer 3	0.135 ± 0.008	0.184 ± 0.006

Table 8.5: Time-varying music emotion prediction using conditional random fields. DBN-SF are features learned from single frames (SF) and DBN-UBM are features learned with a universal background model (UBM) approach to DBN pretraining.

Comparing to Table 8.3, the second layer of the DBN performs better than both spectral contrast and ENT features, where both of those features achieved an ℓ^2 error of 0.130. MFCC features perform only slightly better than the learned representations, at an ℓ^2 error of 0.122.

8.4.2 DBN Multi-Frame Features

The next set of experiments uses the UBM multi-frame feature learning approach (DBN-UBM) discussed in the previous section (Section 8.3). The UBM approach is used for pretraining, as it was shown to improve overall mode error. Looking at the CRF results for these features, it can be seen that the error has increased slightly. The best performing feature here is also the second layer, with an A-V ℓ^2 error of 0.133. This result differs from the previous section, where with the MLR model, the DBN-UBM features improved performance over single frames and improved the A-V prediction error of the DBN model itself significantly.

This performance could be attributed to a variety of factors, with the most obvious being the reduced dynamics of the individual feature dimensions over time. Observing Figure 8.4, there is very little change in the features over time. This is a large change from the single frame method, where the individual dimensions were quite dynamic (see Figure 7.11). With such little dynamics, CRF features that consider edges (changes in the acoustic domain) may not be able to extract relevant information from the signal and may lead to curse of dimensionality.

8.5 Discussion

The MoodSwings Turk database is a reasonably well-annotated corpus that should hopefully serve to advance the field of music emotion recognition. These experiments have further demonstrated its correlation to the original MoodSwings database, and the corpus has been used here to evaluate a variety of algorithms.

Overall, conditional random fields have been shown to be a powerful tool for modeling time-varying musical emotion. The CRF approach is shown to be superior to MLR, both at predicting single A-V mean values as well as full emotion space heatmaps. In looking at standard features (Section 8.2), the best performing feature for CRF prediction is MFCCs, which differs from the MLR method where spectral contrast performs best. This perhaps indicates that there is more information to be gained out of MFCCs when modeling the temporal evolution of emotion.

Using the earth mover’s distance, it is possible to better analyze the similarity between heatmaps by also taking into account adjacent bin densities. While the MLR method appears to perform slightly higher when the ground truth distributions are not smoothed, this is a result of blocking artifacts in the ground truth. The Gaussian density is a smooth function, which is much more likely to be similar to a sparse ground truth distribution than the CRF predictions, which take on arbitrary shapes and are not necessarily as smooth. Overall, the ground truth representation could significantly benefit from more data.

Acoustic feature learning also is very promising for this application. In looking at the MLR experiments (Section 8.3), the first set of results for learning features from single frames (DBN-SF) agrees with what was found in the previous chapter (Section 7.7). The second layer features perform best for this method, outperforming spectral contrast, which is the best performing standard feature. However, there is a slight variation in that here the DBN-SF features are found to be better than spectral contrast, both in predicting single points and distributions. In the previous chapter, the DBN features were found to be more accurate in terms of mean prediction, but spectral contrast was shown to perform slightly better in terms of KL, though it was strongly emphasized that an incorrect mean is much worse an error than an incorrectly sized or rotated covariance.

In trying to improve the features by including multiple timescales, an improvement is seen in A-V ℓ^2 error from 0.133 to 0.129, which is encouraging. In analyzing the reconstructed spectra from first layer features, an interesting result is found similar to the previous chapter. The overall representation is very sparse in terms of frequency and seems to target very specific frequencies to contribute to the overall emotion features. Analyzing the features in Figure 8.4, it is noted that there is most definitely an emotion change as we progress from the slower and heavily minor sounding verse into the higher tempo rock chorus. Changes are reflected in all three layers' features in that area of the clip. It is also noted that it appears as if the features do not exactly line up with the spectrogram, which is a result of including past data in our feature computation. When the spectrum changes abruptly it takes several frames for our model to catch up. This is not seen as a limitation, as humans have a reaction time too, which perhaps is reflected in the increased MLR performance on these features.

Furthermore, while the performance increase between the third layer features of DBN-MF and DBN-UMB is small, the performance of the DBN model itself is reduced from 0.194 to 0.140, which we find to be highly encouraging. These results indicate that UBM pretraining is providing us a model that is much better suited (regularized) for emotion finetuning.

In applying the DBN features to CRFs, the single-frame approach is very promising. The second layer features perform better than the other standard features investigated, with the exception of MFCCs. However, the fact that multi-frame UBM features show decreased performance is an interesting result. This method increased the overall accuracy of the DBN model itself at predicting A-V emotion, as well as MLR methods that employed features learned using these techniques. As previously stated, this is most likely attributed to decreased dynamics in the feature space due to aggregating past data.

In future work, shrinking layer sizes should be investigated in the UBM approach, where the dimensionality reduction power of the RBM can be better taken advantage of. Furthermore, it may also be interesting to investigate multiple stages of finetuning. Such an approach would first follow that of [93] for reducing the dimensionality of unlabeled data. It may be possible to gain a more

accurate UBM by applying a finetuning stage that involved unraveling the model to reconstruct the unlabeled data. Those model parameters could then be adapted to emotion or any other type of music prediction.

Chapter 9: Instrumental Dataset Experiments

This chapter will investigate the robustness of the algorithms developed in this work for musical emotion prediction using instrumental data. Instrumental data presents an interesting problem for several reasons, first and foremost because it removes the influence of lyrics. This work seeks to extract emotion purely through analysis of the acoustic content, and therefore these methods have no access to text data containing the lyrics. Removing lyrics can ensure that the collected data is truly a reflection of the acoustic content, and not a reflection of a potentially conflicting message in lyrics. Furthermore, removing the influence of lyrics also potentially opens the door for much wider variations in opinion, creating interesting A-V emotion-space distributions. Finally, this data has been hand chosen to contain large shifts in emotion-space dynamics, as opposed to the MoodSwings audio corpus, which contained temporal information but had no guarantee of dynamics. This dataset should therefore better serve to evaluate algorithms that seek to model such dynamics.

The first experiments presented in this chapter apply the method of time-varying A-V Gaussian prediction originally presented using MoodSwings Lite (Section 7.5), and then later evaluated using MoodSwings Turk (Section 8.1). For initial experiments, it was desired to see how well this simplistic method functions on this new and complex dataset. The varying opinions and high A-V dynamics of these songs, as opposed to the previous MoodSwings data, present an interesting opportunity to evaluate the performance of these methods. Finally, conditional random fields are investigated, demonstrating that the graphical modeling techniques developed in the previous chapters also function well in this domain.

9.1 Predicting A-V Gaussian Distributions

This section will investigate multiple linear regression over time using the instrumental dataset. These experiments seek to parameterize A-V Gaussian distributions automatically at one-second intervals. In training these regressors, the collected label distributions are extracted using the

Feature/ Topology	Average Mean Distance	Average KL Divergence	Average Randomized KL Divergence	T-test
MFCC	0.152 ± 0.005	1.206 ± 0.100	1.650 ± 0.111	42.15
Chroma	0.178 ± 0.005	3.805 ± 0.168	4.011 ± 0.181	42.44
S. Shape	0.156 ± 0.005	1.269 ± 0.092	1.631 ± 0.112	43.61
S. Contrast	0.152 ± 0.004	1.159 ± 0.092	1.629 ± 0.119	42.56
M.L. Combined	0.147 ± 0.005	1.092 ± 0.092	1.687 ± 0.114	41.27

Table 9.1: MLR results for short-time (one-second) A-V labels, repeating the experiments of Table 7.6 with labels collected via MTurk.

Kalman approach originally discussed in Section 7.6. While it is not expected that the performance should be identical on these experiments, it is expected that the relative performance between features will be similar.

Just as in the prior experiments, the systems utilize supervised machine learning algorithms to map A-V labels to content-based audio features (e.g., MFCCs). Prediction performance for each feature, as well as combined performance using a multi-layer regression method for late-feature fusion, is shown in Table 9.1.

Shown in Figure 9.1 are the predictions for a few selected audio clips using this method. Shown in grey are the individual user ratings, red ellipses are the estimated distribution of the collected data, and blue ellipses are the predicted distributions, all of which get darker over time. In general, the dataset contains complicated targets in terms of prediction, as many of them exhibit emotion-spaces more than once throughout the entire clip. Furthermore, there are generally varying opinions about the emotion, all of which are valid. For example, in the clip shown in the top right, most annotators agree that it moves from low to high arousal (energy), but some annotators feel a strong movement towards high-valence, high-arousal (excited), while others feel that it moves towards low-valence, high-arousal (angry). In listening to the clip, at around 40 seconds the drums and bass enter with a steady, and intense groove and could easily be argued to convey high-valence and high-arousal emotions. However, at the same time the horn section enters with a very dissonant melody that could easily be considered to convey low-valence. Both of these opinions are valid, and overall it could be argued that perhaps this clip exhibits both emotions simultaneously.

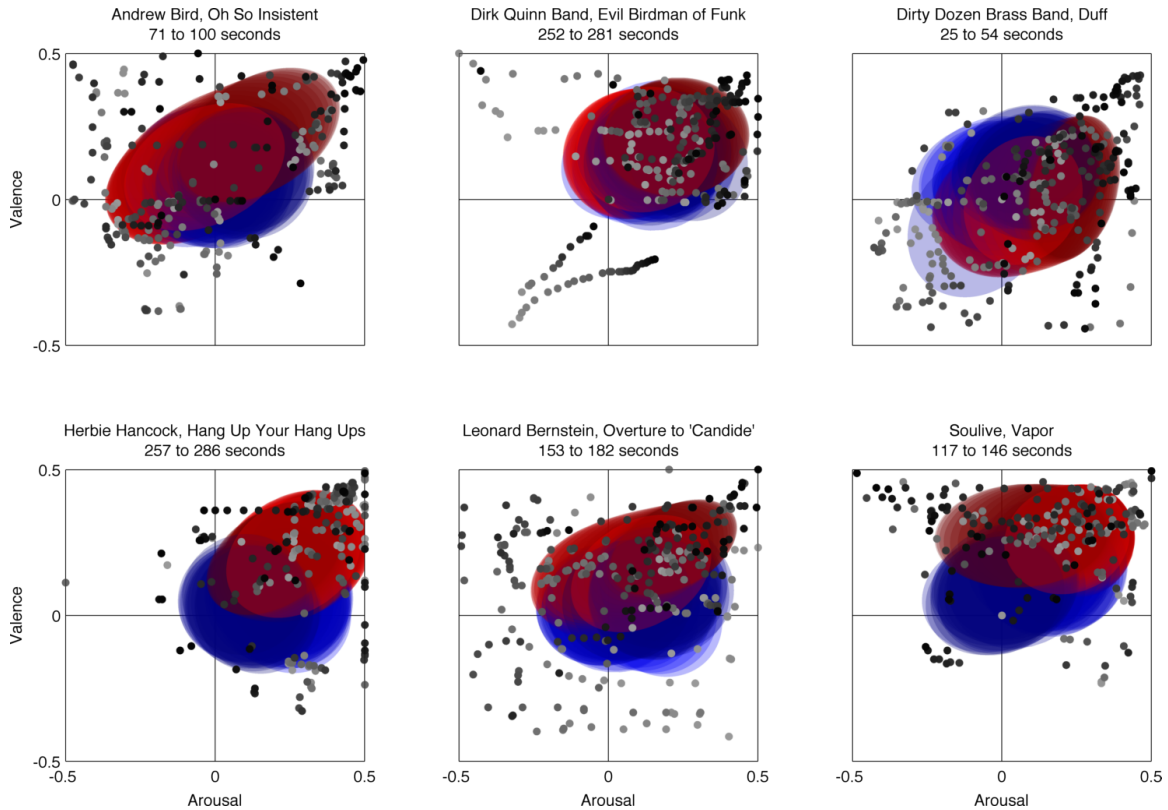


Figure 9.1: Time-varying A-V emotion distribution predictions of instrumental data using the M.L. combined method (see Section 7.5). Shown in gray are the individual user ratings, red ellipses are the estimated distribution of the collected data, and blue ellipses are the predicted distributions, all of which get darker over time.

9.2 Conditional Random Fields

Given clips that contain valid disagreement in emotion, the single bivariate Gaussian representation used in the MLR experiments is seen as a modeling limitation. Therefore, in these next experiments, the conditional random fields approach is investigated, as it is capable of predicting A-V heatmaps that can represent arbitrary A-V distributions. This approach was originally developed in Chapter 8 and is now applied to the problem of predicting complex emotion-space distributions of instrumental data.

The results for this approach are shown in Table 9.2, and as in the previous chapter, the results of CRFs trained without edge features are provided as well. That approach contains no model of the temporal evolution of the sequences and is provided to benchmark how much the approach benefits

Acoustic Feature	Prediction Method	A-V Mean ℓ^2 Error	Heatmap Earth Mover's Distance	Heatmap Error Unsmoothed ($\times 10^{-2}$)	Heatmap Error Smoothed G.T. ($\times 10^{-2}$)	Heatmap Error Smoothed ($\times 10^{-2}$)
Contrast	CRF	0.153 ± 0.004	0.202 ± 0.004	1.356 ± 0.002	0.563 ± 0.009	0.379 ± 0.010
MFCC	CRF	0.154 ± 0.005	0.201 ± 0.004	1.358 ± 0.006	0.582 ± 0.005	0.379 ± 0.011
ENT	CRF	0.158 ± 0.002	0.205 ± 0.003	1.363 ± 0.005	0.545 ± 0.012	0.386 ± 0.008
Contrast	CRF-I	0.155 ± 0.004	0.203 ± 0.004	1.372 ± 0.003	0.457 ± 0.006	0.379 ± 0.011
MFCC	CRF-I	0.157 ± 0.002	0.205 ± 0.002	1.373 ± 0.001	0.468 ± 0.005	0.383 ± 0.009
ENT	CRF-I	0.161 ± 0.003	0.208 ± 0.003	1.378 ± 0.003	0.458 ± 0.003	0.391 ± 0.008
Contrast	MLR	0.162 ± 0.004	0.200 ± 0.004	1.360 ± 0.004	0.485 ± 0.010	0.399 ± 0.010
MFCC	MLR	0.163 ± 0.006	0.202 ± 0.004	1.361 ± 0.004	0.507 ± 0.010	0.408 ± 0.012
ENT	MLR	0.165 ± 0.000	0.204 ± 0.003	1.362 ± 0.004	0.505 ± 0.012	0.410 ± 0.010

Table 9.2: Emotion prediction results for conditional random fields (CRF) trained on sequence examples as well as independent examples (CRF-I). Multiple linear regression (MLR) results are provided as a baseline.

from a temporal model. Furthermore, MLR results are again displayed, and the values for A-V ℓ^2 error vary slightly from the previous experiments; this is because the collected data distributions are not smoothed using the Kalman approach applied in the last section, which is removed to provide an accurate comparison to the CRF methods. Furthermore, just as in the previous chapter, MLR heatmaps are generated by integrating the probability distribution of the predicted Gaussian under each heatmap bin.

Overall, spectral contrast performs best in terms of the A-V mean predictions using the CRF method. However, there is slight disagreement between the varying metrics used to determine the performance of the heatmaps. In the end, the MLR method actually performs slightly better than CRFs in terms of earth mover's distance, with error values of 0.200 and 0.202, respectively. However, in terms of the smoothed heatmap differences, the CRF method performs best with an error of 0.379 versus 0.399.

9.3 Discussion

The MLR error seen in these experiments is slightly higher than that found on the previous datasets, but overall the features do rank in the same order. Spectral contrast is the best performing single feature, and the multi-level combined approach improves performance better than the individual feature regressors. In general, these clips seem to be varying in emotion space distribution over

time, not just in terms of parameterization, but perhaps in type of distribution family. Such a result makes the case for heatmap representations.

Application of CRFs on this dataset provided results that varied slightly from what was seen on the MoodSwings Turk dataset (see Chapter 8). It is difficult to clearly rank the performance of these methods, as there is disagreement between the various metrics. Overall, CRFs display the best performance using the A-V ℓ^2 error metric as well as the smoothed and unsmoothed heatmap differences, but the MLR method performs best using the earth mover’s distance. These results point towards the CRF displaying superior performance, but future work must further investigate the relative significance of these various metrics.

Chapter 10: Conclusions and Future Directions

A variety of methods have been presented in this work for the representation, modeling, and prediction of musical emotion. The overall focus of the work should be seen as understanding human arousal-valence (A-V) responses to music through content-based (signal processing) techniques. The main contributions of this work are as follows:

- Collected multiple datasets spanning multiple genres of music, including a pop music database collected via an online collaborative gaming approach [6], a pop music database crowdsourced through Amazon’s Mechanical Turk (MTurk) [7, 8], and a database that consists entirely of instrumental music.
- Proposed a range of methods for representing arousal-valence disagreement in human responses to music, including aggregating the data from multiple subjects to a single A-V point [6], representing disagreement in collected data as an A-V distribution [9, 10], and using emotion space heatmaps that can represent arbitrary distributions and modes [11].
- Developed multiple frameworks for acoustic feature identification, including those based on perceptual evaluations of existing domains for a bottom-up style feature selection process [12], as well as methods to learn sets of basis functions directly from magnitude spectra that are informative to musical emotion [7, 13, 14].
- Developed multiple machine learning approaches for relating high-dimensional acoustic content to low-dimensional A-V representations, including multi-class classification [6], direct A-V regression using a variety of methods [6], regression methods for the automatic parameterization of A-V Gaussian distributions [9, 10], and a final method using conditional random fields (CRFs) that is capable of automatically parameterizing A-V heatmaps over time [11].

10.1 Overall Conclusions

This work investigated a variety of methods for human data collection, including an online collaborative gaming approach using MoodSwings, and an online crowdsourcing system approach, using Amazon’s Mechanical Turk, which offered small sums of payment for data labeling. Overall, both approaches show similar results, but each have their own set of tradeoffs. With the gaming approach, the largest challenge is making the experience fun so users will return and continue to contribute data. Quality control is achieved by assigning scores based on A-V agreement between two players, and bonus points are awarded to the player who moves their cursor to those A-V coordinates first, making it impossible to win by simply chasing the other player. Overall it was found to be quite difficult to raise interest in the game outside of the research community, and even more difficult to attract repeat players. Conversely, the MTurk approach tended to attract large numbers of participants and massive data collections, but quality control was very challenging and time consuming. Many Turkers attempted to game the system by either leaving their cursor at the origin or make shapes such as squares or figure eights in the A-V space. These were easy to identify, but there were also Turkers who failed to read the directions and did not understand the A-V space. Given the subjectivity of the task, those Turkers were much harder to identify. Furthermore, MTurk offers the ability to reject bad or malicious work from Turkers, but rejecting such data can lead to negative reviews of the task and often requires correspondence with the individual, which can be very time consuming. Turkers must maintain a certain acceptance rate in order to participate in MTurk and it was found after running several tasks that reviews claiming unfair rejection can scare off potential workers, essentially making it difficult to attract good workers because of rejecting bad ones.

The perceptual studies presented in Chapter 4 offered a theoretical backing for the use of standard acoustic representations (i.e., MFCC, chroma) that are used in both speech processing and music information retrieval. While these studies demonstrated that such features most likely do contain a significant amount of information about musical emotion, users were not able to exactly label reconstructions from features as conveying the same emotion as the originals. The roughly 10% error shown (see Table 4.1) between these ratings could perhaps be seen as a theoretical upper limit.

For instance, the best system for predicting emotion, conditional random fields with MFCC features, generally makes predictions with about 12% error (see Table 8.3). However, a very interesting result is that while chroma perhaps fared the best in the perceptual studies, it tended to be the one of the poorest performing features on nearly all of the emotion prediction experiments, which is most likely attributed to the loss of information in aggregated representations.

In looking to take things a step further, feature learning offers a promising direction, with most approaches demonstrating learned feature representations that obtain higher accuracy on the emotion prediction tasks than any other feature. These approaches were originally presented in Chapter 7 on the MoodSwings Lite dataset, and then later refined using the MoodSwings Turk dataset in Chapter 8. The universal background model (UBM) method developed in the later approaches (see Chapter 8) offered additional regularization in terms of starting conditions, however, the finetuning of the model remains a difficult obstacle. The application of iterative numerical optimization methods, such as gradient descent on functions that lack a guarantee of convexity, is very unlikely to find a global minima. These approaches undoubtedly lead to interesting and promising results, but will require further regularization in order to take the next step.

Overall, conditional random fields offered some of the most promising results in modeling the temporal evolution of musical emotion. Using emotion space heatmaps, the models are able to represent arbitrary A-V distributions; this is especially promising in the case of instrumental music (e.g., jazz, classical), where there may be large disparity in human opinions at each moment in a song.

10.2 Future Directions

In looking towards future directions in the data collection methods, there are a variety of tradeoffs that need to be taken into consideration. In continuing with Mechanical Turk, it would be helpful to develop a series of tests that Turkers must pass in order to be allowed to participate in the task. Those tests would seek to verify a basic understanding of the space before workers are allowed to label data, and would hopefully reduce the overall number of submissions that need to be rejected, while still ensuring high quality ratings. Future approaches cannot reject large numbers of submissions,

as this hurts the ability of the task to attract participants. Given the difficulty and time-consuming nature of data filtering and correspondence with participants on MTurk, the advantages over bringing subjects into the lab is perhaps limited, especially at institutions that have access to large numbers of undergraduates who can receive course credit for participation in such studies.

In seeking to improve feature learning methods, future approaches should investigate regularizing the DBN finetuning. The experiments discussed in Chapter 5 sought to relate invariances (or variances) in musical parameters to those in emotional content. Perhaps future feature learning approaches could use such knowledge to regularize the optimization process in feature learning, though requiring key invariant features, for instance, would be difficult and necessarily require that all examples be annotated with such data. Future work should investigate this topic in greater depth and continue to identify high-level acoustic transformations that are invariant to emotion.

In analyzing the multiple machine learning algorithms utilized in this work, a variety of metrics were investigated: Euclidean distances between A-V means, relative entropy between A-V Gaussian distributions, absolute differences between A-V heatmaps, absolute differences between smoothed A-V heatmaps, and the earth mover’s distance between A-V heatmaps. In moving forward with these approaches, more investigation is necessary into the optimal metric for analyzing performance, especially in the context of instrumental data, as comparing these metrics in some cases did not point to a clearly superior method. One interesting way to achieve this would be through a perceptual study, using a labeled corpus. For instance, to compare the relative information contained in two metrics, a user could be presented with a song that is randomly chosen from the dataset, as well as two other songs, each of which is selected by finding the most similarly labeled song using one of the two metrics, respectively. The more informative metric would be chosen to be the one that more often locates the song with a more similar emotion.

With the CRF approach, it may also be interesting to investigate alternate feature functions to operate on continuous valued data, as well as additional regularization schemes, to ensure the model represents the neighborhood relationship of the heatmap bins. It would be possible to add an additional regularization parameter that ensures adjacent bins are assigned similar weights. Doing

so would remove the necessity for including “jittered” duplicates, and more importantly would ensure that the model produces smooth heatmap estimates. In these experiments it has been shown that CRF accuracy is improved by applying a smoothing filter to the predicted heatmaps, and an approach containing these additional regularization parameters should remove this necessity and perhaps lead to more accurate predictions overall.

Appendix A: Acoustic Feature Extraction

This chapter investigates the computation of acoustic features for music emotion recognition. Previous work has indicated that there is no single dominant feature in determining musical emotion, but rather many that play a role (e.g., loudness, timbre, harmony). Since the experiments discussed in Chapters 7 - 9 focus on the tracking of emotion over time, this chapter will focus solely on time-varying features. This collection contains many features that are popular in music information retrieval (Music-IR) and speech processing, encompassing both psychoacoustic as well as music-theoretic representations.

A.1 Mel-Frequency Cepstral Coefficients

Mel-frequency cepstral coefficients (MFCCs) are among the most widely used acoustic features in speech and audio processing. MFCCs are essentially a low-dimensional representation of the frequency spectrum warped according to the mel-scale, which reflects the nonlinear frequency sensitivity of the human auditory system. MFCCs can be defined as [62, 63]:

$$M_{m,c} = \sum_{b=1}^{40} \hat{X}_{m,b} \cos \left[c \left(b - \frac{1}{2} \right) \frac{\pi}{40} \right], c = 1, 2, \dots, 20 \quad (\text{A.1})$$

$$\hat{X}_{m,b} = m_b[k] \log \left| \sum_{n=0}^{N-1} x_m[n] e^{-j \frac{2\pi n k}{N}} \right|. \quad (\text{A.2})$$

Normally, MFCCs are implemented over short-time segments, and this implementation accordingly divides the audio into overlapping segments and applies a Hanning window function to reduce edge effects. The discrete Fourier transform (DFT) of each short-time segment is computed using the fast Fourier transform (FFT) algorithm. The magnitude of the frequency components is determined and the mel-spaced triangular filters (the design of the triangular filters is discussed below in greater detail) are applied via multiplication in the frequency domain. Continuing with the cepstrum calcu-

lation, the log of the mel-filtered energies is calculated, which serves, to some extent, to deconvolve the audio by transforming multiplications in the frequency-domain (and thus, convolutions in the time-domain) into additions. As a final step, the inverse DFT is applied to the log of the mel-filtered data using the DCT (sine components are not needed since the input is guaranteed to be real and even). This step is also used to reduce the dimensionality of the data to the desired quantity, and it has been shown that the DCT has the additional effect of decorrelating the vector of feature components [63].

A.1.1 Mel-Spaced Triangular Filters

The filterbank range is determined by defining minimum and maximum frequencies in Hertz and converting those to the mel-scale. Within the defined mel-range, the desired number of filters are linearly spaced (according to mels) while ensuring 50% overlap with adjacent filters. Each filter's low-edge, center, and high-edge frequencies (f_l , f_c , and f_h , respectively) is then converted to Hertz.

As each mel-filter is applied by means of a multiplication in the frequency domain, it is necessary for the filters to be represented at the same granularity as the DFT of the audio signal. The frequency samples for each of the triangular filters are generated from the ideal filter parameters, linearly interpolating between the center and edge frequency values. Lastly, the amplitude for each filter is normalized such that the total energy of each filter is equal to one. Shown in Figure A.1 is a typical filterbank containing 40 mel-spaced filters employed in an MFCC calculation.

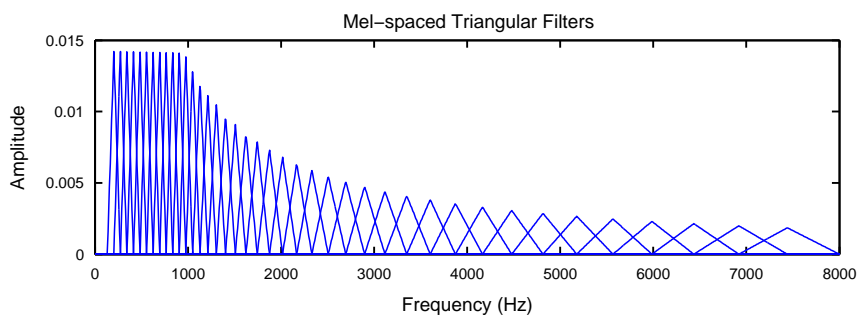


Figure A.1: Forty band mel-warped triangular filterbank

A.2 Autocorrelation of chroma

The chroma feature is an often-used representation to estimate spectral energies of the different musical pitch classes (A, A#, etc.) within a short time-interval. It is essentially a circular version of the logarithmically warped spectrogram, where the frequencies corresponding to pitches in different octaves are added together to provide a summary of energy at each of the 12 pitch classes. Using this feature, it is sometimes possible to obtain an indication of the overall musical key and modality [121]. In these experiments, however, the autocorrelation of each short-time chroma vector is utilized, providing a shift-invariant feature. In preliminary experiments, it was found that this feature performs better than raw chroma, since it promotes similarity in terms of the modes of harmony (e.g. major, minor, augmented, and diminished chords) as opposed to particular chords (e.g., A major vs. D major).

A.3 Spectral Shape Features

In music and audio processing, spectral shape features are often related to timbral texture [122]. For each spectral shape function, we begin by dividing the data into short-overlapping segments, applying a Hanning window, and computing the magnitude DFT.

A.3.1 Spectral Centroid

Spectral centroid is defined as the weighted-average (center of mass) of the spectrum,

$$C_m = \frac{\sum_{k=0}^{K-1} F[k] |X_m[k]|}{\sum_{k=0}^{K-1} |X_m[k]|}, \quad (\text{A.3})$$

where $X_m[k]$ is the DFT of short-time segment m , and $F[k]$ is a vector of frequencies corresponding to the bins of the magnitude spectrum.

A.3.2 Spectral Flux

Spectral flux is defined as the Euclidean distance between successive spectral frames:

$$F_m = \left(\sum_{k=0}^{K-1} (|X_m[k]| - |X_{m-1}[k]|)^2 \right)^{\frac{1}{2}}. \quad (\text{A.4})$$

Again, $X_m[k]$ is the discrete spectrum of the current analysis frame m and $X_{m-1}[k]$ is the spectrum of the previous frame.

A.3.3 Spectral Rolloff

Spectral rolloff is defined as the frequency beneath which a given proportion of the total spectral energy lies, typically 85%:

$$R_m = \frac{f_s}{K} r_m = \frac{f_s}{K} \left(\arg \sum_{k=0}^{r_m} |X_m[k]| = 0.85 \sum_{k=0}^{K-1} |X_m[k]| \right). \quad (\text{A.5})$$

Here $|X_m[k]|$ is the magnitude of the k th frequency sample of the current frame, and r_m is the frequency sample number that produces the desired 85% rolloff.

A.3.4 Spectral Flatness

Spectral flatness quantifies how close the spectral distribution is to uniform (white). The equation for computing spectral flatness, U_m is as follows,

$$U_m = \frac{K \sqrt[K]{\prod_{k=0}^{K-1} |X_m[k]|}}{\sum_{k=0}^{K-1} |X_m[k]|}, \quad (\text{A.6})$$

where again $|X_m[k]|$ is the magnitude spectrum of the current frame.

A.4 Octave-Based Spectral Contrast

The spectral contrast feature provides a rough representation of the harmonic content in the frequency domain based upon the identification of peaks and valleys in the spectrum, separated into different frequency sub-bands [123]. Whereas the other spectrum-based features (MFCCs, SSDs)

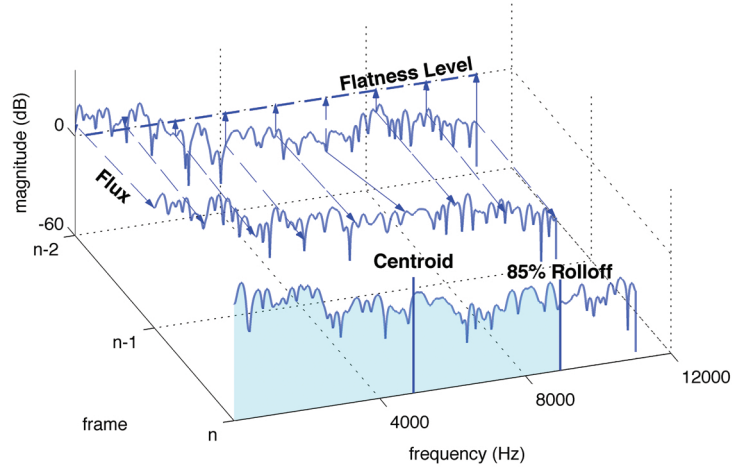


Figure A.2: Graphical depiction of SSDs.

provide information about the wideband characteristics of the signal, spectral contrast focuses on the narrowband (harmonic) content, which is often the dominant component in musical signals.

To compute the spectral contrast features, we first divide the magnitude spectrum into k octave based sub-bands. The frequency range of the first band is 0 - 200 Hz, the second is 200 - 400 Hz, and the remaining sub-bands are similarly calculated by doubling the cutoff frequency of the upper band edge (except for the last band, which has a range of 6400 Hz - Nyquist). Since most meaningful spectral content within music occurs in the lower range of frequencies, we reduce computation by limiting the number of bands to seven.

After the sub-bands have been determined, the peaks, valleys, and differences are calculated. The maxima and minima within a local area are calculated to eliminate spurious values. First, the vector of sub-band magnitudes are ordered in descending order. Then the first αN values, where α is an appropriately tuned constant, are averaged to yield the peak value. The same process is completed for the valleys, except that the vectors are ordered in ascending order. The spectral contrast is simply the difference between the peaks and valleys. The equations for the contrast features are as

follows:

$$P_k = \frac{1}{\alpha N} \sum_{i=1}^{\alpha N} X_{k,i}, \quad (\text{A.7})$$

$$V_k = \frac{1}{\alpha N} \sum_{i=1}^{\alpha N} X_{k,N-i+1}, \quad (\text{A.8})$$

$$SC_k = P_k - V_k. \quad (\text{A.9})$$

Here, α has been defined as 0.02 and the i th bin in the k th sub-band is represented by $X_{k,i}$.

A.5 Echo Nest Timbre

Echo Nest Timbre (ENT) features are a proprietary 12-dimensional acoustic feature developed by the Echo Nest corporation. These features have recently received a great deal of attention due to the release of the million song dataset [124]. The million song dataset is the first publicly released large scale dataset, and because of illegality of releasing audio content, is limited to features used by the Echo Nest. As it would be of great interest to leverage the million song dataset in music emotion recognition, these features are investigated in this work for comparison to existing features.

While the exact computation of these features is not known exactly, a good deal of information is available. Timbre features are extracted on events (i.e. note onsets) throughout a piece of audio, with therefore varying hop and window sizes. Tristan Jehan, the creator of the ENT feature describes them as a “high-res spectrogram through psychoacoustic filters (inner-outer ear filter, dB compression, cochlea warping, frequency and temporal masking) and its resulting auditory surface converted to 12 ‘optimal’ coefficients on a segment basis.” Shown in Figure A.3 are the 12 basis functions, as provided by Tristan Jehan.

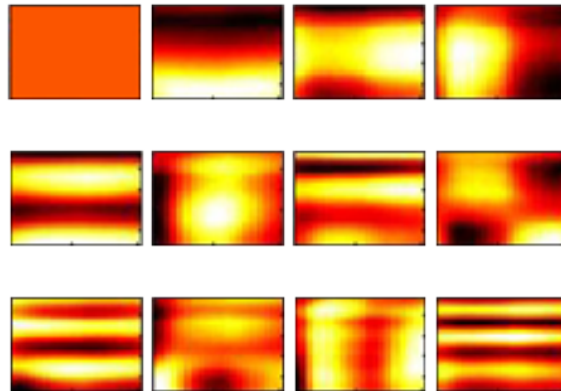


Figure A.3: EchoNest timbre feature basis functions [125].

Appendix B: Restricted Boltzman Machine Tutorial

A restricted Boltzman machine is a type of log-linear Markov random field with hidden units [93, 94, 95, 96, 97]. This section will first discuss the overall topic of energy-based probabilistic models, then the addition of hidden units, and finally sampling and training in RBMs.

B.1 Energy-Based Models

RBMs fall into a category of probabilistic models known as energy-based models. In developing an energy-based model, the variables of interest are associated with a scalar energy quantity. The standard form of such a probabilistic model is as follows,

$$P(\mathbf{x}) = \frac{e^{-\text{Energy}(\mathbf{x})}}{Z}, \quad (\text{B.1})$$

where $\mathbf{x} \in \mathbb{R}^{N \times 1}$. The normalizing function Z is also referred to as the partition function, as it is similar to the function in physical systems that describes equilibrium,

$$Z = \sum_{\mathbf{x}} e^{-\text{Energy}(\mathbf{x})}. \quad (\text{B.2})$$

B.2 Hidden Variables

In order to define the energy function for an RBM, it is necessary to introduce hidden variables. An RBM has a two layer architecture, containing a layer of visible nodes \mathbf{v} and a layer of hidden nodes \mathbf{h} . These models are restricted in the sense that they contain only connections between the visible and hidden layer (i.e., no visible-visible or hidden-hidden connections). A graphical depiction is shown in Figure B.1.

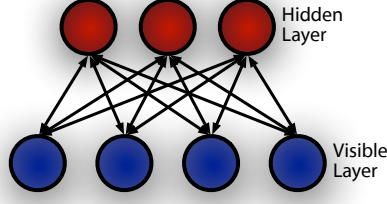


Figure B.1: Restricted Boltzman machine topology.

These generative models express their probability distribution via an energy function, which is shown here as a joint configuration between an observed part \mathbf{v} and hidden part \mathbf{h} [99],

$$E(\mathbf{v}, \mathbf{h}) = - \sum_{i \in \text{visible}} b_i v_i - \sum_{j \in \text{hidden}} a_j h_j - \sum_{i,j} v_i h_j w_{ij}. \quad (\text{B.3})$$

In the experiments discussed in this work (Chapter 7 - 8), the visible layer is a spectrogram $\mathbf{v} \in \mathbb{R}^{I \times 1}$ and the hidden layer $\mathbf{h} \in \mathbb{R}^{J \times 1}$. The model has parameters $W \in \mathbb{R}^{I \times J}$, with biases $a \in \mathbb{R}^{J \times 1}$ and $b \in \mathbb{R}^{I \times 1}$. The probability distribution then becomes,

$$P(\mathbf{v}) = \sum_{\mathbf{h}} P(\mathbf{v}, \mathbf{h}) = \sum_{\mathbf{h}} \frac{e^{-E(\mathbf{v}, \mathbf{h})}}{Z}. \quad (\text{B.4})$$

As only \mathbf{v} is observed, it is necessary to obtain the marginal by summing over all \mathbf{h} . Continuing to borrow notation from physics, “free energy” in the model is expressed as,

$$\mathcal{F}(\mathbf{v}) = -\log \sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}, \quad (\text{B.5})$$

which is just the marginalization of energy in the log domain [96]. Applying this to the RBM distribution, it simplifies to the form of Equation B.1,

$$P(\mathbf{v}) = \frac{e^{-\mathcal{F}(\mathbf{v})}}{Z}, \quad (\text{B.6})$$

and the normalizing factor Z is defined as,

$$Z = \sum_{\mathbf{v}} e^{-\mathcal{F}(\mathbf{v})}. \quad (\text{B.7})$$

B.3 Binomial Hidden Units

The standard approach in RBMs is to use binary input data, and therefore both the hidden and visible layers are binary. But in practice, it has also been found that binomial RBMs work well on real valued data scaled within the range $[0, 1]$, which is interpreted as probabilities of node activation. Given a training vector \mathbf{v} , the probability of hidden node activation is found as follows [97],

$$p(h_j = 1|\mathbf{v}) = \sigma(b_j + \sum_i w_{ij}v_i). \quad (\text{B.8})$$

The function $\sigma(x)$ is the logistic sigmoid function,

$$\sigma(x) = \frac{1}{1 + \exp(-x)}. \quad (\text{B.9})$$

Given a hidden vector, it is also very easy to obtain an unbiased sample of state of a visible units,

$$p(v_i = 1|\mathbf{h}) = \sigma(a_i + \sum_j w_{ij}h_j). \quad (\text{B.10})$$

However, the hidden vector is latent and must be determined in order to train the model. Many methods exist for training models that rely on latent data, the most common being expectation maximization (EM). The method used here is contrastive divergence (CD), which follows a very similar iterative structure as EM. In the first part, inference occurs to estimate the latent variables, and in the second part, the model parameters are updated. The main differences are approximations for tractability and computational complexity, due to the complex nature of the underlying graphical model.

B.4 Sampling in RBMs

In order to perform RBM training, it is necessary to be able to obtain accurate samples of the model's distribution. As the RBM is a Markov chain, this is done through Gibbs sampling. In Gibbs sampling, given a distribution $p(\mathbf{z}) = p(z_1, \dots, z_m)$ and some initial state, the procedure involves replacing one of the variables by a value drawn from the distribution of that variable, and conditioned on all of the remaining variables [100].

In RBMs there are only two distributions of importance, that of the visible and the hidden units. Furthermore, because there are only visible-hidden connections (i.e. independence within layer), all nodes in a layer can be updated at the same time,

$$\mathbf{h}^{(n+1)} \sim \sigma(b_j + \sum_i w_{ij} v_i), \quad (\text{B.11})$$

$$\mathbf{v}^{(n+1)} \sim \sigma(a_i + \sum_j w_{ij} h_j). \quad (\text{B.12})$$

This essentially becomes a procedure of going back and forth between the two layers. Training data can be used at the visible layer to obtain estimates of the hidden layer, and then those hidden layer values are used to obtain estimates of the visible layer. In Figure B.2, this back and forth process can be seen. Note that this figure is different from Figure B.1, because each circle represents update of an entire layer.

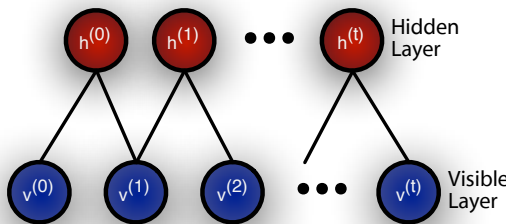


Figure B.2: Gibbs sampling in restricted Boltzmann machines In this figure $v^{(0)}$ refers to all visible nodes and $h^{(0)}$ refers to all hidden nodes at time zero.

More precisely the equations of B.11 - B.12 are used to obtain the estimates for the next layer at the next time step given the state of the current layer at the current time step [96]:

$$\mathbf{v}^{(0)} \sim \text{Training Example}, \quad (\text{B.13})$$

$$\mathbf{h}^{(0)} \sim P(\mathbf{h}|\mathbf{v}^{(0)}), \quad (\text{B.14})$$

$$\mathbf{v}^{(1)} \sim P(\mathbf{v}|\mathbf{h}^{(0)}), \quad (\text{B.15})$$

$$\mathbf{h}^{(1)} \sim P(\mathbf{h}|\mathbf{v}^{(1)}). \quad (\text{B.16})$$

B.5 Training RBMs

The partials of the model distribution with respect to each of the parameters take the form [126],

$$\frac{\partial \log P(\mathbf{v})}{\partial W} = \mathbf{v}' E[\mathbf{h}|\mathbf{v}] - E[\mathbf{v}'\mathbf{h}], \quad (\text{B.17})$$

$$\frac{\partial \log P(\mathbf{v})}{\partial a} = E[\mathbf{h}|\mathbf{v}] - E[\mathbf{h}], \quad (\text{B.18})$$

$$\frac{\partial \log P(\mathbf{v})}{\partial b} = \mathbf{v} - E[\mathbf{v}]. \quad (\text{B.19})$$

The model parameter W is initialized to be a zero mean Gaussian distribution with a standard deviation of usually about 0.01, and bias parameters a and b are set to zero [97]. Following from these equations, sampling is performed on each iteration for up to k steps; in practice, it has been shown that 1 step works quite well [95, 96].

$$W = W + \epsilon(\mathbf{v}^{(0)}\hat{\mathbf{h}}^{(0)'} - \mathbf{v}^{(1)}\hat{\mathbf{h}}^{(1)'}), \quad (\text{B.20})$$

$$a = a + \epsilon(\hat{\mathbf{h}}^{(0)} - \hat{\mathbf{h}}^{(2)}), \quad (\text{B.21})$$

$$b = b + \epsilon(\mathbf{v}^{(0)} - \mathbf{v}^{(1)}), \quad (\text{B.22})$$

where ϵ is the “learning rate” of the model, which controls how much the parameters are allowed to change on each iteration.

Appendix C: Support Vector Machine Tutorial

Motivated by the formulation of the perceptron, an SVM attempts to construct a classification hyperplane of maximal separation between two classes of data [102, 103, 100]. Figure C.1 shows an example of a maximum margin hyperplane that is shown as the solid line. The two parallel transport functions sit up against the data on either side of the hyperplane and will be explained in more detail.

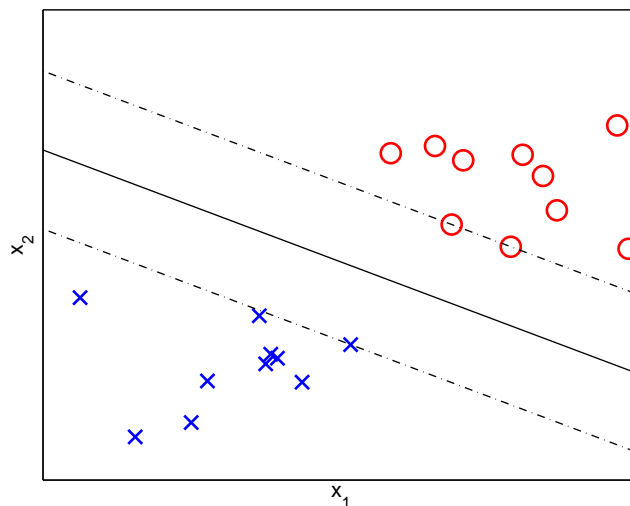


Figure C.1: An example of a maximum margin classifier. In this example x_1 and x_2 are the dimensions of the input data, each blue X represents an example in the negative class, and each red O represents an example in the positive class.

C.1 Perceptron

The hyperplane is expressed through its normal equation, which generalizes over any input data example $\mathbf{x} \in \mathbb{R}^{N \times 1}$, with normal vector $\mathbf{w} \in \mathbb{R}^{N \times 1}$, and has some scalar offset $b \in \mathbb{R}$:

$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b. \quad (\text{C.1})$$

In seeking to determine the parameters (\mathbf{w}, b) of the model, the perceptron has a very loose definition. Any line that satisfies the following constraints for all training inputs \mathbf{x}_i and labels $y_i \in [-1, 1]$ will separate the data:

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) > 0. \quad (\text{C.2})$$

Therefore, there is no unique solution that satisfies this equation. Shown in Figure C.2 are multiple perceptron classification boundaries for the given data. It can be seen that some of the classification boundaries solve the problem quite well while others solve it quite poorly.

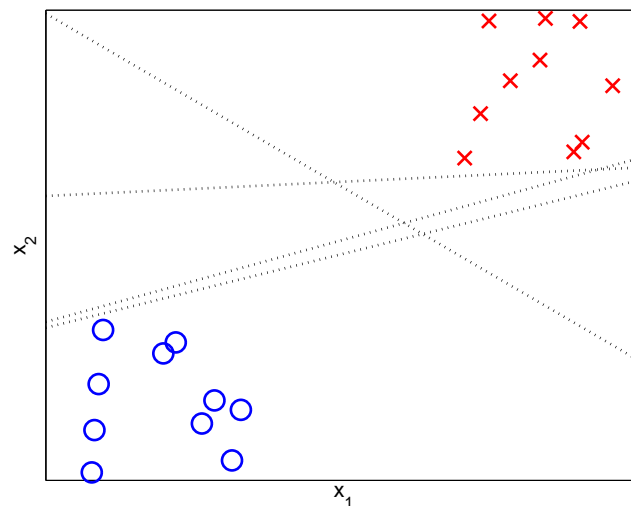


Figure C.2: Four valid perceptron classification boundaries that satisfy the training criteria described in Equation C.2.

C.2 Obtaining a Unique Solution

In order to produce a more optimal classification boundary, it is wished to determine one that provides maximal separation between the two classes of data. This is done by defining two parallel transports such that they are tangential to the data points corresponding to the minimum distance between the two classes, and such that any datapoint lying on one of them evaluates to ± 1 . Referring back to Figure C.1, the solid line is the classification hyperplane, and the dashed lines are the parallel transports. Therefore, if we define two input data examples \mathbf{x}_1 and \mathbf{x}_2 , each lying on one of the two

parallel transports such that:

$$\langle \mathbf{w}, \mathbf{x}_1 \rangle + b = +1, \quad (\text{C.3})$$

$$\langle \mathbf{w}, \mathbf{x}_2 \rangle + b = -1. \quad (\text{C.4})$$

It is then possible to show [127]:

$$\langle \mathbf{w}, (\mathbf{x}_1 - \mathbf{x}_2) \rangle = 2, \quad (\text{C.5})$$

$$\left\langle \frac{\mathbf{w}}{\|\mathbf{w}\|}, (\mathbf{x}_1 - \mathbf{x}_2) \right\rangle = \frac{2}{\|\mathbf{w}\|}. \quad (\text{C.6})$$

Equation C.5 simply states that the functional margin between those two points must equal two. But to maximize the geometric margin, it is necessary to normalize \mathbf{w} , yielding Equation C.6. To determine the maximum margin classifier, it is necessary to maximize $\frac{2}{\|\mathbf{w}\|}$. This function is unattractive for several reasons, primarily because it suffers from a potential division by zero. But it can be easily inferred that minimizing $\frac{1}{2}\|\mathbf{w}\|$ or $\frac{1}{2}\|\mathbf{w}\|^2$ yields an identical solution. As convex optimization will be used for the minimization process, the latter will be chosen, as it is a convex function with a unique minimum value. The full optimization problem can be stated as follows,

$$\text{minimize } \frac{1}{2}\|\mathbf{w}\|^2 \quad (\text{C.7})$$

$$\text{subject to } y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - 1 \geq 0 \text{ for all } 1 \leq i \leq m. \quad (\text{C.8})$$

This will be referred to as the primal form of the optimization problem. While it is possible to solve in the primal form, most off-the-shelf convex optimization algorithms, such as quadratic programming [106], solve this problem in its Lagrange dual form. In the dual form, the optimization problem and

constraints are combined into one equation,

$$L(\mathbf{w}, b, \alpha) = \text{Primal Objective} + \sum_i \alpha_i c_i, \quad (\text{C.9})$$

$$= \frac{1}{2} \|\mathbf{w}\|^2 + \sum_{i=1}^m \alpha_i (1 - y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b)). \quad (\text{C.10})$$

The Lagrange function is specifically nice for optimization because it has a saddlepoint at the optimal solution. It can be seen that L has a minimum with respect to \mathbf{w} and b for the optimal alpha. As a result, it is possible to take the derivatives of L with respect to \mathbf{w} and b and simply optimize in the α 's [127]. After reducing at the saddlepoint, we are left with the optimization problem as follows,

$$\text{minimize } \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle - \sum_{i=1}^m \alpha_i, \quad (\text{C.11})$$

$$\text{subject to } \sum_{i=1}^m \alpha_i y_i = 0 \text{ and } \alpha_i \geq 0. \quad (\text{C.12})$$

Given a trained model it can also be found that,

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i \quad \text{and hence} \quad f(\mathbf{x}) = \sum_{i=1}^m \alpha_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b. \quad (\text{C.13})$$

The Karush Kuhn Tucker (KKT) conditions state that at the optimal solution the constraint multiplied by the Lagrange multiplier is equal to zero,

$$\alpha_i (1 - y_i (\langle \mathbf{x}_i, \mathbf{x} \rangle + b)) = 0, \quad (\text{C.14})$$

and the classification boundary can be expressed in the same form as the perceptron in Equation C.1.

C.3 Kernel Methods

The objective when using kernel functions is to project the data into a space where it is linearly separable, which often involves using a non-linear projection function, resulting in a space of much

higher dimension. Shown in Figure C.3 is single dimensional data that is not linearly separable in the current space in which it is represented; however, if the data is projected to a higher dimensional space using a simple polynomial kernel, it can easily be separated linearly.

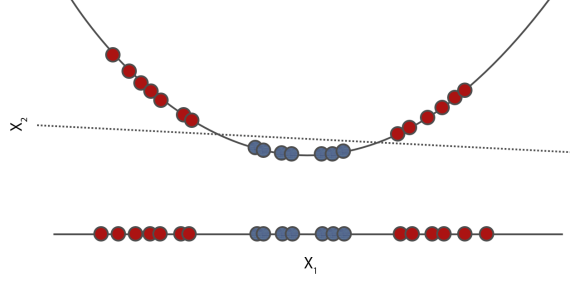


Figure C.3: Simple kernel projection example

In all classification experiments discussed in this work, a distance-based radial basis function (RBF) kernel is used,

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2), \gamma > 0. \quad (\text{C.15})$$

As the kernel space is considered to be of infinite dimension, the data and the normals can now only be defined implicitly with respect to each other. However, the only modification necessary to the optimization problem is to replace the inner product with the kernel function,

$$\text{minimize } \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^m \alpha_i, \quad (\text{C.16})$$

$$\text{subject to } \sum_{i=1}^m \alpha_i y_i = 0 \text{ and } \alpha_i \geq 0. \quad (\text{C.17})$$

As \mathbf{w} is no longer linear (i.e., the kernel must be computed on unseen examples), the decision function can only be expressed as follows,

$$f(\mathbf{x}) = \sum_{i=1}^m \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b, \quad (\text{C.18})$$

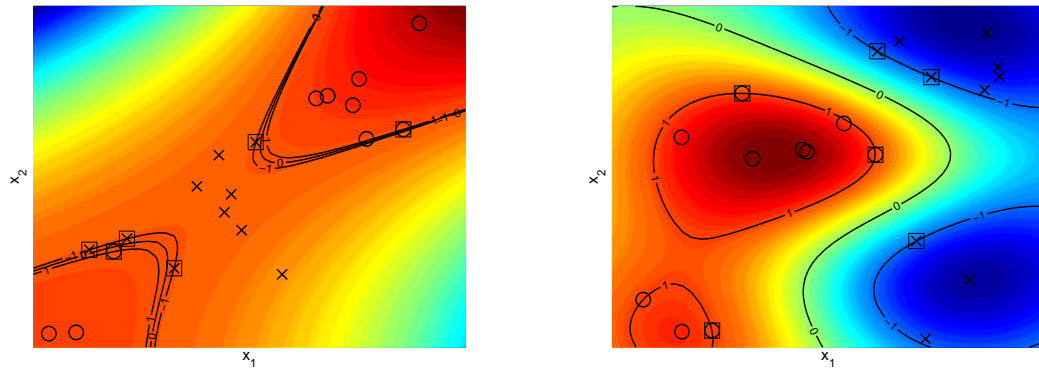


Figure C.4: SVM with a third order polynomial kernel (left) and a radial basis kernel (right). The two classes of data are denoted with X and O. The classification hyperplane is denoted as 0 with parallel transports +1 and -1. Examples that are support vectors are marked with a square \square .

where the \mathbf{x}_i are the support vectors (i.e., training examples that lie on the parallel transports necessary for expressing the hyperplane).

In Figure C.4 two kernel examples are shown. In the left plot a third order polynomial kernel is used, and in the right plot an RBF kernel is implemented. The two classes are marked as x and o, and the contour lines marked as -1, 0, and 1 show the hard decision boundary and its respective parallel transports. In addition, a filled contour plane is shown in the background displaying the full decision function throughout the space.

Appendix D: Kalman/Rauch-Tung-Striebel Estimation

Given an unknown testing example \mathbf{y} , the estimation of the hidden state \mathbf{y} is performed via the following forward-backward Kalman/RTS recursions [108, 109]:

$$\mathbf{x}_t^{t-1} = A\mathbf{x}_{t-1}^{t-1} \quad (\text{D.1})$$

$$V_t^{t-1} = AV_{t-1}^{t-1}A' + Q \quad (\text{D.2})$$

$$K_t = V_t^{t-1}C'(CV_t^{t-1}C' + R)^{-1} \quad (\text{D.3})$$

$$\mathbf{x}_t^t = \mathbf{x}_t^{t-1} + K_t(\mathbf{y}_t - C\mathbf{x}_t^{t-1}) \quad (\text{D.4})$$

$$V_t^t = V_t^{t-1} - K_tCV_t^{t-1} \quad (\text{D.5})$$

Next, we perform the following backward (RTS) recursions:

$$J_{t-1} = V_{t-1}^{t-1}A'(V_t^{t-1})^{-1} \quad (\text{D.6})$$

$$\mathbf{x}_{t-1}^T = \mathbf{x}_{t-1}^{t-1} + J_{t-1}(\mathbf{x}_t^T - A\mathbf{x}_{t-1}^{t-1}) \quad (\text{D.7})$$

$$V_{t-1}^T = V_{t-1}^{t-1} + J_{t-1}(V_t^T - V_t^{t-1})J_{t-1}' \quad (\text{D.8})$$

Appendix E: Conditional Random Fields Tutorial

CRFs are powerful graphical models that are trained to predict the conditional probability $p(\mathbf{y}|\mathbf{x})$ for a sequence of labels \mathbf{y} given a sequence of features \mathbf{x} [110, 111]. Treating our features as deterministic, we retain the rich local subtleties present in the data, which is especially promising in content-based audio analysis where there is no shortage of rich data. Furthermore, the system provides a model of both the relationships between acoustic data and emotion space parameters and also how those relationships evolve over time.

E.1 Definition

Conditional random fields are a type of log-linear model, similar to logistic regression, that are defined using feature functions that produce binary outcomes relating observations \mathbf{x} to states y . These functions are weighted by parameter λ_K , where K is the total number of binary features, each defined by its own feature function f_k . In this work, the linear-chain case of CRFs is investigated, which restricts features to apply on only the current and previous position of the labels, and thus the distribution of a CRF takes the form:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp \left\{ \sum_{k=1}^K \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\}. \quad (\text{E.1})$$

The expression $f_k(y_t, y_{t-1}, \mathbf{x}_t)$ is used to generalize over all feature function types, using the maximal Markov order allowed by the linear-chain model. However, in the experiments discussed in this work both zeroth and first order Markov feature functions are used. Zeroth order Markov features are generally referred to as node features and only rely on the current position of the data and the labels, $f_k(y_t, x_t)$. These feature functions span $N \times L$ binary features, where N is the number of output classes and L is the number of discrete values which x_t can take on. In addition, it is also common to define node features that operate on the value of x_t from both the current as well as past or future time steps, and hence \mathbf{x}_t is therefore notated as a vector. Those define feature

functions to span $N \times L^R$ possibilities, where R is the the number of time steps of encapsulated in \mathbf{x}_t .

First order Markov features are referred to as edge features and can be expressed in terms of $f_k(y_t, y_{t-1}, x_t)$. Edge features produce a total of $N \times N \times K$ binary features, and can therefore be very costly. The general intuition with these features is that the input data x_t might have a different meaning for each current state, depending on each previous state. Furthermore, it is also common to learn node features that are only dependent on the labels, $f_k(y_t, y_{t-1})$. These features model the dynamics of the label space without regard for the features and produce $N \times N$ features.

The constant $Z(\mathbf{x})$ is referred to as the partition function, and is a normalization function for a specific instance or sequence:

$$Z(\mathbf{x}) = \sum_{\mathbf{y}} \exp \left\{ \sum_{k=1}^K \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\}. \quad (\text{E.2})$$

E.2 Estimating CRF Parameters

Following typical maximum likelihood estimation methods, the conditional log-likelihood of the parameters θ can be expressed as follows:

$$\ell(\theta) = \sum_{i=1}^N \log p(\mathbf{y}^{(i)} | \mathbf{x}^{(i)}), \quad (\text{E.3})$$

where $\mathbf{y}^{(i)}$ and $\mathbf{x}^{(i)}$ are sequences of labels and observations, respectively, and N is the total number of label observation pairs. Substituting the definition of $p(\mathbf{y} | \mathbf{x})$ (Equation E.1) yields the following form:

$$\ell(\theta) = \sum_{i=1}^N \sum_{t=1}^T \sum_{k=1}^K \lambda_k f_k(y_t^{(i)}, y_{t-1}^{(i)}, \mathbf{x}_t^{(i)}) - \sum_{i=1}^N \log Z(x^{(i)}). \quad (\text{E.4})$$

After adding in regularization penalty $1/(2\sigma^2)$ this leaves:

$$\ell(\theta) = \sum_{i=1}^N \sum_{t=1}^T \sum_{k=1}^K \lambda_k f_k(y_t^{(i)}, y_{t-1}^{(i)}, \mathbf{x}_t^{(i)}) - \sum_{i=1}^N \log Z(x^{(i)}) - \sum_{k=1}^K \frac{\lambda_k^2}{2\sigma^2}. \quad (\text{E.5})$$

In order to compute maximum likelihood, the partial derivatives are taken with respect to the model parameters, which in this case are simply the weights on the feature functions λ_k . The partial derivatives take the form [111]:

$$\frac{\partial \ell}{\partial \lambda_k} = \sum_{i=1}^N \sum_{t=1}^T f_k(y_t^{(i)}, y_{t-1}^{(i)}, \mathbf{x}_t^{(i)}) - \sum_{i=1}^N \sum_{t=1}^T \sum_{y, y'} f_k(y, y', \mathbf{x}^{(i)}) p(y, y' | \mathbf{x}^{(i)}) - \sum_{k=1}^K \frac{\lambda_k^2}{2\sigma^2}, \quad (\text{E.6})$$

but this function cannot be maximized in closed form, so methods similar to gradient descent are used, with the most common technique being limited-memory BGFS [111].

E.3 Inference

Inference methods for conditional random fields are nearly identical to those used in hidden Markov models [128]. During training, the forward-backward algorithm is used in order to obtain the marginal probabilities $p(y_t, y_{t-1}, \mathbf{x})$ as well as the partition function. In adapting forward-backward to CRFs, the weight vector on the transitions from state i to j $\Psi(i, j, x)$ is defined as [128, 111]:

$$\Psi_t(i, j, x) = \sum_{k=1}^K \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}_t). \quad (\text{E.7})$$

In the same way, Viterbi decoding can be used to find the most likely state combination for an unlabeled observation, \mathbf{x} , but what is more interesting with CRFs is that it is possible to estimate the conditional probability of every state, for every time step in an unlabeled observation [129].

Bibliography

- [1] J. S. Downie, “Music information retrieval,” *Annual Review of Information Science and Technology*, vol. 37, no. 1, pp. 295–340, 2003.
- [2] C. C. Pratt, *Music as the language of emotion*. The Library of Congress, December 1950.
- [3] Y. E. Kim, E. M. Schmidt, R. Migneco, B. G. Morton, P. Richardson, J. Scott, J. A. Speck, and D. Turnbull, “Music emotion recognition: A state of the art review,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Utrecht, Netherlands, 2010.
- [4] J. A. Russell, “A circumplex model of affect,” *J. Personality Social Psychology*, vol. 39, pp. 1161–1178, 1980.
- [5] R. E. Thayer, *The Biopsychology of Mood and Arousal*. Oxford, U.K.: Oxford Univ. Press, 1989.
- [6] E. M. Schmidt, D. Turnbull, and Y. E. Kim, “Feature selection for content-based, time-varying musical emotion regression,” in *MIR ’10: Proceedings of the ACM International Conference on Multimedia Information Retrieval*, Philadelphia, PA, March 2010.
- [7] E. M. Schmidt and Y. E. Kim, “Learning emotion-based acoustic features with deep belief networks,” in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, October 2011.
- [8] J. A. Speck, E. M. Schmidt, B. G. Morton, and Y. E. Kim, “A comparative study of collaborative vs. traditional annotation methods,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Miami, Florida, 2011.
- [9] E. M. Schmidt and Y. E. Kim, “Prediction of time-varying musical mood distributions from audio,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Utrecht, Netherlands, August 2010.
- [10] —, “Prediction of time-varying musical mood distributions using Kalman filtering,” in *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*, Washington, DC, December 2010.
- [11] —, “Modeling musical emotion dynamics with conditional random fields,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Miami, FL, October 2011.
- [12] E. M. Schmidt, M. Prockup, J. Scott, B. Dolhansky, B. Morton, and Y. E. Kim, “Relating perceptual and feature space invariances in music emotion recognition,” in *Proceedings of the International Symposium on Computer Music Modeling and Retrieval (CMMR)*, London, UK, June 2012.
- [13] E. M. Schmidt and Y. E. Kim, “Modeling the acoustic structure of musical emotion with deep belief networks,” in *Neural Information Processing Systems (NIPS) Workshop on Music and Machine Learning*, Sierra Nevada, Spain, December 2011.
- [14] E. M. Schmidt, J. Scott, and Y. E. Kim, “Feature learning in dynamic environments: Modeling the acoustic structure of musical emotion,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Porto, Portugal, October 2012.

- [15] Y. E. Kim, E. Schmidt, and L. Emelle, “Moodswings: A collaborative game for music mood label collection,” in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Philadelphia, PA, September 2008.
- [16] M. Barthet, G. Fazekas, and M. Sandler, “Multidisciplinary perspectives on music emotion recognition: Implications for content and context-based models,” in *Proceedings of the International Symposium on Computer Music Modeling and Retrieval (CMMR)*, London, UK, June 2012.
- [17] P. Juslin and P. Luakka, “Expression, perception, and induction of musical emotions: A review and questionnaire study of everyday listening,” *Journal of New Music Research*, vol. 33, no. 3, p. 217, 2004.
- [18] A. Mehrabian and J. A. Russell, *An Approach to Environmental Psychology*. MIT Press, 1974.
- [19] C. McKay, “Emotion and music: Inherent responses and the importance of empirical cross-cultural research,” Course Paper. McGill University, 2002.
- [20] L.-L. Balkwill and W. F. Thompson, “A cross-cultural investigation of the perception of emotion in music,” *Music Perception*, vol. 17, no. 1, pp. 43–64, 1999.
- [21] T. Fritz, S. Jentschke, N. Gosselin, D. Sammler, I. Peretz, R. Turner, A. D. Friederici, and S. Koelsch, “Universal recognition of three basic emotions in music,” *Current Biology*, vol. 19, no. 7, pp. 573 – 576, 2009.
- [22] K. Hevner, “Experimental studies of the elements of expression in music,” *American Journal of Psychology*, no. 48, pp. 246–268, 1936.
- [23] M. G. Rigg, “Speed as a determiner of musical mood,” *Journal of Experimental Psychology*, vol. 27, pp. 566–571, 1940.
- [24] G. D. Webster and C. G. Weir, “Emotional responses to music: Interactive effects of mode, texture, and tempo,” *Motivation and Emotion*, vol. 29, pp. 19–39, 2005.
- [25] P. N. Juslin, J. Karlsson, E. Lindström, A. Friberg, and E. Schoonderwaldt, “Play it again with feeling: Computer feedback in musical communication of emotions,” *Journal of Experimental Psychology: Applied*, vol. 12, no. 2, pp. 79–95, 2006.
- [26] D. M. Randel, *The Harvard dictionary of music / edited by Don Michael Randel*, 4th ed. Cambridge, MA: Belknap Press of Harvard University Press, 2003.
- [27] L. Gagnon and I. Peretz, “Mode and tempo relative contributions to happy-sad judgements in equitone melodies,” *Cognition & Emotion*, vol. 17, no. 1, pp. 25–40, 2003.
- [28] S. Dalla Bella, I. Peretz, L. Rousseau, and N. Gosselin, “A developmental study of the affective value of tempo and mode in music,” *Cognition*, vol. 80, no. 3, Jul. 2001.
- [29] G. Gerardi and L. Gerken, “The development of affective responses to modality and melodic contour,” *Music Perception*, vol. 12, no. 3, pp. 279–290, 1995.
- [30] G. Husain, W. Thompson, and E. G. Schellenberg, “Effects of musical tempo and mode on arousal, mood, and spatial abilities,” *Music Perception*, vol. 20, no. 2, pp. 151–171, 2002.
- [31] P. Juslin and J. Sloboda, *Music and Emotion: theory and research*. Oxford Univ. Press, 2001.
- [32] E. Schubert, “Update of the Hevner adjective checklist,” *Perceptual and Motor Skills*, vol. 96, pp. 1117–1122, 2003.
- [33] M. Zentner, D. Grandjean, and K. R. Scherer, “Emotions evoked by the sound of music: Characterization, classification, and measurement,” *Emotion*, vol. 8, p. 494, 2008.

- [34] “The All Music Guide.” [Online]. Available: <http://www.allmusic.com>
- [35] X. Hu, J. Downie, C. Laurier, M. Bay, and A. Ehmann, “The 2007 mirex audio mood classification task: Lessons learned,” in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Philadelphia, PA, 2008.
- [36] C. Laurier, M. Sordo, J. Serra, and P. Herrera, “Music mood representation from social tags,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Kobe, Japan, 2009.
- [37] J. Liebetrau, S. Schneider, and R. Jezierski, “Application of free choice profiling for the evaluation of emotions elicited by music,” in *Proceedings of the International Symposium on Computer Music Modeling and Retrieval (CMMR)*, London, UK, June 2012.
- [38] T. Eerola, O. Lartillot, and P. Toivainen, “Prediction of multidimensional emotional ratings in music from audio using multivariate regression models,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Kobe, Japan, 2009.
- [39] L. Mion and G. D. Poli, “Score-independent audio features for description of music expression,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 458–466, 2008.
- [40] E. Bigand, “Multidimensional scaling of emotional responses to music: The effect of musical expertise and of the duration of the excerpts,” *Cognition and Emotion*, vol. 19, no. 8, p. 1113, 2005.
- [41] U. Schimmack and R. Reisenzein, “Experiencing activation: energetic arousal and tense arousal are not mixtures of valence and activation,” *Emotion*, vol. 2, no. 4, p. 412, 2002.
- [42] D. Watson and L. Clark, *PANAS-X: Manual for the Positive and Negative Affect Schedule*, expanded form ed., University of Iowa, 1994.
- [43] A. Tellegen, D. Watson, and L. A. Clark, “On the dimensional and hierarchical structure of affect,” *Psychological Science*, vol. 10, no. 4, pp. 297–303, 1999.
- [44] M. McVicar, T. Freeman, and T. D. Bie, “Mining the correlation between lyrical and audio features and the emergence of mood,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Miami, FL, October 2011.
- [45] M. McVicar and T. D. Bie, “CCA and multi-way extension for investigating common components between audio, lyrics and tags,” in *Proceedings of the International Symposium on Computer Music Modeling and Retrieval (CMMR)*, London, UK, June 2012.
- [46] J. Libeks and D. Turnbull, “Exploring artist image using content-based analysis of promotional photos,” in *Proc. of the Int. Computer Music Conf.*, 2010.
- [47] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, “Semantic annotation and retrieval of music and sound effects,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 16, no. 2, 2008.
- [48] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas, “Multilabel classification of music into emotion,” in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Philadelphia, PA, 2008.
- [49] F. Miller, M. Stiksel, and R. Jones, “Last.fm in numbers,” *Last.fm press material*, February 2008.
- [50] P. Lamere and O. Celma, “Music recommendation tutorial notes,” ISMIR Tutorial, September 2007.

- [51] L. von Ahn, “Games with a purpose,” *Computer*, vol. 39, no. 6, pp. 92–94, 2006.
- [52] M. I. Mandel and D. P. W. Ellis, “A web-based game for collecting music metadata,” in *Proceedings of the 8th International Conference on Music Information Retrieval*, Vienna, Austria, ISMIR 2007, pp. 365–366.
- [53] D. Turnbull, R. Liu, L. Barrington, and G. Lanckriet, “A game-based approach for collecting semantic annotations of music,” in *Proceedings of the 8th International Conference on Music Information Retrieval*, Vienna, Austria, ISMIR 2007, pp. 535–538.
- [54] E. L. M. Law, L. von Ahn, R. B. Dannenberg, and M. Crawford, “TagATune: a game for music and sound annotation,” in *Proceedings of the 8th International Conference on Music Information Retrieval*, Vienna, Austria, 2007.
- [55] L. Barrington, D. Turnbull, D. O’Malley, and G. Lanckriet, “User-centered design of a social game to tag music,” *ACM KDD Workshop on Human Computation*, 2009.
- [56] P. Ipeirotis, “Demographics of mechanical turk,” in *CeDER Working Papers*. NYU Stern School of Business, 2010.
- [57] R. Snow, B. O’Connor, D. Jurafsky, and A. Ng, “Cheap and fast - but is it good? Evaluating non-expert annotations for natural language tasks,” in *Proc. Empirical Methods in NLP*, 2008.
- [58] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movellan, “Whose vote should count more: Optimal integration of labels from labelers of unknown expertise,” in *Advances in neural information processing systems*. MIT Press, 2009.
- [59] A. Sorokin and D. Forsyth, “Utility data annotation with Amazon mechanical turk,” in *CVPR Workshops*, 2008.
- [60] J. H. Lee, “Crowdsourcing music similarity judgments using mechanical turk,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Utrecht, Netherlands, August 2010.
- [61] M. I. Mandel, D. Eck, and Y. Bengio, “Learning tags that vary within a song,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Utrecht, Netherlands, August 2010.
- [62] S. Davis and P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences,” *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 28, no. 4, pp. 357–366, 1980.
- [63] B. Logan, “Mel frequency cepstral coefficients for music modeling,” in *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Plymouth, MA, September 2000.
- [64] K. F. MacDorman, S. Ough, and C.-C. Ho, “Automatic emotion prediction of song excerpts: Index construction, algorithm design, and empirical comparison,” *Journal of New Music Research*, vol. 36, no. 4, pp. 281–299, 2007.
- [65] MIRtoolbox. [Online]. Available: <https://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mirtoolbox>
- [66] L. Lu, D. Liu, and H. J. Zhang, “Automatic mood detection and tracking of music audio signals,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 5–18, 2006.
- [67] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. Chen, “A regression approach to music emotion recognition,” *Audio, Speech, and Language Processing, IEEE Trans. on*, vol. 16, no. 2, pp. 448–457, 2008.

- [68] H. Lee, Y. Largman, P. Pham, and A. Ng, “Unsupervised feature learning for audio classification using convolutional deep belief networks,” in *Advances in neural information processing systems*. Cambridge, MA: MIT Press, 2009, pp. 1096–1104.
- [69] P. Hamel and D. Eck, “Learning features from music audio with deep belief networks,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Utrecht, Netherlands, August 2010.
- [70] T. Li and M. Ogihara, “Detecting emotion in music,” in *Proceedings of the 4th International Conference on Music Information Retrieval*, Baltimore, MD, October 2003.
- [71] M. I. Mandel, G. E. Poliner, and D. P. W. Ellis, “Support vector machine active learning for music retrieval,” *Multimedia Systems*, vol. 12, no. 1, pp. 3–13, Aug 2006.
- [72] J. Skowronek, M. McKinney, and S. Par, “A demonstrator for automatic music mood estimation,” in *Proceedings of the 8th International Conference on Music Information Retrieval*, Vienna, Austria, 2007.
- [73] Y. Vaizman, R. Y. Granot, and G. Lanckriet, “Modeling dynamic patterns for emotional content in music,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Miami, FL, October 2011.
- [74] G. Tzanetakis, “Marsyas submissions to MIREX 2007,” MIREX 2007.
- [75] G. Peeters, “A generic training and classification system for MIREX08 classification tasks: Audio music mood, audio genre, audio artist and audio tag,” MIREX 2008.
- [76] C. Cao and M. Li, “Thinkit’s submissions for MIREX2009 audio music classification and similarity tasks.” ISMIR, MIREX 2009.
- [77] A. Smola and B. Schölkopf, “A tutorial on support vector regression,” *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, 2004.
- [78] D. Shrestha and D. Solomatine, “Experiments with adaboost. rt, an improved boosting scheme for regression,” *Neural computation*, vol. 18, no. 7, pp. 1678–1710, 2006.
- [79] D. Cabrera, S. Ferguson, and E. Schubert, “Pysound3: software for acoustical and psychoacoustical analysis of sound recordings,” in *Proc. of the Intl. Conf. on Auditory Display*, Montreal, Canada, June 26-29 2007, pp. 356–363.
- [80] G. Tzanetakis and P. Cook, “Marsyas: a framework for audio analysis,” *Organized Sound*, vol. 4, no. 3, pp. 169–175, 1999.
- [81] B. Han, S. Rho, R. B. Dannenberg, and E. Hwang, “Smers: Music emotion recognition using support vector regression,” in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, Kobe, Japan, 2009.
- [82] J. Madsen, J. B. Nielsen, B. S. Jensen, and J. Larsen, “Modeling expressed emotions in music using pairwise comparisons,” in *Proceedings of the International Symposium on Computer Music Modeling and Retrieval (CMMR)*, London, UK, June 2012.
- [83] B. G. Morton, J. A. Speck, E. M. Schmidt, and Y. E. Kim, “Improving music emotion labeling using human computation,” in *HCOMP ’10: Proc. of the ACM SIGKDD Workshop on Human Computation*, Washington, DC, 2010.
- [84] The “uspop2002” Pop Music data set. [Online]. Available: <http://labrosa.ee.columbia.edu/projects/musicsim/uspop2002.html>
- [85] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley-Interscience, 2001.

- [86] Amazon Mechanical Turk (MTurk). [Online]. Available: <http://mturk.com>
- [87] Beatles midi music homepage. [Online]. Available: <http://earlybeatles.com/>
- [88] D. Ellis. Chroma features analysis and synthesis. [Online]. Available: <http://www.ee.columbia.edu/~dpwe/resources/matlab/chroma-ansyn>
- [89] ——. Plp and rasta in matlab. [Online]. Available: <http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/>
- [90] The Music Information Retrieval Evaluation eXchange (MIREX). [Online]. Available: <http://www.music-ir.org/mirex>
- [91] J.-C. Wang, H.-Y. Lo, S.-K. Jeng, and H.-M. Wang, “Mirex 2010: Audio classification using semantic transformation and classifier ensemble,” in *MIREX*, 2010.
- [92] E. C. Smith and M. S. Lewicki, “Efficient auditory coding,” *Nature*, vol. 439, no. 7079, pp. 978–82, Feb 2006.
- [93] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 313, no. 5786, pp. 504–7, Jul 2006.
- [94] G. E. Hinton, S. Osindero, and Y. Teh, “A fast learning algorithm for deep belief nets,” *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
- [95] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, “Greedy layer-wise training of deep networks,” in *Advances in neural information processing systems*. Cambridge, MA: MIT Press, 2007, pp. 153–160.
- [96] Y. Bengio, “Learning deep architectures for AI,” *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.
- [97] G. Hinton, “A practical guide to training restricted Boltzmann machines,” University of Toronto, Tech. Rep. UTML TR 2010–003, 2010.
- [98] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio, “Theano: A CPU and GPU math expression compiler,” in *Proceedings of the Python for Scientific Computing Conference (SciPy)*, 2010.
- [99] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities,” *Proceedings of the National Academy of Sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.
- [100] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2006.
- [101] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 2000.
- [102] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, pp. 273–295, 1995.
- [103] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2000.
- [104] L. Kuncheva, J. Bezdek, and R. Duin, “Decision templates for multiple classifier fusion: an experimental comparison,” *Pattern Recognition*, vol. 34, no. 2, pp. 299–314, 2001.
- [105] B. Taskar. University of Pennsylvania CS520 Lecture Notes. [Online]. Available: <https://alliance.seas.upenn.edu/~cis520/wiki>
- [106] S. Boyd, *Convex Optimization*. Cambridge University Press, 2004.

- [107] S. Siddiqi, B. Boots, and G. Gordon, “A constraint generation approach to learning stable linear dynamical systems,” in *Advances in Neural Information Processing Systems (NIPS)*. Cambridge, MA: MIT Press, 2008, pp. 1329–1336.
- [108] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [109] Z. Ghahramani and G. E. Hinton, “Parameter estimation for linear dynamical systems,” University of Toronto, Tech. Rep. CRG-TR-96-2, 1996.
- [110] J. Lafferty, A. McCallum, and F. Pereira, “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data,” in *ICML*, 2001.
- [111] C. Sutton and A. McCallum, “An introduction to conditional random fields for relational learning,” in *Introduction to Statistical Relational Learning*, L. Getoor and B. Taskar, Eds. MIT Press, 2007, ch. 4, pp. 93–127.
- [112] Y. Kim, D. Williamson, and S. Pilli, “Towards quantifying the “album effect” in artist identification,” in *Proceedings of ISMIR 2006 Seventh International Conference on Music Information Retrieval*, September 2006.
- [113] E. Pampalk, “Computational models of music similarity and their application in music information retrieval,” Ph.D. dissertation, Johannes Kepler University, Linz, March 2006.
- [114] L. L. Lapin, *Probability and statistics for modern engineering*, 2nd ed. PWS-KENT, 1990.
- [115] MoodSwings Single Player. [Online]. Available: <http://music.ece.drexel.edu/mssp>
- [116] StarCluster. [Online]. Available: <http://web.mit.edu/star/cluster/>
- [117] Amazon Elastic Compute Cloud (EC2). [Online]. Available: <http://aws.amazon.com/ec2/>
- [118] MoodSwings Turk Dataset. [Online]. Available: <http://music.ece.drexel.edu/research/emotion/moodswingsturk>
- [119] CRF++. [Online]. Available: <http://crfpp.sourceforge.net/>
- [120] O. Pele and M. Werman, “Fast and robust earth mover’s distances,” in *ICCV*, 2009.
- [121] M. A. Bartsch and G. H. Wakefield, “To catch a chorus: Using chroma-based representations for audio thumbnailing,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, October 2001.
- [122] G. Tzanetakis and P. Cook, “Musical genre classification of audio signals,” *Speech and Audio Processing, IEEE Transactions on*, vol. 10, no. 5, pp. 293–302, 2002.
- [123] D. Jiang, L. Lu, H. Zhang, J. Tao, and L. Cai, “Music type classification by spectral contrast feature,” in *Proc. Intl. Conf. on Multimedia and Expo*, vol. 1, 2002, pp. 113–116.
- [124] T. Bertin-Mahieux, D. P. W. Ellis, B. Whitman, and P. Lamere, “The million song dataset,” in *Proceedings of the Society for Music Information Retrieval (ISMIR) Conference*, Miami, FL, October 2011.
- [125] T. Jehan, “Creating music by listening,” Ph.D. dissertation, Massachusetts Institute of Technology, September 2005.
- [126] Y. Bengio. (2012) Probabilistic models for deep architectures. [Online]. Available: <http://www.iro.umontreal.ca/~bengioy/ift6266/H12/html/deepgm.en.html>
- [127] A. Smola, “Kernel methods and support vector machines,” Machine Learning Summer School Lecture Notes, March 2008.

- [128] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, Feb 1989.
- [129] A. Culotta and A. McCallum, “Confidence estimation for information extraction,” in *Human Language Technology (HLT) Conference*, 2004.

