



Open Research Online

The Open University's repository of research publications
and other research outputs

Improved methods for pattern discovery in music, with applications in automated stylistic composition

Thesis

How to cite:

(2011). Improved methods for pattern discovery in music, with applications in automated stylistic composition. PhD thesis The Open University.

For guidance on citations see [FAQs](#).

© 2011 Tom Collins
Version: Version of Record

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online's [data policy](#) on reuse of materials please consult the policies page.

oro.open.ac.uk

THOMAS EDWARD COLLINS

BA Cantab, BA Oxon

**Improved methods for
pattern discovery in music,
with applications in
automated stylistic composition**



Submitted 1 August, 2011

for the degree of Doctor of Philosophy

Faculty of Mathematics, Computing and Technology,
The Open University

Supervisors:

Robin Laney, Alistair Willis, and Paul H. Garthwaite

Abstract

Computational methods for intra-opus pattern discovery (discovering repeated patterns within a piece of music) and stylistic composition (composing in the style of another composer or period) can offer insights into how human listeners and composers undertake such activities. Two studies are reported that demonstrate improved computational methods for pattern discovery in music. In the first, regression models are built with the aim of predicting subjective assessments of a pattern's salience, based on various quantifiable attributes of that pattern, such as the number of notes it contains. Using variable selection and cross-validation, a formula is derived for rating the importance of a discovered pattern. In the second study, a music analyst undertook intra-opus pattern discovery for works by Domenico Scarlatti and Johann Sebastian Bach, forming a benchmark of target patterns. The performance of two existing algorithms and one of my own creation, called SIACT (Structure Induction Algorithm with Compactness Trawling), is evaluated by comparison with this benchmark. SIACT out-performs the existing algorithms with regard to recall and, more often than not, precision. A third experiment is reported concerning human judgements of music excerpts that are, to varying degrees, in the style of mazurkas by Frédéric Chopin. This acts as an evaluation for two computational models of musical style, called Racchman-Oct2010 and Racchmaninof-Oct2010 (standing for RAndom Constrained CHain of MArkovian Nodes with INheritance Of Form), which are developed over two chapters. The latter of these models applies SIACT and the formula for rating pattern importance, using temporal and registral positions of discovered patterns from an existing template piece to guide the generation of a new passage of music. The precision and runtime of pattern discovery algorithms, and their use for audio summarisation are among topics for future work. Data and code related to this thesis is available on the accompanying CD (or at <http://www.tomcollinsresearch.net>)

Acknowledgements

Thank you to Rachael for filling the Nottingham years with love and laughter, and for not letting me work at weekends. Can we get over our fundamental disagreement concerning the use of italics? I *hope* so. Not only have my parents and family tolerated a near-decade in higher education, they have supported me lovingly (and often financially) throughout this time. I am deeply grateful to the whole family, but especially to my parents, to Amy and Alex, and to my grandparents.

My supervisors Robin, Alistair, and Paul have been superb. Thank you for giving me the freedom to research creatively, for offering expert advice, and for the time spent reading and discussing my work. Thank you to Jeremy Thurlow at Robinson College for undertaking the pattern discovery tasks, which has made it possible to evaluate various algorithms. Thanks also to Jeremy and many previous supervisors and teachers for giving me an appetite for learning and independent thinking.

There is a strong sense of community among PhD students at The Open University—thanks fellow students for contributing to this, thanks to Marian and Robin for organising postgraduate forum, and to Catherine, Debbie, Mary, Debby, and Gemma for all their help.

I have stayed with a lot of friends and family in order to attend various events during the PhD. My thanks to Rachael's family (especially the Percivals), the Burridge family, Ryan, Louisa and Simon, Kenny, Hendo and Chris, Guy and Nina, Tiggy, Chris and Ed, Tim, Becca, Chris and Rhian, Jamie and Kirsten, Alec and Hannah, James MacLaurin, Annie and Amy, Ella, Devin, Nicki and Phil, Ben and Steph, Viz, Edd, and Rob, Jack Perrins, Becky, Vicky and Mark, Will and Ellie.

Regards to the colleagues/revellers I have got to know through various workshops and conferences: Hendrik, Thomas, Anastassia, Gökce, Charles, VJ, Will, Andres, Jesse, Dave, Peter, and Soren.

Some of the music data for this project came from Kern Scores, so a huge thank you to Craig Stuart Sapp and the Center for Computer Assisted Research in the Humanities at Stanford University. I am grateful to over fifty individuals who have participated in experiments for me over the course of the project. I am indebted also to David Temperley, Dave Meredith, Jamie Forth, and several anonymous reviewers for their advice on publications related to my thesis. Last but not least, many thanks to my examiners, Mr. Chris Dobbyn and Prof. Geraint Wiggins, for finding the time to assess this work.

Tom Collins,
Nottingham, home of Robin Hood,
August 2011.

Related publications

Chapters 4 and 6 are based on the following journal paper.

Tom Collins, Robin Laney, Alistair Willis, and Paul H. Garthwaite. Modelling pattern importance in Chopin's mazurkas. *Music Perception*, 28(4):387-414, 2011.

Chapters 4 and 7 are based on the following conference paper.

Tom Collins, Jeremy Thurlow, Robin Laney, Alistair Willis, and Paul H. Garthwaite. A comparative evaluation of algorithms for discovering translational patterns in Baroque keyboard works. In J. Stephen Downie and Remco Veltkamp, editors, *Proceedings of the International Symposium on Music Information Retrieval*, pages 3-8, Utrecht, 2010. International Society for Music Information Retrieval.

The following conference paper describes a first attempt at computational modelling of musical style.

Tom Collins, Robin Laney, Alistair Willis, and Paul H. Garthwaite. Using discovered polyphonic patterns to filter computer-generated music. In Dan Ventura, Alison Pease, Rafael Pérrez y Pérrez, Graeme Ritchie, and Tony Veale, editors, *Proceedings of the International Conference on Computational Creativity*, pages 1-10, Lisbon, 2010. University of Coimbra.

Contents

List of tables	xii
List of figures	xiv
1 Introduction	1
2 Music representations	9
2.1 Musical instrument digital interface (MIDI)	9
2.2 The generalised interval system and viewpoints	13
2.3 Geometric representations	19
2.4 Some music-analytical tools	22
3 Calculating probabilities and statistics in music	29
3.1 The empirical probability mass function	29
3.2 Empirical distributions and models of music perception	37
3.3 An introduction to Markov models	44
4 Discovery of patterns in music	51
4.1 Algorithms for pattern discovery in music	60
4.2 The family of Structure Induction Algorithms	70
4.3 Recall and precision	74
4.4 The SIA family applied beyond the musical surface	76
5 Algorithmic composition	81
5.1 Motivations	81
5.2 Example briefs in stylistic composition	86
5.3 Early models of musical style	95
5.4 Recurring questions of the literature survey	101
5.5 The use of viewpoints to model musical style	102
5.6 Experiments in Musical Intelligence (EMI)	105

5.6.1	Syntactic meshing	106
5.6.2	Semantic meshing	110
5.6.3	Signatures	118
5.6.4	Templagiarism	120
5.7	Issues of evaluation	122
6	Rating discovered patterns	125
6.1	Method	128
6.1.1	Participants and instructions	128
6.1.2	Selection of excerpts and patterns	132
6.1.3	Explanatory variables	133
6.2	Results	139
6.2.1	Predictive value of individual variables	147
6.2.2	Predictive ability of participants and the formula	148
6.3	Discussion	150
6.3.1	Conclusions	150
6.3.2	Future work	161
7	The recall of pattern discovery algorithms	165
7.1	The problem of isolated membership	165
7.2	Evaluation	170
7.3	Discussion	174
7.3.1	Conclusions	174
7.3.2	Future work	180
8	Markov models of stylistic composition	183
8.1	Realisation and musical context	184
8.2	Orders and state spaces	187
8.3	A beat-spacing state space	192
8.3.1	Partition point, minimal segment, semitone spacing	192
8.3.2	Details of musical context to be retained	201
8.4	Random generation Markov chain	203
8.5	Periodic and absorbing states	211
9	Generating patterned stylistic compositions	215
9.1	Stylistic shortcomings of RGMC	215
9.1.1	Sources	216
9.1.2	Range	217
9.1.3	Low-likelihood chords	218
9.1.4	A sense of departure and arrival	224
9.2	RAndom Constrained CHain of MArkovian Nodes	227

9.3 Pattern inheritance	231
10 Evaluating models of stylistic composition	241
10.1 Evaluation questions	241
10.2 Methods for answering evaluation questions	242
10.3 Judges and instructions	245
10.4 Selection and presentation of stimuli	249
10.5 Results	255
10.5.1 Answer to evaluation question 3	255
10.5.2 Answer to evaluation question 1	256
10.5.3 Answer to evaluation question 2	260
10.5.4 Answer to evaluation question 4	261
10.5.5 Answer to evaluation question 5	262
10.6 Conclusions and future work	265
11 Conclusions and future work	271
11.1 Conclusions	271
11.1.1 Revisiting hypotheses from the introduction	273
11.1.2 The precision and runtime of SIA	278
11.2 Future work	283
11.2.1 SIACT and audio summarisation	284
Appendices	291
A Mathematical definitions	291
B Introduction to music representations	327
B.1 Audio, and mathematical definitions	332
B.2 Symbolic representation of music	341
B.2.1 Staff notation	341
B.2.2 The elements of staff notation	343
B.2.3 MusicXML and kern	353
B.2.4 An object-oriented approach	360
B.3 Primary concepts of music theory	361
C Explanatory variables included in the regressions	367
D Top-rated patterns in Chopin's op.56 no.1	385
E Four computer-generated mazurka sections	399
References	405

List of Tables

5.1	Reproduced from Pearce et al. (2002). Motivations for developing computer programs which compose music	82
5.2	Assessment criteria for composition unit, adapted from AQA (2009)	96
6.1	Fittings for individual pattern attributes and block variables .	141
6.2	Consensus test results	149
6.3	Ratings and various attributes for patterns <i>E</i> , <i>F</i> , <i>G</i> , and <i>I</i> .	153
7.1	Results for three algorithms on the intra-opus pattern discovery task	173
10.1	Mean stylistic success ratings and classification percentages for stimuli	257
10.2	Contrasts for two ANOVAs, one conducted using concertgoer ratings of stylistic success as the response variable, the other using expert ratings	259
A.1	The rhythmic density of various opening movements from known and supposed Vivaldi 'cello concertos	318

List of Figures

2.1	Piano-roll notation for bars 50 and 51 of Fig. B.8 (performed and accurate versions)	12
2.2	Viewpoint lists for eighteen notes extracted from Fig. B.8	16
2.3	A digraph representing plausible (black) and implausible (red) successions of notes in an excerpt of music	18
2.4	Plots (of MPN against ontine, and duration against ontine) for dataset representations of bars 50-53 from Fig. B.8	21
2.5	The output of HarmAn (Pardo and Birmingham, 2002) when applied to Fig. B.8	25
2.6	A keyscape for the output of a key finding algorithm (Sapp, 2005) when applied to Fig. B.8	27
2.7	The output of an automatic transcription algorithm available with a program called Melodyne, as applied to a portion of audio from ‘To the end’ by Blur (1994)	28
3.1	Bars 1-19 of <i>The unanswered question</i> (c 1930-1935) by Charles Ives	32
3.2	The empirical probability mass function for MIDI note numbers (MNN) from Fig. 3.1	33
3.3	The empirical probability mass function for pairs of MMN modulo 12 and duration from Fig. 3.1	34
3.4	The empirical probability mass function for MMN modulo 12 weighted by duration from Fig. 3.1	35
3.5	Two empirical probability mass functions plotted side by side for different dataset windows from Fig. 3.1	37
3.6	The likelihood profile for the excerpt shown in Fig. 3.1 and dataset defined in (3.1)	42
3.7	Bars 3-10 of the melody from ‘Lydia’ op.4 no.2 by Gabriel Fauré	45

4.1	Bars 1-4 from the first movement of the Piano Sonata no.11 in B \flat major op.22 by Ludwig van Beethoven, with annotations	52
4.2	Bars 1-4 from the first movement of the Piano Sonata no.8 in A minor K310 by Wolfgang Amadeus Mozart, with annotations	53
4.3	Bars 1-4 and 25-28 from the fourth movement of the Octet in F major D803 by Franz Schubert, with annotations	53
4.4	Bars 38-43 and 131-136 from the first movement of the Piano Concerto no.1 by Béla Bartók, with annotations	55
4.5	Bars 18-24 from the Allemande of the Chamber Sonata in B minor op.2 no.8 by Arcangelo Corelli, with annotations	56
4.6	Excerpts from ‘Albanus roseo rutilat’ by John Dunstable, with annotations	57
4.7	Bars 1-12 from the first movement of the String Quartet in G minor, ‘The Horseman’, op.74 no.3 by Joseph Haydn, with annotations	59
4.8	Flow chart depicting a framework for a pattern matching system	63
4.9	Flow chart depicting a framework for a pattern discovery system	64
4.10	Bars 1-3 of the Introduction from <i>The Rite of Spring</i> (1913) by Igor Stravinsky, with annotations	65
4.11	Bars 13-16 of the Sonata in C major L3 by Domenico Scar- latti; a plot of morphetic pitch number against ontime for this excerpt; the same plot annotated with patterns	67
4.12	A Venn diagram (not to scale) for the number of patterns (up to translational equivalence) in a dataset, as well as the typical number of patterns returned by various algorithms	72
4.13	A Venn diagram to show different collections of musical pat- terns for a retrieval task	76
4.14	Reproduced from Volk (2008). Bars 62-65 of <i>Moment musical</i> op.94 no.4 by Schubert, annotated with onsets, local metres and extensions	78
5.1	Bars 1-8 of the Mazurka in G minor from <i>Soirées musicales</i> op.6 no.3 by Clara Schumann	84
5.2	Bars 1-10 of ‘Moro, lasso’ from Madrigals book 6 by Carlo Gesualdo, Prince of Venosa, Count of Conza	85
5.3	Bars 46-54 of the first movement from the Symphony no.1 in D major, ‘The Classical’, op.25 by Sergey Prokofiev	85
5.4	A melody for harmonisation in the style of Johann Sebastian Bach (AQA, 2009)	87

5.5	A ground bass by Gottfried Finger above which parts for string or wind ensemble are to be added (Cambridge University Faculty of Music, 2010a)	87
5.6	Five subjects, one to be chosen for development as a fugal exposition (Cambridge University Faculty of Music, 2010a) . .	88
5.7	A graph showing the typical conditional dependence structure of a hidden Markov model	91
5.8	Segments of music adapted from a dice game attributed to Mozart, K294d	98
5.9	A graph with vertices that represent bar-length segments of music from Fig. 5.8	99
5.10	Bars 1-8 of the Mazurka in G minor op.67 no.2 by Frédéric Chopin	107
5.11	Graphs for new dice games based on segments from Figs. 5.1 and 5.10	108
5.12	Bars 1-8 of the Mazurka in G minor op.67 no.2 by Chopin, annotated with my SPEAC analysis	112
5.13	Bars 1-28 of the Mazurka no.4 in E minor by David Cope with Experiments in Musical Intelligence	116
5.14	Bars 1-28 of the Mazurka in F minor op.68 no.4 by Chopin .	117
6.1	Bars 1-20 from the Mazurka in G \sharp minor op.33 no.1 by Chopin, with annotations	131
6.2	A rhythmic representation of bars 1-20 from the Mazurka in G \sharp minor op.33 no.1 by Chopin, with annotations	134
6.3	Bars 1-20 from the Mazurka in G \sharp Minor op.33 no.1 by Chopin, with annotations	135
6.4	A plot of the forward model's mean prediction against the observed mean prediction for each of the ninety patterns	144
6.5	A plot of the backward model's mean prediction against the observed mean prediction for each of the ninety patterns	145
6.6	Observed and predicted ratings for patterns 1-20 (from the first two excerpts)	146
6.7	Bars 45-68 from the Mazurka in C major op.56 no.2 by Chopin, with annotations	154
6.8	A rhythmic representation of bars 45-68 from the Mazurka in C major op.56 no.2 by Chopin, with annotations	155
6.9	Bars 40-52 from the Mazurka in C \sharp minor op.41 no.1 by Chopin, with annotations	156
6.10	Bars 45-68 from the Mazurka in C major op.56 no.2 by Chopin, with annotations	159

6.11	Bars 17-36 from the Mazurka in C minor op.30 no.1 by Chopin, with annotations	160
6.12	Box-and-whisker plots to explore the relationship between model performance and excerpt length	162
7.1	Bars 1-19 from the Sonata in C minor L10 by D. Scarlatti, with annotations	171
7.2	Filtered and rated results when SIACT is applied to a repre- sentation of bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	178
7.3	A digraph with vertices representing patterns of the same name from Fig. 7.2	179
7.4	A digraph with vertices representing patterns in a dataset . .	181
8.1	Reproduction of Fig. 3.7. Bars 3-10 of the melody from ‘Lydia’ op.4 no.2 by Gabriel Fauré	185
8.2	Bars 1-13 of ‘If ye love me’ by Thomas Tallis	194
8.3	Bars 1-13 of the ‘If ye love me’ by Tallis, annotated with par- tition points and minimal segments	196
8.4	Bars 23-27 of the Mazurka in B major op.63 no.1 by Chopin .	200
8.5	Bars 15-19 of the Mazurka in C minor op.56 no.3 by Chopin, as notated and as it would be played	201
8.6	Bars 115-120 of the Mazurka in C major op.24 no.2 by Chopin, with annotations	203
8.7	Realised generated output of a random generation Markov chain (RGMC) for the model $(I^{(4)}, L^{(4)}, A^{(4)})$	208
9.1	Bars 1-8 of the Mazurka in E minor op.41 no.2 by Chopin, with two corresponding likelihood profiles	220
9.2	Bars 1-9 of the Mazurka in B major op.56 no.1 by Chopin; pseudo-plots of lowest- and highest sounding, and mean MNNs against ontime; two likelihood profiles; realised generated out- put of a constrained RGMC	223
9.3	Bars 1-2 of the chorale ‘Herzlich lieb hab ich dich, o Herr’, as harmonised (R107, BWV245.40) by J.S. Bach	225
9.4	Passages generated by forwards and backwards random gen- eration Markov chains	230
9.5	A representation of the information retained in a <i>template with patterns</i>	233

9.6	Passage generated by the model Racchmaninof-Oct2010, standing for RAndom Constrained Chain of MArkovian Nodes with INheritance Of Form. Used in the evaluation in Chapter 10 as stimulus 29	237
11.1	Three plots of MPN against ontine for bars 13-16 of the Sonata in C major L3 by D. Scarlatti, annotated with patterns and vectors	281
11.2	A plot of MNN against ontine for bars 1-8 of ‘To the end’ by Blur (1994), from the Melodyne automatic transcription algorithm, annotated patterns <i>A</i> , <i>B</i> , <i>B'</i> , and <i>C</i>	286
A.1	Plots of the sinusoidal functions sine and cosine	300
A.2	A cube before, during, and after a rotation	303
B.1	A portion of the audio signal from the song ‘To the end’ by Blur (1994)	329
B.2	A transcription of bars 1-8 from ‘To the end’ by Blur (1994)	330
B.3	Transcription of bars 12-13 (with upbeat) from ‘Little wing’ by Jimi Hendrix Experience (1967)	331
B.4	A zoomed-in portion of the audio signal from the song ‘To the end’ by Blur (1994)	334
B.5	Separate and superposed harmonics, approximating the portion of audio signal from ‘To the end’ by Blur (1994)	340
B.6	Facsimile and modern transcription of ‘Regnat’ from the manuscript F (166v V, Dittmer, 1966-7)	342
B.7	The first movement from <i>Five pieces for David Tudor</i> no.4 (piano, 1959) by Sylvano Bussotti	343
B.8	Bars 50-58 of ‘L’invitation au voyage’ (1870) by Henri Duparc, with annotations	344
B.9	A collection of notes annotated above with their pitch names, and below with MIDI note numbers (MNN) and morphetic pitch numbers (MPN)	352
B.10	Examples of scales, the cycle of fifths, and chords	361
B.11	Names for different degrees of the major and minor scales, and Roman numeral notation.	365
D.1	First occurrence of pattern <i>A</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	385
D.2	Second occurrence of pattern <i>A</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	386

D.3	First occurrence of pattern <i>B</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	387
D.4	Second occurrence of pattern <i>B</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	388
D.5	First occurrence of pattern <i>C</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	389
D.6	Second occurrence of pattern <i>C</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	390
D.7	First occurrence of pattern <i>D</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	391
D.8	Second occurrence of pattern <i>D</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	392
D.9	Occurrences of pattern <i>E</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	393
D.10	Occurrences of pattern <i>F</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	394
D.11	Occurrences of pattern <i>G</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	395
D.12	Occurrences of pattern <i>H</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	396
D.13	Occurrences of pattern <i>I</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	397
D.14	Occurrences of pattern <i>J</i> , discovered by SIACT in bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin	398
E.1	Mazurka section generated by the model Racchman-Oct2010. This excerpt corresponds to stimulus 19 in Chapter 10 (System A, p. 252)	400
E.2	Mazurka section generated by the model Racchman-Oct2010. This excerpt corresponds to stimulus 20 in Chapter 10 (System A, p. 252)	401
E.3	Mazurka section generated by the model Racchmaninof-Oct2010. This excerpt corresponds to stimulus 27 in Chapter 10 (System B, p. 253)	402
E.4	Mazurka section generated by the model Racchmaninof-Oct2010. This excerpt corresponds to stimulus 28 in Chapter 10 (System B*, p. 253)	403

1

Introduction

As with language, there was a time when music had no written or recorded formats. Music passed from one generation to another by singing and playing from memory, involving varying degrees of imagination and improvisation. One rarely stops to consider the merits and demerits of writing. The advantages of record keeping, education, and communication of news, opinions, and ideas are so great as to make the development of writing seem inevitable, and dwarf potential disadvantages. The merits and demerits of writing down music as symbols are more hotly debated: if I study a symbolic representation of a piece of music, am I in danger of neglecting the *music as heard*? This debate on music representation raises further questions that might otherwise be overlooked. How is an incoming stream of musical information organised by the ears and brain into percepts, and how do cognitive structures develop with the experience of music?

Music and mathematics—I am sometimes told—is an unusual combination. Historically, the study of music alongside mathematics was not at all unusual, both being part of a medieval university curriculum called the quadrivium (*c* 1300-1500), which comprised of arithmetic, geometry, astronomy, and music. René Descartes (1596-1650) was yet to introduce to

geometry the use of coordinates to plot points in two-, three-, etc. dimensional space. If the quadrivium and Cartesian coordinates, as they became known, had been contemporaneous, then a *geometric representation of music* in which notes appear as points in space would perhaps have been devised long before the late-twentieth century. Different *viewpoints* of a piece of music can also be represented as strings of symbols. Both geometric and string-based representations of music allow the application of mathematical concepts, *in principle*.

In principle, because until use of computers became widespread in the latter half of the twentieth century, there seemed little or no advantage to be gained from applying these mathematical concepts. Without computers, for example, one could discover repeated patterns in a piece of music by: (1) writing out the piece in a geometric representation; (2) identifying collections of points that occur twice or more (application of mathematical concepts). In a fraction of the time this would take, however, one could study the staff notation of the piece, perhaps play/sing it through or listen to a recording, and discover the same or very similar patterns. Similarly, without computers one could create a new piece of music by: (1) separating and writing out small portions from several existing pieces; (2) connecting previously unconnected portions according to certain probabilities (application of mathematical concepts). In a fraction of the time, however, a competent composer could play/sing through several existing pieces of music, allow their musical brain to undertake the separating and reconnecting activities, and so devise a similarly successful new piece.

Today, in contrast, researchers can program computers to discover repeated patterns in a piece of music and generate passages of music based on existing pieces, faster than humans undertaking the corresponding tasks, though typically not better than humans, as measured by appropriate methods. Arguments abound about whether the aim should be for computer programs to emulate human behaviour on musical tasks, or to extend human capabilities. Either way, it seems emulation of behaviour would be an informative precursor to extension of capability. As such, in this thesis the main motivation for investigating methods of pattern discovery in music is to shed some light on how ordinary listening and expert analysis work, and on how structure is induced by an incoming stream of musical information. In terms of applications, an improved method for pattern discovery might become part of a tool for music analysis. Perhaps it is optimistic to expect music analysts to welcome or employ such a discovery tool, especially if it requires learning a command-line language, but Huron (2001b) demonstrates effective use of a pattern matching tool in preparing a music-analytic essay.

A second application of a method for pattern discovery is in automated composition, where the patterns discovered in an existing piece become an abstract template for a new piece. A primary motivation for investigating computational approaches to stylistic composition is to shed light on *musical style* and the study of style. It seems that more computer models for stylistic composition have focused on generating or harmonising chorale melodies (hymn tunes) than on any other genre (Nierhaus, 2009). While chorales are an acceptable starting point, I have delved into A-level and university music syllabuses in order to unearth some alternative composition briefs.

A computational model that generates sections or entire pieces may lend an insight into musical style, but it is of no (honest) use to the estimated 50 000 students in England and Wales who respond to stylistic composition briefs each year.¹ Many of the models described here might be adapted (or were originally intended) to suggest a next melody note or chord, when given some preceding melody notes or chords and a database of music from the intended style. Developing a composition assistant based on such suggestions is beyond the scope of this thesis, but it is a motivating factor, and one that raises issues about the nature of human and computational creativity.

Chapters 2-5 of this thesis constitute the literature review. Chapters 6-10 contain original contributions (with Chapters 6, 7, and 10 describing evaluations). Chapter 11 is devoted to conclusions and future work.

Chapter 2 introduces geometric representations of music, which are the starting point for pattern discovery and music generation in later chapters. For readers curious about other music representations and definitions of terms such as *audio*, *pitch*, *staff notation*, *scale* etc., please see Appendix B. Calculation of probabilities and statistics in music is the subject of Chapter 3. The probability distributions constructed here are used in subsequent chapters to model aspects of music perception, necessitating a thorough account that begins with manipulating and counting vector representations of notes. An introduction to Markov models for music is given in Sec. 3.3, a topic that is revisited in Chapters 5, 8, and 9.

Chapter 4 begins with examples of five types of musical repetition, which are collected together and labelled the *proto-analytical class*. The task of

¹This figure is based on candidate numbers for the four main examination boards, and UCAS course registration statistics. The actual figure may be higher or lower, due to optional components in some syllabuses.

intra-opus discovery of translational patterns is introduced, and terms such as *pattern* are given mathematical definitions. Three algorithms from the SIA family (Meredith et al., 2003; Forth and Wiggins, 2009) are reviewed, as the patterns that these algorithms return are most consistent with the proto-analytical class. A short but important section considers how to determine when an improved method for pattern discovery has been achieved: two metrics, called *recall* and *precision* (Manning and Schütze, 1999), are defined. Chapter 5, the last in the literature review, focuses on computational modelling of musical style. Early models of musical style are discussed, as well as more recent models (Ebcioğlu, 1994; Conklin and Witten, 1995; Allan, 2002; Pearce, 2005). Cope's (1996, 2001, 2005) work on the databases and programs referred to collectively as Experiments in Musical Intelligence (EMI) has been particularly influential, so is reviewed in detail.

I will continue to map out the narrative of this thesis in relation to my research question and hypotheses:

How can current methods for pattern discovery in music be improved and integrated into an automated composition system?

This question can be broken down into two halves:

Question 1. How can current methods for pattern discovery in music be improved?

Question 2. How can methods for pattern discovery in music be integrated into an automated composition system?

Chapters 6 and 7 address question 1. Chapter 6 describes an experiment that attempts to answer a crucial question in the context of discovering re-

peated musical patterns: what makes a pattern important? Research that attempts to answer this question may have implications for improving the recall and precision of computational pattern discovery methods. Therefore, this question is addressed in Chapter 6, before trying to improve recall and precision values in Chapter 7. In Chapter 6, a variety of pre-existing and novel formulae for rating the perceptual salience of discovered patterns are discussed. An experiment is described in which music undergraduates rated already-discovered patterns, giving high ratings to patterns that they would prioritise mentioning in an analytical essay (cf. Sec. 2.4). My hypothesis is that a linear combination of existing and novel formulae for the perceptual salience of discovered patterns will offer a better explanation of the participants' ratings than any of the proposed formulae do individually.

In Chapter 7, I define and demonstrate the *problem of isolated membership*. My hypothesis is that the recall and precision values of certain computational methods for pattern discovery in music are adversely affected by the problem of isolated membership. A subsequent hypothesis is that the problem can be addressed by a method that I call *compactness trawling*. In contrast to the experiment reported in Chapter 6, where a group of participants was recruited to elicit ratings, the ground truth used in Chapter 7 was formed by a single expert.

Considering question 2 in more detail, the term *automated composition system* is rather broad. That is, suppose a program generates a fractal (a mathematical pattern, approximations of which can be found in nature, from clouds to cauliflowers) and then sonifies the fractal, converting it into rhythms and pitches (Sherlaw Johnson, 2003). This program could be classed as an

automated composition system, and one that integrates patterns into compositions. This is not what I mean, however, by integrating discovered patterns into an automated composition system. Almost immediately, the literature review of algorithmic composition (Chapter 5) focuses on stylistic composition. Renowned composers—and more often music students—undertake projects in stylistic composition. These projects range from devising an accompaniment to a given melody, to composing a full symphony in the style of another composer or period.

Markov chains, state spaces, and constraints for generating stylistic compositions are the subjects of Chapters 8 and 9. Two models are developed, each capable of generating the opening section of a piece in the style of pieces contained in a database—a database from which the models could be said to learn (Mitchell, 1997). My hypothesis is that a random generation Markov chain (RGMC) with appropriate state space and constraints is capable of generating passages of music that are judged as successful, relative to an intended style. The hypothesis suggests wide applicability of the model, in that the same type of state space and set of constraints might be used with equal success for different databases that contain music from different genres/periods. This aspect of the hypothesis is not tested, however, as the two models described in Chapter 9 (called Racchman-Oct2010 and Racchmaninof-Oct2010; acronyms explained in due course) are evaluated for one stylistic composition brief only: the generation of the opening section of a mazurka in the style of Frédéric Chopin (1810-1849).

A second hypothesis tested by the evaluation is that altering the RGMC to include *pattern inheritance* from a designated template piece leads to

higher judgements of stylistic success. It should be stressed that the patterns inherited from the template piece are of an abstract nature (not actual note collections). As described in Chapter 9, the temporal and registral positions of discovered repeated patterns from the template piece are used to guide the generation of a new passage of music. The difference between the models Racchman-Oct2010 and Racchmaninof-Oct2010 is that the latter includes pattern inheritance. A collection of mazurkas generated by another model is available for the purpose of comparison (Cope, 1997), and there is a well developed framework for evaluating models of musical style (Pearce and Wiggins, 2007; Amabile, 1996). The framework, called the Consensual Assessment Technique (CAT), is adopted (and adapted a little) in an experiment reported in Chapter 10. As well as demonstrating improved methods for pattern discovery in music, this dissertation contains the first full description and thorough evaluation of a model for generating passages in the style of Chopin mazurkas.

Chapter 11 contains concluding remarks on the improved methods for pattern discovery and their application in modelling musical style, as well as suggestions for future work. Two topics are considered in some detail: the precision and runtime of SIA (and hence the SIA family); and the adaptation of SIAC for audio summarisation.

Literature review: Music representations

Geometric representations of music are exploited in subsequent chapters, for pattern discovery and music generation. This review begins with *piano-roll notation*, which is a widely known geometric representation often used to display MIDI files. Next I discuss the generalised interval system (Lewin, 1987/2007) and viewpoints (Conklin and Witten, 1995), which are frameworks for representing aspects of staff notation (Sec. B.2.1) as sets (Def. A.7) and groups (Def. A.19). While the main definitions of this chapter, in Sec. 2.3, do not require an understanding of the generalised interval system, this system is included because it gives the theoretical explanation for why some viewpoints (e.g., ontime and MIDI note number) are more amenable than others (such as metric level and contour) to being treated as translational point-sets. Finally in this chapter, the topics of chord labelling, keyscapes, and automatic transcription are introduced. These topics are revisited in Secs. 4.4, 10.5.5, and 11.2.1 respectively.

2.1 Musical instrument digital interface

Musical instrument digital interface (MIDI) is a means by which an electronic instrument (such as a synthesiser, electronic piano, drum kit, even

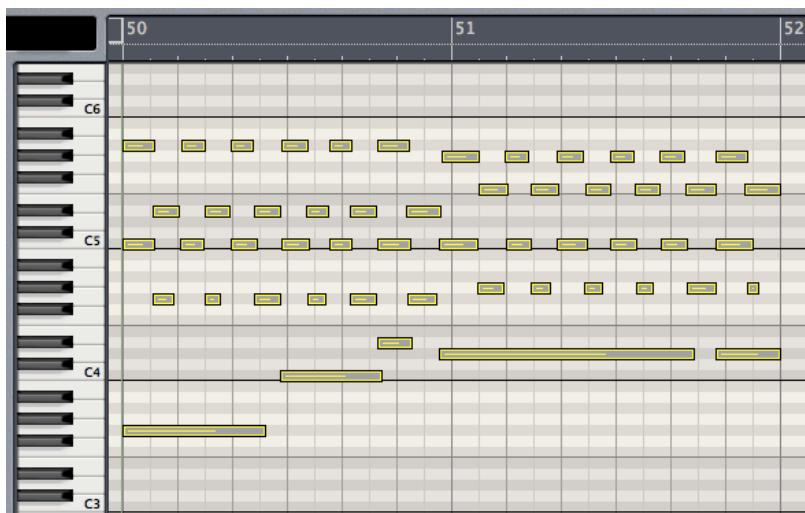
a customised guitar, etc.) can connect to a computer and hence communicate with music software (Roads, 1996). If a performer presses a key on a MIDI-enabled instrument, a message is sent to the computer, registering that a certain key has been pressed, and how hard it was pressed. The time of this so-called *note-on event* is also registered. When the performer releases the key, another message is sent and a note-off event registered. The duration of the performed event can be obtained by subtracting the time of the note-on event from that of the note-off event. Even if the performer is playing from staff notation and attempting to adhere to the reference beat of a metronome, it is difficult to recover an accurate staff-notation representation from the MIDI events, due to expressive and/or unintended timing alterations (Desain and Honing, 1989). The difference between MIDI events that are performed and MIDI events that are accurate with regards staff notation is illustrated by Figs. 2.1A and 2.1B respectively. (These figures contain versions of the piano part from bars 50 and 51 of Fig. B.8.) It can be seen from Fig. 2.1A that the performer's tempo accelerates during bar 50, and they arrive early with the three-note chord that begins bar 51. Further, some notes appear longer or shorter than others, even though they are scored with the same duration. In Fig. 2.1B, on the other hand, ontimes and durations are exactly as they appear in the score. The representation of music shown in Fig. 2.1, of MIDI note number plotted against ontime, with duration indicated by the length of a line segment, is known as *piano-roll notation*, named after mechanically-operated pianos that use rolls of material marked in such a way. The process of transforming the information in Fig. 2.1A into the information in Fig. 2.1B is called *quantisation*.

Given the difficulties of quantisation, a database of music in MIDI format tends not to be a reliable source for obtaining a symbolic representation of a piece that is accurate with respect to staff notation. MIDI was not designed with this example of use in mind, however. It is a means of communicating with music software in real time, and applications include a program that listens to a performer’s MIDI input (perhaps improvised) and responds with its own improvisations (Pachet, 2002).

In a note-on event, pitch is represented by an integer called *note number*, or MIDI note number (MNN). It is not possible to determine from a note-on event whether, say, MNN 60 corresponds to pitch B♯3, C4, or D♭4. That is, there is not a one-to-one correspondence between pitch and MNN. Meredith (2006a) defines morphetic pitch number (MPN) as the height of a note on the staff. The MPN of B♯3 is 59, the MPN of C4 is 60, and the MPN of D♭4 is 61 (see also Fig. B.9).¹ Hence, if the pitch of a note (cf. Def. B.6) is known, its MNN and MPN can be determined. Vice versa, if the MNN and MPN of a note are known, then its pitch can be determined. Without going into more detail, there is a bijection between pitch and pairs of MIDI note and morphetic pitch numbers. There is also a bijection between each pitch class and $(x_1, x_2) \in (\mathbb{Z}_{12} \times \mathbb{Z}_7)$, where \mathbb{Z}_{12} is the set of MIDI note numbers modulo 12, and \mathbb{Z}_7 is the set of morphetic pitch numbers modulo 7.

¹This numbering is different (shifted) to that of Meredith (2006a), because I find counting easier when C4 (or middle C) has MNN 60 and MPN 60.

A



B

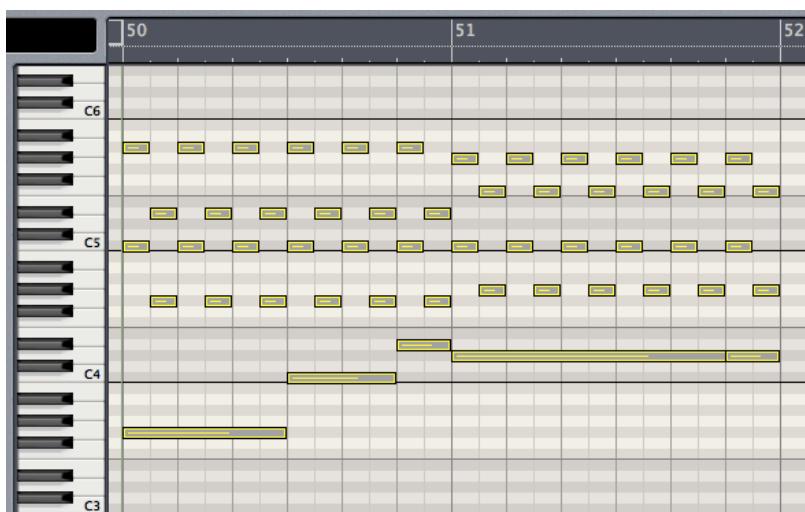


Figure 2.1: These figures contain versions of the piano part from bars 50 and 51 of Fig. B.8. The representation of music—of MIDI note number plotted against onttime, with duration indicated by the length of a line segment—is known as piano-roll notation. (A) The piano-roll notation, as it was performed; (B) Each MIDI event has been quantised so that it is accurate with respect to the corresponding staff notation.

2.2 The generalised interval system and viewpoints

Definitions and Examples A.17-A.22 in Appendix A introduce the concept of a *group acting on a set*. What is the relevance of this concept to music? Lewin (1987/2007) suggests that certain sets of musical elements are acted on by certain groups of musical intervals. For instance, let $\Omega = \mathbb{Z}$ represent MIDI note numbers, and $G = \mathbb{Z}$ represent semitone intervals. The combination of two semitone intervals produces a third semitone interval (for instance, 4 semitone intervals followed by 3 semitone intervals gives an interval of 7 semitones). The group $(G, +)$ of semitone intervals acts on the set Ω of MIDI note numbers. The manner in which a semitone interval is capable of transforming a MIDI note number is analogous to the manner in which, say, a rotation is capable of transforming a cube vertex. Although aspects of music had been represented numerically by Simon and Sumner (1968) and Longuet-Higgins (1976), it was Lewin (1987/2007) who recognised the group action of musical intervals on musical elements, and defined this as a *generalised interval system*. The last sentence of Example A.22 is equivalent to Lewin's (1987/2007) definition of a generalised interval system. The clear distinction between musical elements and musical intervals has laid the foundation for a number of practical extensions (Wiggins, Harris, and Smaill, 1989; Conklin and Witten, 1995; Meredith, Lemström, and Wiggins, 2002), as well as theoretical musings (Ockelford, 2005; Tymoczko, 2011).

Plausible musical sets and groups include:

Ontime and inter-ontime interval. The set $\Omega_1 = \mathbb{R}$ of ontimes is acted

on by the group ($G_1 = \mathbb{R}, +$) of time intervals, also called inter-onset intervals (IOI or loi).

MIDI note number and semitone interval. These have been discussed already. I will use $\Omega_2 = \mathbb{Z}$ to denote the set of MIDI note numbers, and ($G_2 = \mathbb{Z}, +$) to denote the group of semitone intervals (int).

Morphetic pitch number and staff step. Likewise, the set $\Omega_3 = \mathbb{Z}$ of morphetic pitch numbers is acted on by the group ($G_3 = \mathbb{Z}, +$) of steps on the staff (or staff steps).

Duration and duration ratio. The set $\Omega_4 = \mathbb{R}_+^*$ of durations is acted on by the group ($G_4 = \mathbb{R}_+^*, \times$) of duration ratios (dr). Lewin (1987/2007) suggests that durations can also be thought of as an additive group, that is $\Omega_4 = \mathbb{R}$, and ($G_4 = \mathbb{R}, +$) is the group. Vice versa, onsets can be thought of as a multiplicative group.

Staff. The set $\Omega_5 = \mathbb{N} \cup \{0\}$ is used to index staves from the top to the bottom of a system. Although it is possible to determine whether two staves $\omega_1, \omega_2 \in \Omega_5$ are the same, it does not make sense for a staff to be transformed or acted on, so there is no associated group action.

Metric level 1. In the set $\Omega_6 = \{t, \perp\}$, where t is a special symbol for ‘true’, and \perp for ‘false’, the element t represents onsets that coincide with the first beat of the bar, and \perp represents onsets that do not. Again, there is no associated group action. Metric level 1 is abbreviated as ml1.

Leap. In the set $\Omega_7 = \{t, \perp\}$, the element t represents the absolute difference

between two morphetic pitch numbers being greater than one, and \perp represents the absolute difference being less than or equal to one. No associated group action.

Contour. In the set $\Omega_8 = \{-1, 0, 1\}$, the element -1 represents the second of two MIDI note numbers being less than the first, 0 represents the two MIDI note numbers being the same, and 1 represents the second of the two MIDI note numbers being greater than the first. No associated group action.

Definition 2.1. Viewpoints (Conklin and Witten, 1995) build on—and in some cases are—generalised interval systems (Lewin, 1987/2007). A *primitive viewpoint* is a set that can be defined from the *musical surface* (Lerdahl and Jackendoff, 1983). For a function $f : A_1 \times A_2 \times \cdots \times A_m \rightarrow B$, where A_1, A_2, \dots, A_m are primitive viewpoints, the set B is called a *derived viewpoint*. A viewpoint may or may not satisfy the properties of a group (cf. Def. A.19). ■

Each of ontime, inter-ontime interval, MIDI note number (MNN), semi-tone interval, morphetic pitch number (MPN), staff step, duration, duration ratio, staff, metric level 1, leap, and contour is a viewpoint. Onetime, MNN, MPN, duration, staff, and metric level 1 are primitive viewpoints, whereas the others are derived. If some notes from a piece are labelled $1, 2, \dots, n$, and A is a viewpoint, then the list $L_A = (a_1, a_2, \dots, a_n)$, where each $a_i \in A$, is a representation of the notes from a particular viewpoint. For $n = 18$ notes extracted from Fig. B.8, fifteen such viewpoint lists (six primitive and nine derived) are given in Fig. 2.2. This figure was prepared by choosing from the viewpoints considered by Conklin and Witten (1995) and Con-

klin and Bergeron (2008). The former paper's focus was chorale melodies (hymn tunes), and the latter's focus was melodies by the singer-songwriter Georges Brassens (1921-1981). Many of the viewpoints for which viewpoint lists appear in Fig. 2.2 were introduced above, so an exhaustive explanation will not be provided. For the viewpoint A of MNN, the viewpoint list $L_A = (72, 72, 75, \dots, 67)$ can be seen in Fig. 2.2. The viewpoints MPN and staff step do not appear. The derived viewpoint called pc in Fig. 2.2 is MIDI note number modulo 12. The first element of all other derived viewpoint lists shown is 'false', or \perp , because these derived viewpoints rely on two elements of one (or more) primitive viewpoint list (usually elements i and $i - 1$, where $i = 2, 3, \dots, n$) in order to be well defined.

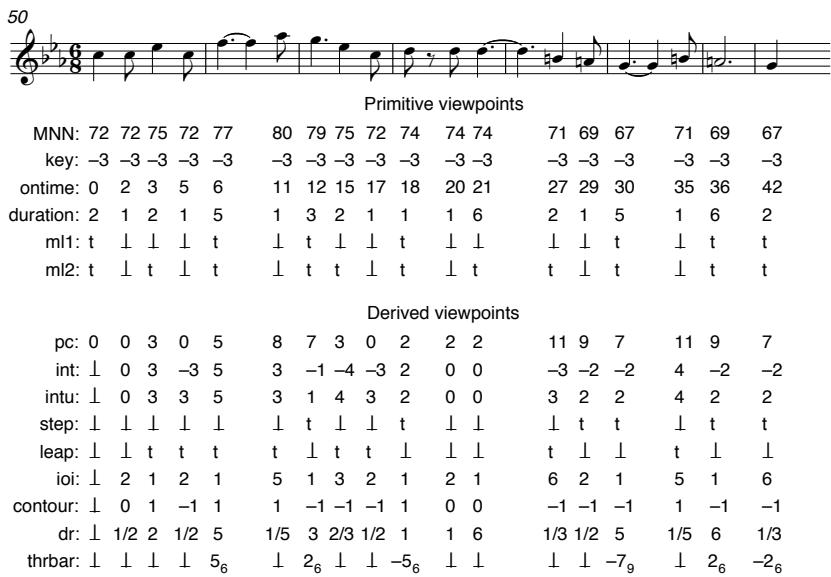


Figure 2.2: Fifteen viewpoint lists (six primitive and nine derived) for $n = 18$ notes extracted from Fig. B.8. This figure was prepared by choosing from the viewpoints considered by Conklin and Witten (1995) and Conklin and Bergeron (2008).

The viewpoints considered musically relevant vary from one genre to another, e.g. from chorale melodies (Conklin and Witten, 1995) to Brassens' melodies (Conklin and Bergeron, 2008). When defining viewpoints, music-theoretic relevance is more important than adherence to the generalised interval system. For instance, the viewpoint contour makes musical sense, but does not give rise to a generalised interval system. It is not associated with a group, whereas a viewpoint such as ontime is associated with the group of inter-ontime intervals.

Determining which notes of a piece to use to form viewpoint lists is straightforward for monophonic pieces, but can be difficult for polyphonic pieces. In a monophonic piece, all notes can be used to form viewpoint lists, in ascending order of ontime. In a polyphonic piece, many perceptually valid successions of notes can be used to form viewpoint lists, as demonstrated by the *directed graph* or *digraph* (Wilson, 1996) in Fig. 2.3. Notes are represented by *vertices*, labelled v_1, v_2, \dots, v_n . An *arc*, written $v_i v_j$, is indicated by an arrow from vertex i to vertex j . Plausible and implausible successions of notes that might be used to form viewpoint lists are represented by *walks*, a walk being a list of arcs, $L = (v_{i_1} v_{i_2}, v_{i_2} v_{i_3}, \dots, v_{i_{m-1}} v_{i_m})$. Some of the walks that I think correspond to perceptually valid succession of notes are shown in black in Fig. 2.3, and one implausible walk is shown in red. If it were possible to determine reliably the top N most perceptually valid successions of notes, given, say, the notes' ontimes, pitches, and durations as input, then the application of viewpoints to polyphony could be widened beyond specific genres and specific retrieval tasks (Conklin, 2002; Bergeron and Conklin, 2008).

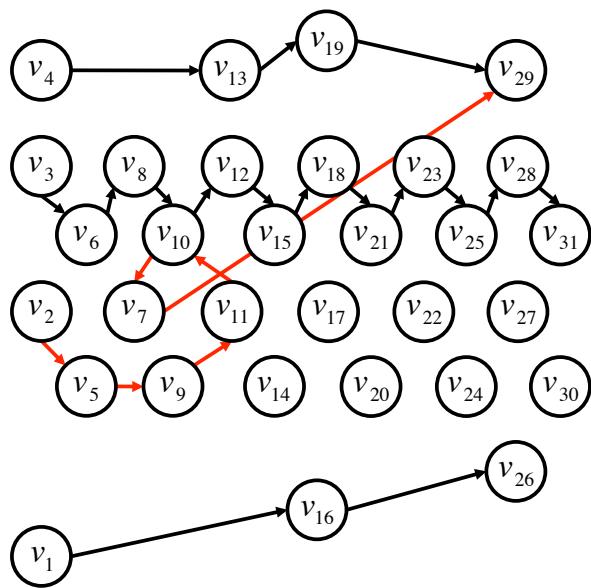


Figure 2.3: A digraph (Wilson, 1996) representing notes (vertices, labelled v_1, v_2, \dots, v_n) from an excerpt of music. An arc, written $v_i v_j$, is indicated by an arrow from vertex i to vertex j . Plausible and implausible successions of notes that might be used to form viewpoint lists are represented by walks, a walk being a list of arcs, $L = (v_{i_1} v_{i_2}, v_{i_2} v_{i_3}, \dots, v_{i_{m-1}} v_{i_m})$.

2.3 Geometric representations

One instance of a geometric representation of music has been given already: piano-roll notation (Fig. 2.1), where the MNN of a note is plotted against its ontime, and a line segment indicates its duration.

Definition 2.2. Dataset and projection. Meredith et al. (2002) define a *dataset* D to be a nonempty, finite subset of vectors in \mathbb{R}^k , written

$$D = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n\}. \quad (2.1)$$

When representing a piece of music as a dataset, Meredith et al. (2002) suggest considering the ontime, MIDI note number (MNN), morphetic pitch number (MPN), duration, and staff of each note. Retaining the notation from Sec. 2.2, an arbitrary element $\mathbf{d} \in D$ is called a *datapoint* and given by

$$\mathbf{d} = (\omega_1, \omega_2, \omega_3, \omega_4, \omega_5), \quad (2.2)$$

where $\omega_1 \in \Omega_1$ is the ontime of the datapoint, $\omega_2 \in \Omega_2$ is its MNN, $\omega_3 \in \Omega_3$ is its MPN, $\omega_4 \in \Omega_4 = \mathbb{R}$ its duration, and $\omega_5 \in \Omega_5$ its staff. The sets $\Omega_1, \Omega_2, \dots, \Omega_5$ are called the *dimensions* of the dataset.

For certain purposes, it is helpful to consider fewer than all five of the dimensions above, e.g. ontime, MNN, and staff. Informally, it is said that the dataset D is projected on to the dimensions of ontime, MNN, and staff, giving a new dataset E with an arbitrary datapoint $\mathbf{e} = (\omega_1, \omega_2, \omega_5)$. Formally, $\mathbf{e} = (\omega_1, \omega_2, 0, 0, \omega_5)$ should be written rather than $\mathbf{e} = (\omega_1, \omega_2, \omega_5)$. This is because a *projection* is a function $f : D \rightarrow D$ such that $f^2(\mathbf{d}) = f(f(\mathbf{d})) = f(\mathbf{d})$. ■

Another example of a projection is the function $g : \mathbb{Z} \rightarrow \mathbb{Z}_{12}$ that maps MIDI note numbers to MIDI note numbers modulo 12. For instance, $g(59) = 11$, and $g(11) = 11$, so $g^2(59) = g(g(59)) = g(11) = 11$.

Appendix B contains an excerpt (Fig. B.8) by Henri Duparc (1848-1933). The dataset for the excerpt is

$$D = \{(0, 55, 57, 3, 2), (0, 72, 67, \frac{1}{2}, 1), (0, 72, 67, 2, 0), (0, 81, 72, \frac{1}{2}, 1), (\frac{1}{2}, 67, 64, \frac{1}{2}, 1), (\frac{1}{2}, 75, 69, \frac{1}{2}, 1), \dots, (56\frac{2}{3}, 74, 68, \frac{1}{3}, 1)\}. \quad (2.3)$$

Two projections of the dataset corresponding to bars 50-53 are plotted in Fig. 2.4. Figure 2.4A is a plot of MPN against ontime, and Figure 2.4B is a plot of duration against ontime.

Definition 2.3. Lexicographic order. Let $\mathbf{d} = (d_1, d_2, \dots, d_k)$ and $\mathbf{e} = (e_1, e_2, \dots, e_k)$ be arbitrary and distinct members of a dataset D . There will be a minimum integer $1 \leq i \leq k$ such that $d_i \neq e_i$. (Otherwise $\mathbf{d} = \mathbf{e}$, which contradicts \mathbf{d} and \mathbf{e} being distinct.) If $d_i < e_i$ then \mathbf{d} is said to be *lexicographically less than* \mathbf{e} , written $\mathbf{d} \prec \mathbf{e}$. Otherwise $e_i < d_i$, so $\mathbf{e} \prec \mathbf{d}$ (Meredith et al., 2002).

A dataset is a set, so it is unordered. It is possible to order the elements of a dataset, however, lexicographically or according to some other rule. A set whose elements can be ordered in this way is called a *totally ordered set*, although I will not make a notational distinction. Throughout subsequent chapters, it will be assumed that a dataset is in lexicographic order, unless stated otherwise. ■

The dataset D in (2.3) is in ascending *lexicographic order*. For instance, $\mathbf{d} = (0, 72, 67, \frac{1}{2}, 1) \prec \mathbf{e} = (0, 72, 67, 2, 0)$. For $j = 1, 2, 3$, $d_j = e_j$. And then

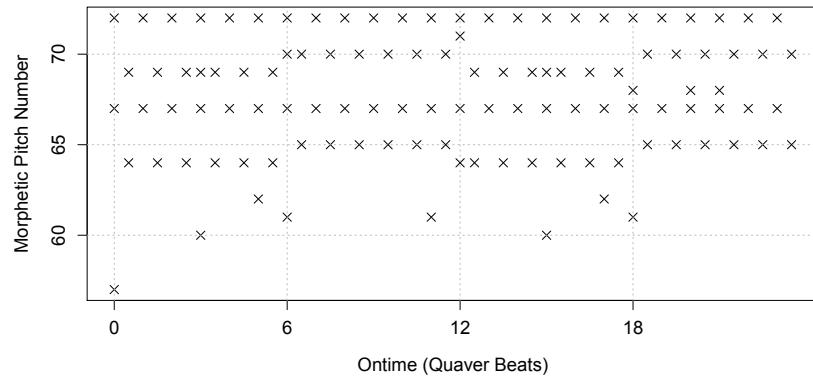
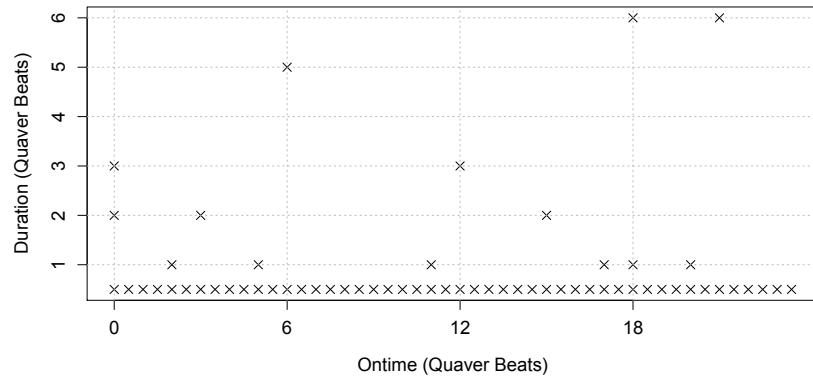
A**B**

Figure 2.4: Plots of two dataset projections corresponding to bars 50-53 for the excerpt by Duparc shown in Fig. B.8. (A) A plot of morphetic pitch number against ontine (measured in quaver beats); (B) A plot of duration against ontine (both measured in quaver beats).

$d_4 = \frac{1}{2} < 2 = e_4$, meaning $\mathbf{d} \prec \mathbf{e}$.

The term *geometric* is used for the representation of music as a dataset because a datapoint can be thought of as a point in Euclidean space. A geometric representation is the main music representation used in subsequent chapters. As mentioned (in Appendix B, pp. 328 and 352), there is the notion of a representation that contains the minimum amount of information necessary for a listener to be able to recognise a familiar piece. Datasets are closer to this minimum category than musicXML or kern, say (cf. Sec. B.2.3). Datasets (and viewpoints) employed to date tend to overlook what Fallows (2001) calls the ‘most consistently ignored components of a musical score’ (p. 271), which has its advantages—in terms of the mathematical manipulation that becomes possible—and disadvantages—in terms of the vital information that could be missed: perhaps some notes in an inner voice are marked staccato (played with shorter durations) and so assume greater perceptual salience relative to surrounding voices, but this is missed because articulation marks are overlooked.

2.4 Some music-analytical tools

For Bent and Pople (2001), music analysis ‘may be said to include the interpretation of structures in music, together with their resolution into relatively simpler constituent elements, and the investigation of the relevant functions of those elements’ (p. 526). In the quotation, the meanings of element and function are different to their mathematical definitions (Appendix A). Here is an example of a music-analytic task:

Essay question. ‘Write a detailed analysis of *either* the Prelude *or* the

Fugue' (Cambridge University Faculty of Music, 2010a, p. 20). 'You are provided with a score of the Prelude and Fugue, which you may annotate in pencil if you wish' (*ibid.*).

Knowledge of the elements of staff notation (cf. Sec. B.2.2) is a prerequisite for answering the above question, but this knowledge alone is insufficient. There are commonly agreed terms for collections and successions of notes that abstract from the musical surface, and afford concision to the music analyst (cf. Sec. B.3). If one reads an issue of the journal *Music Analysis* or an introduction to the discipline (Cook, 1987), it is evident that music analysis is as much about developing and criticising concepts and tools as it is about using them as the basis for writing essays on particular pieces. The body of concepts and tools can be thought of as *music theory*, with different parts being more or less widely understood, accepted, and applied by different analysts.

Due to the potential for ambiguity, defining an *algorithm* that undertakes a music-analytical task such as chord labelling (Pardo and Birmingham, 2002) or key finding (Sapp, 2005) is a challenge. An algorithm can be defined (loosely) as a computational procedure taking some values as input and producing some values as output (Cormen, 2001). An introduction to algorithms as applied to bioinformatics is given by Jones and Pevzner (2004), addressing a variety of tasks/problems (some have parallels in music analysis), algorithmic design techniques, and matters such as correctness and complexity (big-O notation). Pardo and Birmingham (2002) give a clear description and thorough evaluation of the HarmAn algorithm for chord labelling. It takes a dataset representation D of a piece/excerpt as input,

where D is projected on to ontime, MNN, and duration. HarmAn produces another dataset E as output, with the first dimension of E being the ontime of a chord label, the second being the MNN modulo 12 of the chord root, the third being an integer between 0 and 5 indicating the chord class (cf. Def. B.10), and the fourth being the duration for which the label is valid.

The output of HarmAn when applied to bars 50-57 of ‘L’invitation au voyage’ by Duparc is shown in Fig. 2.5. The pair (A, 4) denotes that the root of the area labelled is pitch class A (I have converted MNN modulo 12 to pitch class), and that the class is 4, a half diminished 7th chord (Def. B.10).

HarmAn produces an acceptable labelling of the excerpt, although in general, the algorithm could be improved by: (1) including a class for the minor 7th chord, which has semitone intervals (3, 4, 3), an instance of which occurs in the first half of bar 55; (2) preventing dominant 7th labels encroaching on foregoing major triad areas. For example, in the annotation for bars 57-58, there is a G dominant 7th label, but a better labelling would be G major triad for bar 57, and G dominant 7th for bar 58. So at present, the dominant 7th label encroaches on a foregoing major triad area.

Sapp’s (2005) key finding algorithm compares the relative weight of each MNN in different windows of the dataset to known major and minor key profiles (Aarden, 2003). It selects the best fitting key profile for each window. Chapter 3 addresses the calculation of probabilities and statistics in music, which relates to the mechanism of the key finding algorithm. A colour-coded plot is given in Fig. 2.6 for the output of the key finding algorithm when applied to bars 50-57 of ‘L’invitation au voyage’ by Duparc. (Again, I have converted from MNN modulo 12 to pitch class.) The yellow box in Fig. 2.6

[Presque lent]

50 *più f* *cresc. molto*

S. C'est pouras - sou - vir, Ton moin - dre dé - sir Qu'ils vien -

Pno. *più f*

(A, 4) (D, 4) (A, 4) (D, 1)

54 *f* *dim.*

nent du bout. du mon - - -

f (G, 0) (B, 4)(G, 0) (E, 4) (D, 1)

57 *mf*

de Les so -

(G, 1) (B, 4)

Figure 2.5: The output of HarmAn (Pardo and Birmingham, 2002) when applied to bars 50-57 of ‘L’invitation au voyage’ by Duparc. The pair (A, 4) denotes that the root of the area labelled is pitch class A (I have converted MNN modulo 12 to pitch class), and that the class is 4, a half diminished 7th chord.

represents the key of a dataset window centred at ontime 9 (quavers). The green box to the left represents the key of a dataset window centred at ontime 6. Each box on this row of the plot represents a window that has length 6 (quavers) in the dataset; on the next row, length 9; on the next row, length 12. The very top box represents a window that has length 57 in the dataset, so this box represents the key of the whole excerpt. Sapp (2005) suggests that these so-called *keyscape plots* ‘display the harmonic [chordal] structure of the music in a hierarchical manner’ (p. 14).

This chapter began with a review of music representations. It has ended with a handful of examples of how algorithms have been defined to analyse and abstract from the musical surface of a piece, and how such abstractions may be represented. Further abstract representations, e.g. trees and more digraphs (cf. Fig. 2.3), will appear in due course. Having kept descriptions of audio and symbolic representations of music separate (in Appendix B), I will conclude this chapter by highlighting one way in which the two representations are being merged. Klapuri and Davy (2006) give an overview of attempts to define an algorithm that takes an audio signal as input and produces a symbolic representation, such as a MIDI file, as output. This class of algorithms is referred to as *automatic transcription algorithms*. I tried the portion of audio shown in Fig. B.1 (from ‘To the end’ by Blur, 1994) as the input to an automatic transcription algorithm. For this input, the automatic transcription algorithm available with a program called Melodyne produced the output shown in Fig. 2.7. It can be seen that one signal (Fig. B.1) has been separated into many smaller component signals (Fig. 2.7). These smaller components have also been summarised as under-

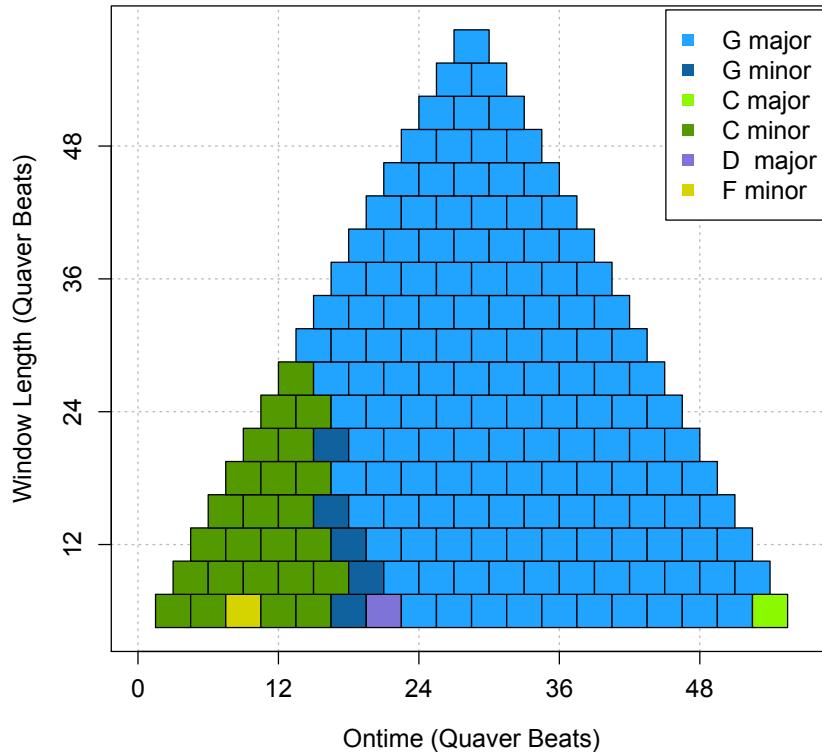


Figure 2.6: A colour-coded plot called a keyscape for the output of a key finding algorithm (Sapp, 2005) when applied to bars 50-57 of ‘L’invitation au voyage’ by Duparc. I have converted MNN modulo 12 to pitch class. The yellow box represents the key of a dataset window centred at ontione 9 (quavers). The green box to the left represents the key of a dataset window centred at ontione 6. Each box on this row of the plot represents a window that has length 6 (quavers) in the dataset; on the next row, length 9; on the next row, length 12. The very top box represents a window that has length 57 in the dataset, so this box represents the key of the whole excerpt.

lying rectangles, which closely resemble—and in fact, define—MIDI events (cf. Fig. 2.1).

**FIGURE REMOVED
FROM ELECTRONIC VERSION
FOR COPYRIGHT REASONS**

Figure 2.7: The output of an automatic transcription algorithm available with a program called Melodyne, as applied to the portion of audio from ‘To the end’ by Blur (1994) shown in Fig. B.1. The output is plotted as transcribed pitch height against time. The signal has been separated into smaller component signals, and the underlying rectangles suggest parallels between this figure and MIDI events (Fig. 2.1). © Copyright 1994 by EMI Music Publishing. Used by permission.

Describing the mathematics of an automatic transcription algorithm is beyond the scope of this dissertation, but suffice it to say that polyphonic audio-MIDI transcription is an open (and difficult) problem in music information retrieval (MIR). The transcription in Fig. 2.7, for example, places the first beat of the bar differently (and incorrectly) compared to the transcription in Fig. B.2. As automatic transcription algorithms improve and the number of such mistakes fall, it will be interesting to investigate applying symbolically-conceived algorithms (e.g. for pattern discovery or composition) to automatically-transcribed audio.

3

Literature review: Calculating probabilities and statistics in music

Models of music perception often involve calculating probabilities, so to give a thorough review of such models, an explanation of empirical distributions is necessary. The work on rating pattern importance in Chapter 6 relies in part on probabilistic calculations, as does one of the constraints in Chapter 9 for guiding music generation. This chapter begins with examples of how to calculate probabilities from dataset representations of music. Recall from Def. 2.2 that a dataset D consists of datapoints, which are vectors containing the ontime, MNN, MPN, duration, and staff of notes. The first step in this process is to form a list D^* in which the ontime (and perhaps other dimensions) of each datapoint has been removed, making it possible to count occurrences. The present chapter goes on to address how models of music perception (Sec. 3.2) and models of musical style (Sec. 3.3) can be built from calculated probabilities.

3.1 The empirical probability mass function

Definition 3.1. Empirical probability mass function. When defining an empirical probability mass function, first, there must be a list $D^* =$

$(\mathbf{d}_1^*, \mathbf{d}_2^*, \dots, \mathbf{d}_n^*)$ with each element $\mathbf{d}_i^* \in \mathbb{R}^k$, $i = 1, 2, \dots, n$. Second, repeated elements are removed from this list to form the ordered set $D' = \{\mathbf{d}'_1, \mathbf{d}'_2, \dots, \mathbf{d}'_{n'}\}$. Third, each $\mathbf{d}'_i \in D'$ has a relative frequency of occurrence in the list D^* , which is recorded in the probability vector $\pi = (\pi_1, \pi_2, \dots, \pi_{n'})$. If X is a discrete random variable that assumes the vector value $\mathbf{d}'_i \in D'$ with probability $\pi_i \in \pi$, $i = 1, 2, \dots, n$, then it is said that X has the *empirical probability mass function* specified by D' and π , or X has the distribution given by D' and π . ■

Example 3.2. An excerpt by Charles Ives (1875-1954) is shown in Fig. 3.1. The dataset for this excerpt (in lexicographic order) is

$$\begin{aligned} D = & \{(0, 43, 50, 12, 4), (0, 59, 59, 18, 3), (0, 74, 68, 18, 2), (0, 91, 78, 12, 1), \\ & (12, 47, 52, 6, 4), (12, 90, 77, 6, 1), (18, 47, 52, 6, 4), (18, 67, 64, 5, 3), \\ & (18, 67, 64, 6, 2), (18, 88, 76, 4, 1), (22, 86, 75, 2, 1), (23, 65, 63, 1, 3), \\ & (24, 48, 53, 2, 4), (24, 64, 62, 4, 3), \dots, (75, 65, 63, 1, 3)\}. \end{aligned} \tag{3.1}$$

To find, say, the empirical probability mass function of the MIDI note numbers (MNN) appearing in this excerpt, first a list is formed consisting of the MNN of each datapoint. These are the second elements of each vector in D :

$$\begin{aligned} D^* = & (43, 59, 74, 91, 47, 90, 47, 67, 67, 88, 86, 65, 48, 64, 74, 84, 43, 76, \\ & 36, 67, 48, 47, 86, 45, 69, 84, 57, 59, 60, 62, 64, 62, 60, 47, 59, 74, \quad (3.2) \\ & 91, 48, 47, 45, 43, 70, 61, 64, 47, 75, 90, 70, 47, 67, 67, 88, 86, 65). \end{aligned}$$

Each $\mathbf{d}_i^* \in D^*$ is an element of \mathbb{R}^1 , $i = 1, 2, \dots, n$. Second, repeated elements

are removed from the list in (3.2) to form the ordered set

$$D' = \{36, 43, 45, 47, 48, 57, 59, 60, 61, 62, 64, 65, 67, 69, 70, 74, 75, 76, 84, 86, 88, 90, 91\}. \quad (3.3)$$

Third, each $\mathbf{d}'_i \in D'$ has a relative frequency of occurrence in the list D^* , which is recorded in the probability vector

$$\pi = \left(\frac{1}{54}, \frac{1}{18}, \frac{1}{27}, \frac{7}{54}, \frac{1}{18}, \frac{1}{54}, \frac{1}{18}, \frac{1}{27}, \frac{1}{54}, \frac{1}{27}, \frac{1}{18}, \frac{1}{27}, \frac{5}{54}, \frac{1}{54}, \frac{1}{27}, \frac{1}{18}, \frac{1}{54}, \frac{1}{54}, \frac{1}{27}, \frac{1}{18}, \frac{1}{27}, \frac{1}{27} \right). \quad (3.4)$$

The discrete random variable X can be defined, with the distribution given by D' in (3.3) and π in (3.4). For instance, $\mathbb{P}(X = 36) = \frac{1}{54}$, $\mathbb{P}(X = 43) = \frac{1}{18}$, etc. The empirical probability mass function for X is plotted in Fig. 3.2. ■

It is vital to appreciate that any combination of dimensions (or indeed, viewpoints) can be used to define empirical distributions. For instance, a list D^* can be formed, consisting of the MNN modulo 12 and duration of each datapoint $\mathbf{d} \in D$ from (3.1).¹ The MNN modulo 12 is the remainder term when the MNN is divided by twelve (cf. Def. A.18), while the duration is the fourth element in each vector. Thus

$$D^* = ((7, 12), (11, 18), (2, 18), (7, 12), (11, 6), (6, 6), (11, 6), (7, 5), (7, 6), (4, 4), (2, 2), (5, 1), (0, 2), (4, 4), \dots, (5, 1)). \quad (3.5)$$

As in obtaining (3.3) from (3.2), it would be possible to remove repeats from this list to form the ordered set D' , and then to form the probability

¹Modulo 12 is relevant because of octave equivalence (cf. Def. B.5), and there being twelve semitones in an octave (cf. Def. B.6).

**FIGURE
REMOVED
FROM
ELECTRONIC
VERSION
FOR
COPYRIGHT
REASONS**

Figure 3.1: Bars 1-19 of *The unanswered question* (c 1930-1935) by Charles Ives. © Copyright 1953 by Southern Music Publishing. Reprinted by permission of Faber Music Ltd., London.

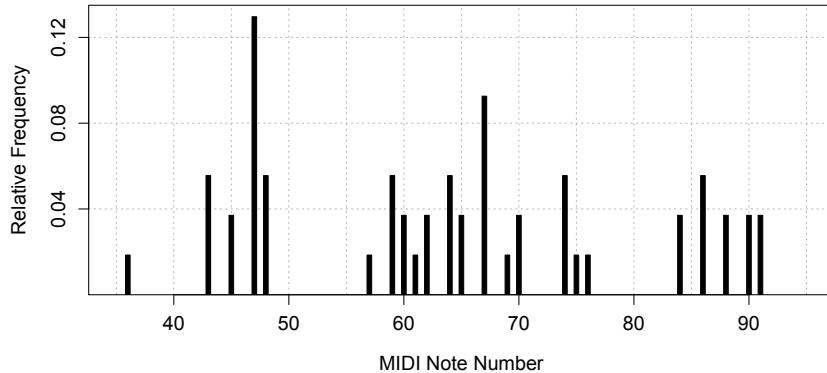


Figure 3.2: The empirical probability mass function for MIDI note numbers (MNN) in bars 1-19 of *The unanswered question* by Ives. The probability vector (3.4) is plotted against the corresponding MNNs (3.3).

vector π containing the corresponding relative frequencies of occurrence. The empirical probability mass function arising from D^* in (3.5) is plotted in Fig. 3.3, starting from the tonic, G. Masses (coloured rectangles) are collected together in this plot according to MNN modulo 12. The left-most, blue collection of rectangles represent the mass associated with $7 \in \mathbb{Z}_{12}$, where \mathbb{Z}_{12} is the set of MNN modulo 12. The element $7 \in \mathbb{Z}_{12}$ is associated with pitch-class G in the excerpt from Fig. 3.1. In Fig. 3.3, some of the largest masses are labelled with their corresponding durations. For instance, the MNN modulo 12 and duration pair $(7, 12)$ is labelled among the collection of blue rectangles. Considering the largest masses in Fig. 3.3, it seems that pitch classes C and D ($0, 2 \in \mathbb{Z}_{12}$) from Fig. 3.1 tend to be associated with relatively short durations (crotchet and minim), whereas pitch classes G and B ($7, 11 \in \mathbb{Z}_{12}$) are associated with a range of durations. This insight is enabled by considering an empirical distribution with *two* dimensions.

Defining an empirical distribution with several dimensions/viewpoints is one way of gaining insight into a piece/excerpt; another way is to take two

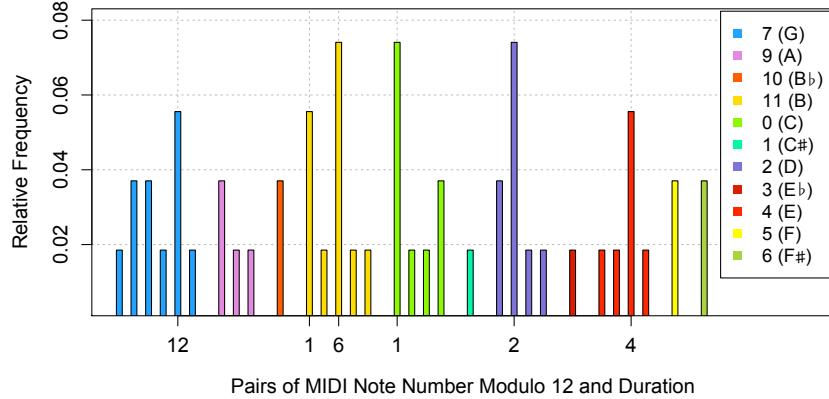


Figure 3.3: The empirical probability mass function for pairs of MMN modulo 12 and duration in bars 1-19 of *The unanswered question* by Ives. Pairs of MMN modulo 12 and duration are collected together in this plot according to MMN modulo 12. Within a collection, masses associated with short durations are to the left; masses associated with longer durations to the right.

or more dimensions/viewpoints and fuse them into a one-dimensional distribution. For instance, consider again the dimensions of MMN modulo 12 and duration, but rather than forming a two-dimensional empirical distribution as above, try fusing them. Weight the mass attributed to $\omega_i \in \mathbb{Z}_{12}$, a MMN modulo 12, by the total duration $\psi_i \in \mathbb{R}$ in crotchet beats for which ω_i sounds over the course of the piece/excerpt. Let $\Psi = \sum_{i=0}^{11} \psi_i$, and define the empirical probability mass function of the discrete random variable X by $\pi_i = \mathbb{P}(X = \omega_i) = \psi_i/\Psi$. The probability mass function for X is plotted in Fig. 3.4. Sapp (2005) uses such an empirical probability mass function, formed over a window of the dataset, to determine the key (and hence the colour) of each box in a keyscape plot (Fig. 2.6). The *correlation* (cf. Def. A.25) of the vector π and each of twenty-four vectors $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{24}$, is calculated. The vectors $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{24}$, one representing each major and minor key and referred to as *key profiles*, are determined by analysing a large

number of pieces in major and minor keys (Aarden, 2003). The window of the dataset with empirical distribution π is thought to be in the key corresponding to key profile \mathbf{a}_j , if the two vectors π and \mathbf{a}_j have a greater correlation than that of the pairs (π, \mathbf{a}_k) , where $k = 1, 2, \dots, j - 1, j + 1, \dots, 24$.

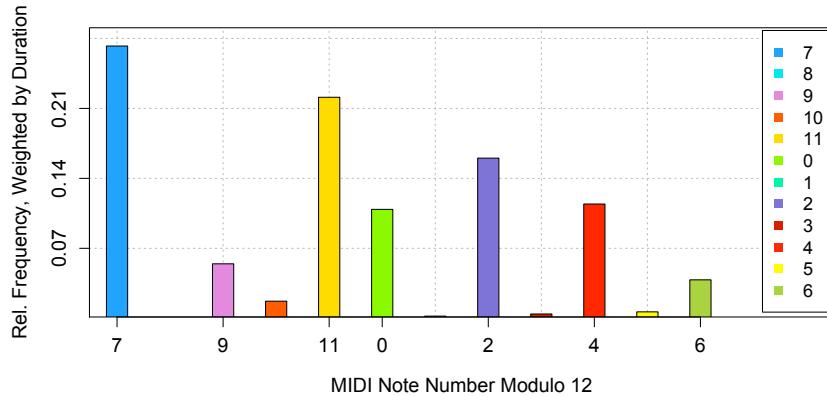


Figure 3.4: The empirical probability mass function for MMN modulo 12 *weighted by duration* in bars 1-19 of *The unanswered question* by Ives. The mass attributed to an MNN modulo 12 is weighted by the total duration in crotchet beats for which that MNN modulo 12 sounds over the course of the excerpt.

A characteristic shared by the three example empirical distributions is that they are defined over an entire excerpt. To emulate a listener's short-term memory, the calculation of empirical probability mass functions can be limited to local time *windows* of the dataset, as follows.

Definition 3.3. Sounding at and between. Recalling the definition of a datapoint $\mathbf{d} = (\omega_1, \omega_2, \dots, \omega_5)$ from (2.2), the datapoint has ontime $\omega_1 \in \Omega_1$ and duration $\omega_4 \in \Omega_4$. The datapoint \mathbf{d} is said to *sound at* time $t \in \mathbb{R}$ if its ontime ω_1 and offtime $\omega_1 + \omega_4$ satisfy $\omega_1 \leq t < \omega_1 + \omega_4$.

A datapoint $\mathbf{d} = (\omega_1, \omega_2, \dots, \omega_5)$ is said to *sound between* times $t_1, t_2 \in \mathbb{R}$, such that $t_1 < t_2$, if its ontime ω_1 and offtime $\omega_1 + \omega_4$ satisfy $\omega_1 < t_2$ and $t_1 < \omega_1 + \omega_4$.

For a piece/excerpt with dataset D , the set of datapoints in D that sound between t_1, t_2 is denoted $D[t_1, t_2]$. It is called the *window of the dataset* between times t_1 and t_2 . ■

Example 3.4. The dataset D is defined as in (3.1), and corresponds to the excerpt by Ives shown in Fig. 3.1. The datapoint $\mathbf{d} = (12, 47, 52, 6, 4) \in D$, for instance, sounds at time 12, time 17, and time 17.9, but not at time 11.9 or time 18. The same datapoint sounds between times $t_1 = 11$ and $t_2 = 13$, times $t_1 = 11$ and $t_2 = 19$, times $t_1 = 13$ and $t_2 = 17$, and times $t_1 = 17$ and $t_2 = 19$, but not between times $t_1 = 11$ and $t_2 = 12$, or times $t_1 = 18$ and $t_2 = 19$.

The window of the dataset between times $t_1 = 12$ and $t_2 = 23$ is

$$\begin{aligned} D[12, 23] = & \{(0, 59, 59, 18, 3), (0, 74, 68, 18, 2), (12, 47, 52, 6, 4), \\ & (12, 90, 77, 6, 1), (18, 47, 52, 6, 4), (18, 67, 64, 5, 3), \\ & (18, 67, 64, 6, 2), (18, 88, 76, 4, 1), (22, 86, 75, 2, 1)\}. \end{aligned} \quad (3.6)$$

The empirical probability mass functions for the dimension of MIDI note number arising from $D[0, 24]$ and $D[52, 76]$ are plotted side by side as blue and red rectangles respectively in Fig. 3.5. These dataset windows correspond to bars 1-6 and 14-19 of Fig. 3.1. It can be seen that bars 14-19 are similar to bars 1-6, the main difference being the entrance of the trumpet in bar 16. The pitch material (and hence MIDI note numbers) of the trumpet is different to that of the strings, so this causes the MNN empirical probability mass function arising from $D[52, 76]$ to be more dispersed than that of $D[0, 24]$. This is visibly the case in Fig. 3.5, where there are more red rectangles ($D[52, 76]$) than blue ($D[0, 24]$), and, where a red and blue rectangle are side

by side, the red rectangle is smaller. ■

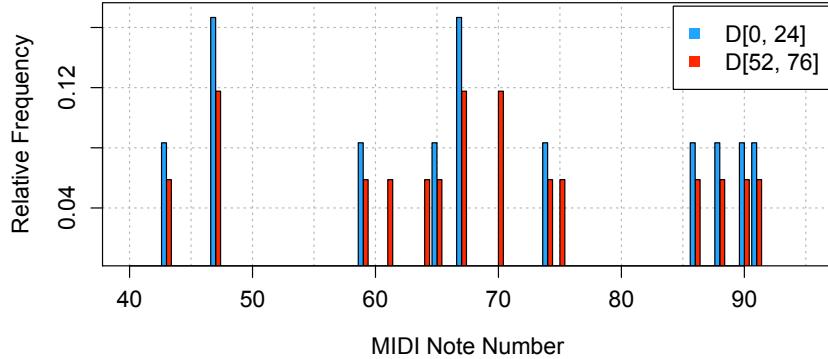


Figure 3.5: Two empirical probability mass functions plotted side by side for different dataset windows from *The unanswered question* by Ives. Mass functions for the dimension of MIDI note number arising from $D[0, 24]$ (corresponding to bars 1-6) and $D[52, 76]$ (corresponding to bars 14-19) are plotted as blue and red rectangles respectively.

3.2 Using empirical distributions to model aspects of music perception

The last paragraph of Example 3.4 in particular raises the question: what is the point of constructing empirical probability mass functions over different dimensions/viewpoints and windows of a dataset? One use of empirical distributions is as a component in a key-finding algorithm, already discussed in relation to Fig. 3.4. As listeners show sensitivity to changes in key (Thompson and Cuddy, 1989; Janata, Birk, Tillmann, and Bharucha, 2003; Tillmann, Janata, Birk, and Bharucha, 2008), a key-finding algorithm (and hence the empirical distributions on which it is based) could be said to model this aspect of music perception. A model of music perception is an

attempt to simulate the responses of participants in an experiment, where the instructions and stimuli encapsulate an apposite music-perceptual task.

Two other aspects of music perception that can be modelled using empirical probability mass functions are *uncertainty* and *likelihood*. I perceive the entrance of the trumpet at bar 16 in Fig. 3.1 to be uncertain. There are several reasons for this. First, the trumpet is a new timbre in a piece that, up to bar 16, is scored for strings. Second, the pitch material of the trumpet part is different to that of the strings. Third, the triplet rhythms in the trumpet part have not been heard previously in the piece. The entropy of a discrete random variable quantifies the uncertainty associated with the random variable's probability mass function (cf. Def. A.41 and Shannon, 1948), so leaving aside the new timbre and triplet rhythms, perhaps the uncertainty due to the pitch material of the trumpet part can be modelled by considering the entropy of appropriate empirical distributions. Let X be a discrete random variable with probability mass function \mathbf{p} given by the blue rectangles in Fig. 3.5. This is the empirical probability mass function for the MNN dimension arising from the dataset window $D[0, 24]$, where D is the dataset from (3.1) that represents Fig. 3.1. Also, let Y be a discrete random variable with the probability mass function \mathbf{q} given by the red rectangles in

Fig. 3.1 (the dataset window here is $D[52, 76]$). The entropy of X is

$$H(X) = - \sum_{i=1}^{10} p_i \log_2 p_i \quad (3.7)$$

$$\approx - (0.083 \log_2 0.083 + 0.167 \log_2 0.167 \\ + 0.083 \log_2 0.083 + \dots + 0.083 \log_2 0.083) \quad (3.8)$$

$$\approx 3.25. \quad (3.9)$$

Similarly,

$$H(Y) = - \sum_{i=1}^{14} q_i \log_2 q_i \quad (3.10)$$

$$\approx - (0.059 \log_2 0.059 + 0.118 \log_2 0.118 \\ + 0.059 \log_2 0.059 + \dots + 0.059 \log_2 0.059) \quad (3.11)$$

$$\approx 3.73. \quad (3.12)$$

There is more uncertainty associated with Y than with X , as $H(Y) > H(X)$, and this reflects the heightened uncertainty associated with the trumpet entrance in bars 14-19 of Fig. 3.1, compared with the strings in bars 1-6. Rather than using just two dataset windows of twenty-four crotchet beats in length, $D[0, 24]$ and $D[52, 76]$, it would be more thorough to consider many dataset windows $D[0, 24], D[1, 25], \dots, D[52, 76]$.² For each dataset window, an appropriate empirical distribution can be calculated, and thence the entropy of the random variable having this distribution. A plot can be constructed of entropy against time, giving an insight into how uncertainty varies over

²The window length (currently 24 crotchet beats) and step size or overlap (currently 1 crotchet beat) are thought of as parameters.

the course of the excerpt. Potter, Wiggins, and Pearce (2007) construct such plots for a monophonic piece, although they use Markov models (see Sec. 3.3) and combinations of viewpoints to form empirical distributions.

The modelling of perceived uncertainty via entropy is often referred to as modelling of *musical expectancy* (Potter et al., 2007). The topic of expectancy has received much attention, with research dating back at least as far as Meyer (1956). Narmour (1990), Schellenberg (1997), and Pearce (2005) are notable recent contributions, among many. The boundaries between musical expectancy and other perceptual phenomena, such as anticipation (Huron, 2006), tonality (Krumhansl and Shepard, 1979), and tension (Farbood, 2006; Lerdahl and Krumhansl, 2007), are often blurred. A recent summary by Trainor and Zatorre (2009) suggests that the brain uses statistical properties from recent musical input to form expectations, and that this use of statistical properties is important for explaining ‘why a note or chord that is musically unexpected continues to evoke an emotional response even when we are familiar with the piece and know at a conscious level that the unexpected chord is coming’ (p. 180).

Here is an attempt to quantify Trainor and Zatorre’s suggestion, using the excerpt by Ives from Fig. 3.1 as an example. For me, the entrance of the trumpet at bar 16 in Fig. 3.1, combined with the string parts, results in some low-likelihood chords, relative to the foregoing pitch material. By calculating a likelihood for each chord in this excerpt and plotting a function of the likelihood against time, it is possible to see if there is a local minimum in likelihood at or just after bar 16 (time 60, in crotchet beats).

Definition 3.5. Likelihood profile. Suppose that the datapoints sound-

ing at time t are elements of the dataset $S = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_l\} \subseteq D$, where D is a dataset also, and that the datapoints in S have MIDI note numbers x_1, x_2, \dots, x_l . I will use the empirical probability mass function π for the MNN dimension, arising from the dataset $(D[t - c_{\text{beat}}, t] \cup S)$, where the constant c_{beat} determines how much of D prior to time t is taken into consideration when forming the empirical distribution. The use of a subset of D local to t is intended as a naïve model of the brain's statistical analysis of recent musical input (Trainor & Zatorre, 2009). The constant c_{beat} can be thought of as the scope of a listener's short-term memory. If, for instance, $c_{\text{beat}} = 16$ and $t = 52$, then datapoints sounding between times $t - c_{\text{beat}} = 36$, and $t = 52$, as well as datapoints in S , are used to form the empirical distribution. Let X_1, X_2, \dots, X_l be independent, identically distributed random variables, each with the probability mass function π . The probability that the random variable X_i assumes the value of the MIDI note number x_i is given by the corresponding element of the probability vector, denoted $\pi(x_i)$. Therefore,

$$\mathbb{P}[(X_1, X_2, \dots, X_l) = (x_1, x_2, \dots, x_l)] = \prod_{i=1}^l \pi(x_i). \quad (3.13)$$

A geometric mean, as defined in (A.24), is taken to avoid a low-likelihood bias towards chords with more notes. The *geometric mean likelihood* of S is

$$\bar{L}(S, t, c_{\text{beat}}) = \left(\prod_{i=1}^l \pi(x_i) \right)^{(1/l)}. \quad (3.14)$$

A plot of the geometric mean likelihood of a piece/excerpt against time is called a *likelihood profile*. This definition of likelihood profile uses an em-

pirical distribution based on MIDI note numbers, but a future version could incorporate other dimensions/viewpoints. ■

For the excerpt shown in Fig. 3.1 and dataset defined in (3.1), the geometric mean likelihood is calculated for each time at which a datapoint begins or ends, and these geometric mean likelihoods are plotted against time in Fig. 3.6. The constant $c_{beat} = 16$. There is a local minimum in likelihood just after bar 16 (time 60, in crotchet beats), which corroborates my assertion that the trumpet entrance and string parts result in some low-likelihood chords.

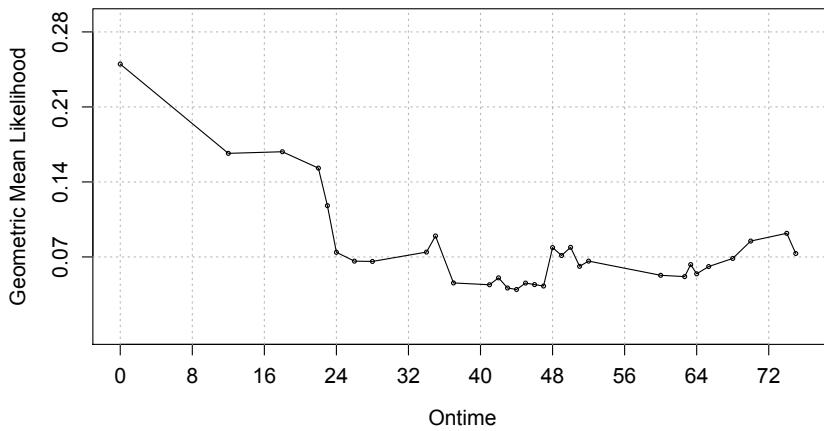


Figure 3.6: The likelihood profile for the excerpt shown in Fig. 3.1 and dataset defined in (3.1). A likelihood profile is a plot of geometric mean likelihood (3.14) against ontime.

Temperley (2004, 2007) takes a very different approach to modelling a similar aspect of music perception: the perceived likelihood of a pitch-class set in a piece. As mentioned previously (p. 35), a key profile is a vector \mathbf{a}_i with twelve elements, indicating the relative frequency of occurrence of each MNN modulo 12 in a piece/excerpt with a particular key. When plotted, a key profile resembles the plot in Fig. 3.4. There are twenty-four key profiles; one

for each major and minor key. As with Def. 3.5, suppose that the datapoints sounding at time t are elements of the dataset $S = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_l\} \subseteq D$, where D is a dataset also. In contrast to Def. 3.5, let $A = \{x_1, x_2, \dots, x_{l'}\}$ be the set of MIDI note numbers modulo 12 that are present in S , and $\{x_{l'+1}, x_{l'+2}, \dots, x_{12}\}$ be the set of MIDI note numbers modulo 12 that are not present. Given that the excerpt in which S appears is in the i th key (where $i \in \{1, 2, \dots, 24\}$), let X_1, X_2, \dots, X_{12} be independent, identically distributed random variables, each with the distribution of the key profile \mathbf{a}_i . The probability that the random variable X_j assumes the value x_j , an MNN modulo 12, is given by the corresponding element of the probability vector, denoted $\mathbf{a}_i(x_j)$. Therefore,

$$\mathbb{P}[(X_1, \dots, X_{12}) = (x_1, \dots, x_{12})] = \left(\prod_{j=1}^{l'} \mathbf{a}_i(x_j) \right) \left(\prod_{j=l'+1}^{12} (1 - \mathbf{a}_i(x_j)) \right). \quad (3.15)$$

To work out the overall likelihood of the pitch-class set A , Temperley (2004) uses (A.41), conditioning on twenty-four equiprobable keys, represented by the events B_1, B_2, \dots, B_{24} :

$$\mathbb{P}(A) = \sum_{i=1}^{24} \mathbb{P}(A | B_i) \mathbb{P}(B_i) \quad (3.16)$$

$$= \sum_{i=1}^{24} \frac{1}{24} \left(\prod_{j=1}^{l'} \mathbf{a}_i(x_j) \right) \left(\prod_{j=l'+1}^{12} (1 - \mathbf{a}_i(x_j)) \right). \quad (3.17)$$

This approach is often referred to as Bayesian, as it uses conditional probabilities. Indeed (3.16) is an equation that appears in Bayes' formula (A.43). Temperley (2004, 2007) uses Bayes' formula to define a key-finding algorithm, as well as algorithms for other music-analytical tasks: for instance, the proba-

bility of a certain succession of keys, given a certain succession of pitch-class sets, is proportional to the product of the probability of the succession of keys (whose calculation requires some assumptions), and the likelihood of a pitch-class set given a certain key (3.15). Both the key-finding algorithm and determining the likelihood of a pitch-class set rely on assumptions, primarily that a piece/excerpt is in one of the twenty-four keys represented by the key profiles. The derivation of geometric mean likelihood in (3.14) does not make this assumption.

This section has highlighted how empirical distributions can be used to model aspects of music perception, how there are many such aspects that might be modelled, and how the same or very similar aspects can be modelled by different but nonetheless plausible approaches. A model's validity is determined by the extent to which it simulates the responses of participants in an experiment, where the instructions and stimuli encapsulate an apposite music-perceptual task. In this respect, several of the models discussed above require further evaluation.

3.3 An introduction to Markov models

Both Potter et al.'s (2007) entropy model and my model for low-likelihood chords can be described as context models in a broad sense, in that probabilities are affected by what has happened recently. In mathematics, this concept was first formalised by Markov (1907/1976), who developed theory about a succession of random variables, X_0, X_1, \dots , where the distribution of X_{n+1} is dependent on the value taken by X_n . This dependency gave rise to the term *Markov chain*, and when data are treated as though their genera-

tion is governed by this dependency, it is said that a *Markov model* is being applied. Instances of the use of Markov models abound, from chemical reactions involving enzyme activity (Savageau, 1995), to the switch between an economy in fast or slow growth (Hamilton, 1989). Here I describe models of stylistic composition, beginning with a Markov model based on the material in Fig. 3.7.



Figure 3.7: Bars 3-10 of the melody from ‘Lydia’ op.4 no.2 by Gabriel Fauré (1845-1924).

Let I be a countable set called the *state space*, with members $i \in I$ called *states*. For example, the set of pitch-classes

$$I = \{F, G, A, B\flat, B, C, D, E\} \quad (3.18)$$

forms a plausible state space for the material shown in Fig. 3.7. For each $i, j \in I$ count the number of transitions from i to j in the melody and record this number, divided by the total number of transitions from state i . This

gives the following *transition matrix*:

$$\begin{matrix}
 & \text{F} & \text{G} & \text{A} & \text{B}\flat & \text{B} & \text{C} & \text{D} & \text{E} \\
 \text{F} & \left(\begin{array}{ccccccccc}
 0 & 3/4 & 1/4 & 0 & 0 & 0 & 0 & 0 & 0 \\
 2/7 & 0 & 4/7 & 1/7 & 0 & 0 & 0 & 0 & 0 \\
 1/8 & 1/2 & 0 & 0 & 1/4 & 1/8 & 0 & 0 & 0 \\
 0 & 0 & 2/3 & 1/3 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1/3 & 0 & 0 & 0 & 1/3 & 1/3 & 0 & 0 \\
 0 & 0 & 1/3 & 1/3 & 0 & 0 & 1/3 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0
 \end{array} \right) = \mathbf{P}. \quad (3.19)
 \end{matrix}$$

For example, in Fig. 3.7 there are four transitions from F, the first element of the state space, hence the denominator 4 in the nonzero entries of the first row of \mathbf{P} in (3.19). Of the four transitions, three are to G, the second element of the state space, hence $p_{1,2} = 3/4$, and one transition to A, the third element, hence $p_{1,3} = 1/4$. So from Fig. 3.7, \mathbf{P} is the result of this counting process for the pitch-classes F, G, A, G, F, G, A, B, . . .³ Putting this matrix to use in a compositional scenario requires the generation of an initial state. For instance

$$\mathbf{a} = (\frac{1}{2}, 0, \frac{1}{2}, 0, 0, 0, 0, 0, 0) \quad (3.20)$$

means that the initial pitch-class of a generated melody will be F with prob-

³This example might be taken to imply that training a model (Mitchell, 1997) consists of defining a transition matrix based solely on observed transitions. While this is the case here and in subsequent chapters, it is often not the case in natural language processing, where zero probabilities can be artificially inflated (*smoothing*, Manning and Schütze, 1999).

ability $\frac{1}{2}$, and A with probability $\frac{1}{2}$. (The probabilities contained in \mathbf{a} do not have to be drawn empirically from the data, but often they are.) I will use upper-case notation $(X_n)_{n \geq 0} = X_0, X_1, \dots$ for a succession (more commonly called a sequence) of random variables, and lower-case notation $(i_n)_{n \geq 0}$ for when these random variables assume values. Suppose $i_0 = A$, then we look along the third row of \mathbf{P} (as A is the third element of the state space) and randomly choose between $X_1 = F$, $X_1 = G$, $X_1 = B$, $X_1 = C$, with respective probabilities $\frac{1}{8}, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}$. Continuing in this fashion, suppose $i_1 = B$. Looking along the fifth row of \mathbf{P} , a random, equiprobable choice is made between $X_2 = G$, $X_2 = C$, $X_2 = D$. And so on. Below are three melodies generated from the Markov model $(I, \mathbf{P}, \mathbf{a})$ using pseudo-random numbers. The number of notes (thirty-two) and phrase structure are maintained from Fig. 3.7.

$$\begin{aligned} & A, G, F, G, F, G, A, B, G, \quad F, G, F, G, A, B, D, E, \\ & B, C, A, F, G, \quad B\flat, A, F, G, A, G, A, B, G, A. \end{aligned} \tag{3.21}$$

$$\begin{aligned} & A, G, A, B, D, C, B\flat, A, F, \quad G, F, A, B, D, C, A, G, \\ & A, G, F, A, F, \quad A, F, G, F, G, A, G, F, A, G. \end{aligned} \tag{3.22}$$

$$\begin{aligned} & F, A, B, G, F, G, F, G, A, \quad B, C, A, G, F, G, F, G, \\ & B\flat, A, G, A, G, \quad A, F, G, B\flat, A, B, G, F, G, A. \end{aligned} \tag{3.23}$$

I return to comment on these melodies at the start of Chapter 8 (p. 184). Here are the formal definitions of Markov model and Markov chain (Norris, 1997).

Definition 3.6. Markov model. A Markov model for a piece (possibly

many pieces) of music consists of:

1. A countable set I called the state space, with a well-defined, onto mapping from the score of the piece to elements of I .
2. A transition matrix \mathbf{P} such that for $i, j \in I$, $p_{i,j}$ is the number of transitions in the music from i to j , divided by the total number of transitions from state i .
3. An initial distribution $\mathbf{a} = (a_i : i \in I)$, enabling the generation of an initial state. ■

Definition 3.7. Markov chain. Let $(X_n)_{n \geq 0}$ be a sequence of random variables, and I , \mathbf{P} , \mathbf{a} be as in Def. 3.6. It is said that $(X_n)_{n \geq 0}$ is a Markov chain if

- (i) \mathbf{a} is the distribution of X_0 ;
- (ii) for $n \geq 0$, given $X_n = i$, X_{n+1} is independent of X_0, X_1, \dots, X_{n-1} , and has distribution $(p_{i,j} : j \in I)$.

Writing these conditions as equations, for $n \geq 0$ and $i_0, i_1, \dots, i_{n+1} \in I$,

- (i) $\mathbb{P}(X_0 = i_0) = \mathbf{a}_{i_0};$
- (ii) $\mathbb{P}(X_{n+1} = i_{n+1} \mid X_0 = i_0, X_1 = i_1, \dots, X_n = i_n)$
 $= \mathbb{P}(X_{n+1} = i_{n+1} \mid X_n = i_n) = p_{i_n, i_{n+1}}.$

Conditions (i) and (ii) apply to finite sequence of random variables as well. It is also possible to model dependence in the opposite direction. That is, for $n \geq 1$, given $X_n = i$, X_{n-1} is independent of X_{n+1}, X_{n+2}, \dots . ■

In summary, this chapter has addressed how to calculate probabilities from dataset representations of music. Several examples were given where these calculations form the basis for modelling aspects of music perception. There are many aspects to music perception that might be modelled, and the same or very similar aspects can be modelled by different but nonetheless plausible approaches. Markov models were introduced, and these appear again in Chapters 5, 8, and 9.

4

Literature review: Discovery of patterns in music

It seems uncontroversial to suggest that *repetition* plays a central role in our perception of musical structure. In Schenker's words (1906/1973): 'Only by repetition can a series of tones be characterized as something definite. Only repetition can demarcate a series of tones and its purpose. Repetition thus is the basis of music as an art' (p. 5).¹

Schenker's discussion presents several kinds of repetition. One of his examples, shown in Fig. 4.1, exemplifies what I will call *exact* repetition. Also cited is the excerpt shown in Fig. 4.2. I prefer to call the latter *repetition with interpolation*, since there are differences between the original statement and repetition: mainly the interpolated notes F5, E5, D5, C5, and B4 in the second half of bar 3. The excerpt in Fig. 4.3 is another example of repetition with interpolation, but this time there is a larger number of interpolated notes. For the sake of clarity, those notes in the original statement that are repeated have black noteheads, as do the repeated notes themselves. There will be further cause to consider how the amount of interpolation affects perception of musical structure. In Fig. 4.3, the durations of some of the

¹There is a connection worth emphasising here between demarcation and Gestalt principles, in general and as applied to music (von Ehrenfels, 1890/1988; Bregman, 1990; Huron, 2001a; Wiggins, 2007).

notes differ between original statement and repetition. For instance, the C3 quaver in bar 1 becomes a C3 crotchet in bar 25. This relaxation of the requirement for exact repetition of duration seems to be in keeping with Schenker's examples.



Figure 4.1: Bars 1-4 from the first movement of the Piano Sonata no.11 in B \flat major op.22 by Ludwig van Beethoven (1770-1827). The brackets are Schenker's (1906/1973, p. 6).

An aspect of repetition overlooked by the analysis in Fig. 4.3 is that some of the notes among the original statement have their onsets shifted relative to others in the repetition. For instance, the downbeat notes of bars 2 and 4 in the first violin are shifted from the downbeats of bars 26 and 28 due to the triplet variation technique (see the arrows in Fig. 4.3), whereas the corresponding notes in the 'cello remain on the downbeat. The issue of repetitions involving shifted notes is revisited in Sec. 4.1 (p. 65).

Schenker observes that 'not only the melody but the other elements of music as well (e.g., rhythm, harmony, etc.) may contribute to the associative effect of more or less exact repetition' (Schenker, 1906/1973, p. 7). In this



Figure 4.2: Bars 1-4 from the first movement of the Piano Sonata no.8 in A minor K310 by Wolfgang Amadeus Mozart (1756-1791). The brackets are Schenker's (1906/1973, p. 5).

Figure 4.3: Bars 1-4 and 25-28 from the fourth movement of the Octet in F major D803 by Franz Schubert (1797-1828). The brackets indicate an instance of repetition with interpolation. For the sake of clarity, those notes in the original statement that are repeated have black noteheads, as do the repeated notes themselves. The arrows are for the purposes of discussion.

spirit, I propose three further types of repetition:

Transposed real. The excerpt shown in Fig. 4.4 is an example of transposed repetition, sometimes referred to as a *real sequence*. If each note in the original statement is transposed up 3 semitones, this gives the notes contained in the repetition. The exactness of the semitonic transposition is what defines a real sequence.

Transposed tonal. *Tonal sequences*, on the other hand, are usually defined as adjusted real sequences, where adjustments are made in order to remain in key. For instance, the first pair of brackets in Fig. 4.5 indicates a tonal sequence. Most notes in the original statement—such as A5 in the first violin—are transposed down 3 semitones to give notes contained in the repetition, but some—such as the F♯5 in the second violin—must be transposed down 4 semitones to remain in key.

Durational. This type of repetition in a score of a piece can be obscured in performance: a staccato crotchet might be perceived as a quaver, and notes are often sustained to thicken the texture. However, I would argue that trying to discover durational repetition is still worthwhile. It can underpin the composition of entire pieces, even if it is lost in performance. For instance, the excerpt in Fig. 4.6 constitutes durational repetition. It is taken from an isorhythmic motet, a defining feature of which is ‘the periodic repetition or recurrence of rhythmic configurations, often with changing melodic content’ (Bent, 2001, p. 618). Durational repetition is also used in later musical periods, often in much more obvious ways.

FIGURE REMOVED FROM ELECTRONIC VERSION FOR COPYRIGHT REASONS

Figure 4.4: Bars 38-43 and 131-136 from the first movement of the Piano Concerto no.1 by Béla Bartók (1881-1945). The brackets indicate an instance of a real sequence. Black noteheads help to show which notes are involved. © Copyright 1927 by Universal Edition. Copyright renewed 1954 by Boosey & Hawkes, Inc., New York. Reproduced by permission.

[Largo]

Violin I

Violin II

Continuo {
 6 (8) 6 # 6 6 6 6 6

21
 5 6 # 7 6 6 5 4 3

Figure 4.5: Bars 18-24 from the Allemande of the Chamber Sonata in B minor op.2 no.8 by Arcangelo Corelli (1653-1713). The first pair of brackets indicates a tonal sequence. The next three brackets indicate another. Black noteheads help to show which notes are involved, and numbers below the continuo part are figured bass notation.

The image shows three staves of musical notation from a four-part setting of the hymn 'Albanus roseo rutilat'. The notation is in common time, with a bass clef on each staff.

Staff 1 (Top):

- Measure 1: Al - - - ba - nus ro - se - o
- Measure 2: Quo - - que fe - ren - - dus
- Measure 3: Al -

Staff 2 (Middle):

- Measure 6: ru - ti - lat
- Measure 33: rum et splen - dor
- Measure 36: pon - - tem; mar -
- Measure 37: do -

Staff 3 (Bottom):

- Measure 36: u - ni - cens (b) dat
- Measure 37: ti - ri - i me
- Measure 38: mi - - - - ni

Brackets are used to group notes of equal duration. In the first staff, the 'o' in 'Al' and the 'd' in 'dus' are bracketed. In the second staff, the 'u' in 'rum' and the 'e' in 'et' are bracketed. In the third staff, the 'u' in 'u-ni' and the 'c' in 'cens' are bracketed, and the 'm' in 'me' and the 'n' in 'ni' are bracketed.

Figure 4.6: Excerpts from 'Albanus roseo rutilat' by John Dunstable (c 1390-1453). The brackets indicate an instance of durational repetition. For the sake of clarity, those notes in the original statement whose durations are repeated have black noteheads, as do the repeated notes themselves.

The class of repetition could be broadened further. However, Schenker's own examples of repetition involving 'other elements of music' arguably stretch the term *repetition* too far, to the point where *imitation* might be more appropriate. For example, in Fig. 4.7 a motif spanning an octave (G3 to G4) is bracketed. The next bracketed occurrence spans a compound minor third (G3 to B \flat 4), so arguably is more accurately described as an imitation rather than a repetition of the original statement. It certainly does not correspond to exact repetition, repetition with interpolation, real or tonal transposition. The bracketed motif does correspond to durational repetition—three consecutive crotchets—but the lack of further bracketed instances suggests it was not this common durational pattern but a particular pitch profile that Schenker had in mind. That said, I concur with Schenker's point of view that repetitive/imitative material can move voices (in this case from the 'cello to the viola). It could be inferred from recent work on separating musical textures into perceptually valid voices and streams (Cambouropoulos, 2008) that repetitive material ought to remain within the same voice, but to stipulate as much at this stage seems premature.

Definition 4.1. Proto-analytical class of repetition types. For ease of reference, the five types of repetition outlined above (exact, with interpolation, transposed real, transposed tonal, and durational) are labelled the *proto-analytical class* of repetition types. ■

The *proto-analytical class* of repetition types can be thought of as the basic constituents of a proper analytical method, but an analytical method consisting of these repetition types alone is plainly insufficient—hence *proto*. The proto-analytical class is oblivious to some basic concepts, for instance

Allegro

The musical score consists of two systems of music. The first system (bars 1-5) features eighth-note patterns with dynamic marks **f** (fortissimo) at the beginning and end. The second system (bars 6-12) shows a transition with dynamic marks **f**, **p** (pianissimo), and a grace note with a slur. Brackets are present above the staves in both systems.

Figure 4.7: Bars 1-12 from the first movement of the String Quartet in G minor, 'The Horseman', op.74 no.3 by Joseph Haydn (1732-1809). The brackets are Schenker's (1906/1973, p. 8).

scale, triad, and octave equivalence, not to mention concepts that comprise more sophisticated analytical methods such as Schenkerian theory (Forte and Gilbert, 1982) or Ockelford's (2005) zygonic theory. Even in terms of handling repetition, there are occasions where the proto-analytical class is not wholly adequate, such as in Fig. 4.3. A more positive characteristic of the class is its adherence to the principle of 'repetition as creator of form' (Schenker, 1906/1973, p. 9), meaning that it does not distinguish between small- and large-scale repetitions. There is no need to agonise over definitions of motif, theme, section, and then go shoehorning repeated material into one category or the other. Rather, the proto-analytical class can be used to identify various instances of repetition in a piece, and then the analyst can categorise these instances according to small- and large-scale considerations if desired. Furthermore, according to Lerdahl and Jackendoff (1983), 'failure to flesh out the notion of parallelism [of which repetition is a component] is a serious gap in our attempt to formulate a fully explicit theory of musical understanding' (p. 53). The ability to discover instances of repetition algorithmically would contribute to filling this gap.

4.1 Algorithms for pattern discovery in music

Although translational patterns (defined on p. 68) are not the only type of pattern that could matter in music analysis, many music analysts would acknowledge that discovering translational patterns forms part of the preparation when writing an analytical essay (as in the analysis essay question

on p. 22). Even if the final essay pays little or no heed to the discovery of translational patterns, neglecting this preparatory task entirely could result in failing to mention something that is musically very noticeable or important. Hence I am motivated by the prospect of automating the discovery task, as it could have interesting implications for music analysts (and music listeners in general), enabling them to engage with pieces in a novel manner. I also consider this task to be an open problem within music information retrieval (MIR), so attempting to improve upon current solutions is another motivating factor.

Definition 4.2. Intra-opus discovery of translational patterns. Given a piece of music in a symbolic representation, discover musically noticeable and/or important translational patterns that occur within one or more geometric representations. ■

In MIR there do not seem to be clear distinctions between the terms pattern *discovery* (Conklin and Bergeron, 2008; Hsu, Liu, and Chen, 2001; Meredith, Lemström, and Wiggins, 2002; Ren, Smith, and Medina, 2004), *extraction* (Lartillot, 2005; Meek and Birmingham, 2003; Rolland, 2001), *identification* (Forth and Wiggins, 2009; Knopke and Jürgensen, 2009), and *mining* (Chiu, Shan, Huang, and Li, 2009), at least in the sense that most of the papers just cited address very similar discovery tasks to that stated in Def. 4.2. Conklin and Bergeron (2008) give the label *intra-opus* discovery to concentrating on patterns that occur *within* pieces. An alternative is *inter-opus* discovery, where patterns are discovered *across* many pieces of music (Conklin and Bergeron, 2008; Knopke and Jürgensen, 2009). This makes it possible to gauge the typicality of a particular pattern relative to

the corpus style. Terms that *are* clearly distinguished in MIR are pattern *discovery* and *matching* (Clifford, Christodoulakis, Crawford, Meredith, and Wiggins, 2006). Pattern matching is the central process in content-based retrieval (Ukkonen, Lemström, and Mäkinen, 2003), where the user provides a query and then the algorithm searches a music database for more or less exact instances of the query. The output is ranked by some measure of proximity to the original query. The flow chart in Fig. 4.8 shows a framework for the task of pattern matching: algorithms cast within this framework abound, as robust pattern matching systems are something of a holy grail in MIR (an overview is given by Downie, 2003; and a specific example is found in Doraisamy and Rüger, 2003). This matching task is quite different from the intra-opus discovery task, where there is neither a query nor a database as such, just a single piece of music, and no obvious way of ranking an algorithm's output. The flow chart in Fig. 4.9 depicts a framework for a pattern discovery system: algorithms that have been or could be cast within this framework are proposed by Meredith et al. (2002); Forth and Wiggins (2009); Conklin and Bergeron (2008); Cambouropoulos (2006); Lartillot (2004). While I have stressed their differences, some authors attempt to address both discovery and matching tasks (Meredith, Lemström, and Wiggins, 2003; Wiggins, Lemström, and Meredith, 2002), suggesting that representations/algorithms that work well for one task might be adapted and applied fruitfully to the other.

Some attempts at pattern discovery have been made with audio representations of music (Peeters, 2007). However, I, like the majority of work cited in this section, begin with a symbolic representation. Work on symbolic

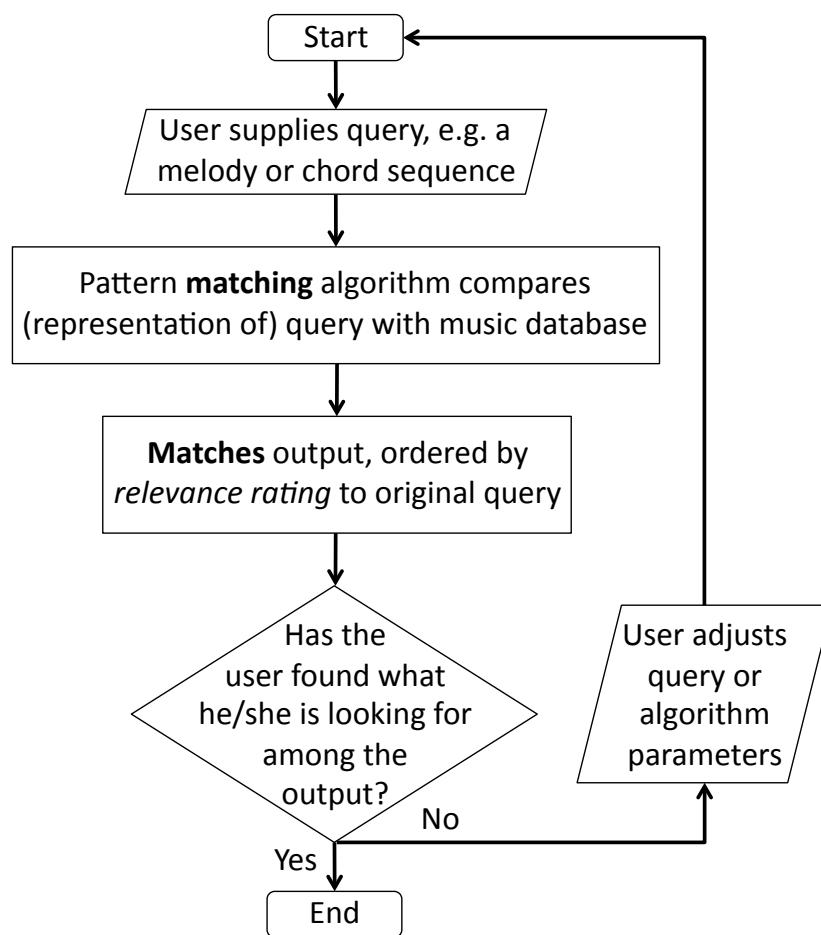


Figure 4.8: Flow chart depicting a framework for a pattern matching system.

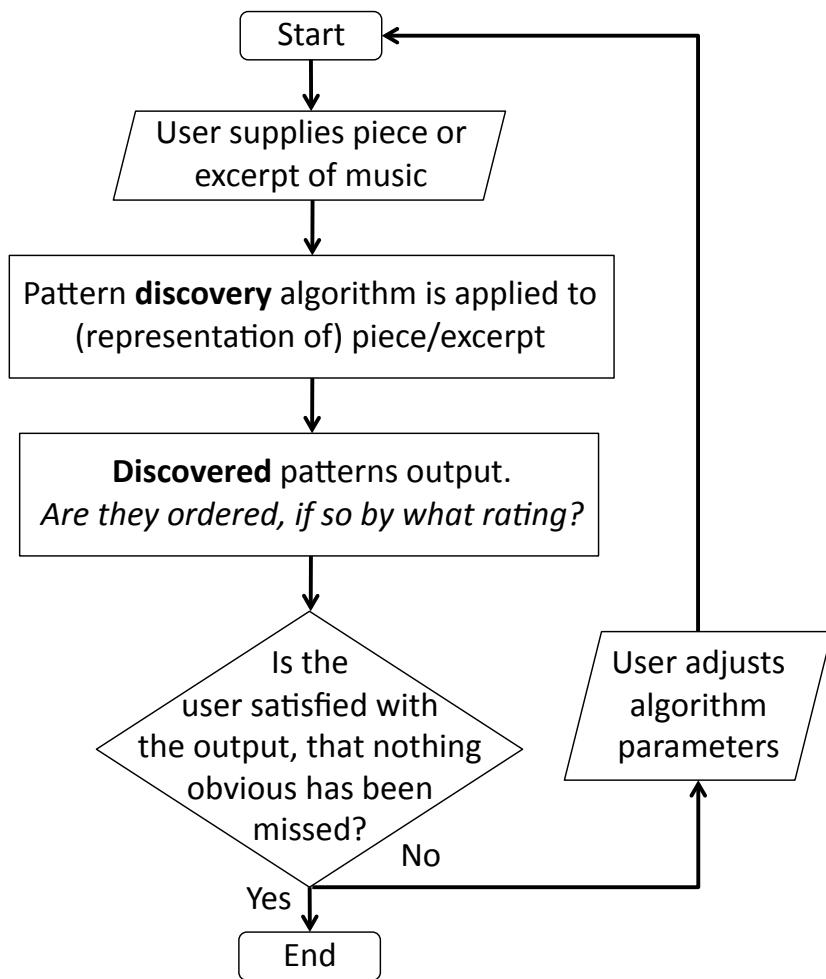


Figure 4.9: Flow chart depicting a framework for a pattern discovery system.

representations can be categorised into string-based (Cambouropoulos, 2006; Chiu et al., 2009; Conklin and Bergeron, 2008; Hsu et al., 2001; Knopke and Jürgensen, 2009; Lartillot, 2005; Meek and Birmingham, 2003; Ren et al., 2004; Rolland, 2001) and geometric approaches (Forth and Wiggins, 2009; Meredith, 2006b; Meredith et al., 2003, 2002), and which approach is most appropriate depends on the musical situation. For instance the string-based method is more appropriate for the excerpt in Fig. 4.10. I propose that the most salient pattern in this short excerpt consists of the notes C5, B4, G4, E4, B4, A4, ignoring ornaments for simplicity. The simplest way to discover the three occurrences of this pattern is to represent the excerpt as a string of MIDI note numbers and then to use an algorithm for pattern discovery in strings. The string 72, 71, 67, 64, 71, 69, ought to be discovered, and the user relates this back to the notes C5, B4, G4, E4, B4, A4. The geometric method could be used here, but it is not so parsimonious, as it involves mapping the onttime-MNN pairs $\{(0, 72), (1, 71), (1\frac{1}{4}, 67), (1\frac{1}{2}, 64), \dots, (8, 69)\}$ to a sequential time domain $\{(0, 72), (1, 71), (2, 67), (3, 64), \dots, (22, 69)\}$.

FIGURE REMOVED FROM ELECTRONIC VERSION FOR COPYRIGHT REASONS

Figure 4.10: Bars 1-3 of the Introduction from *The Rite of Spring* (1913) by Igor Stravinsky (1882-1971), annotated with MIDI note numbers and ontimes in crotchets starting from zero. For clarity, phrasing is omitted and ornaments are not annotated. © Copyright 1912, 1921 by Hawkes & Son Ltd., London. Reproduced by permission of Boosey & Hawkes Music Publishers Ltd.

On the other hand, the geometric method is better suited to finding the most salient pattern in Fig. 4.11A, consisting of all the notes in bar 13

except the tied-over G4. This pattern occurs again in bar 14, transposed up a fourth, and then once more at the original pitch in bar 15. Each note is annotated with its relative height on the stave (MPN), taking C4 to be 60. Underneath the stave, ontimes are measured in quaver beats starting from zero. The first note in this excerpt, G3, can be represented by the data-point $\mathbf{d}_1 = (0, 57)$, since it has ontime 0 and morphetic pitch number 57. A scatterplot of morphetic pitch number against ontime for this excerpt is shown in Fig. 4.11B. Restricting attention to bars 13-15, the dataset is

$$D = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_{26}\}. \quad (4.1)$$

A *pattern* is defined as a non-empty subset of a dataset. As an example, consider the patterns

$$P = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_8\}, \text{ and } Q = \{\mathbf{d}_9, \mathbf{d}_{11}, \mathbf{d}_{12}, \dots, \mathbf{d}_{17}\}. \quad (4.2)$$

The vector that translates \mathbf{d}_1 to \mathbf{d}_9 is

$$\mathbf{d}_9 - \mathbf{d}_1 = (3, 60) - (0, 57) = (3, 3) = \mathbf{v}. \quad (4.3)$$

This vector has been given the label $\mathbf{v} = (3, 3)$. It is this same vector \mathbf{v} that translates \mathbf{d}_2 to \mathbf{d}_{11} , \mathbf{d}_3 to \mathbf{d}_{12} , ..., \mathbf{d}_8 to \mathbf{d}_{17} . Recalling the definitions of P and Q from (4.2), it is more succinct to say that ‘the translation of P by \mathbf{v} is equal to Q ’. This translation is indicated in Fig. 4.11C.

Looking at Fig. 4.11C it is evident that as well as Q being a translation of P , pattern R is also a translation of P . Meredith et al. (2002) call $\{P, Q, R\}$

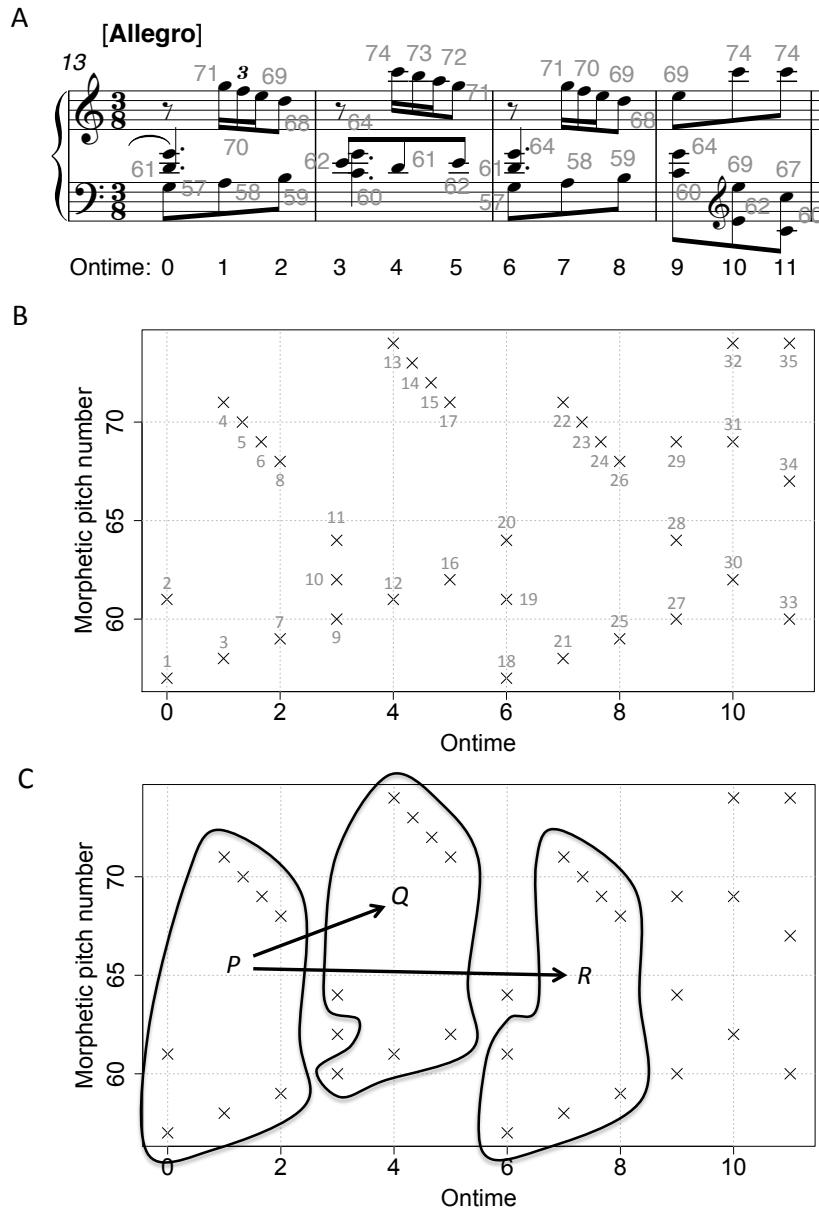


Figure 4.11: (A) Bars 13-16 of the Sonata in C major L3 by Domenico Scarlatti (1685-1757), annotated with morphetic pitch numbers and ontimes; (B) each note from the excerpt is converted to a point consisting of an onttime and a morphetic pitch number. Morphetic pitch number is plotted against onttime, and points are labelled in lexicographical order d_1 to d_{35} ; (C) the same plot as above, with three ringed patterns, P , Q , R . Arrows indicate that both Q and R are translations of P .

the *translational equivalence class* of P in D , notated

$$\text{TEC}(P, D) = \{P, Q, R\}. \quad (4.4)$$

The TEC gives all the occurrences of a pattern in a dataset.

So P is an example of a *translational pattern*, as translations of P , namely Q and R , exist in the dataset D . Some formal definitions follow.

Definition 4.3. Translational pattern and related concepts (Meredith et al., 2002). Let D be a dataset with k dimensions (cf. Def. 2.2). A *pattern* P is defined as a non-empty subset of the dataset D .

For an arbitrary vector $\mathbf{v} \in \mathbb{R}^k$, and an arbitrary pattern P , the *translation* of the pattern P by the vector \mathbf{v} is defined by

$$\tau(P, \mathbf{v}) = \{\mathbf{p} + \mathbf{v} : \mathbf{p} \in P\}. \quad (4.5)$$

Let P, Q be arbitrary patterns. It is said that P is *translationally equivalent* to Q , written $P \equiv_{\tau} Q$, if and only if there exists some vector $\mathbf{v} \in \mathbb{R}^k$ such that $Q = \tau(P, \mathbf{v})$. It can be shown that \equiv_{τ} is an *equivalence relation* in the proper mathematical sense (cf. Def. A.23).

For a pattern P in a dataset D , the pattern P is a *translational pattern in D* if there exists at least one subset $Q \subseteq D$ such that P and Q contain the same number of elements, and *one* nonzero vector \mathbf{v} translates each datapoint in P to a datapoint in Q .

For an arbitrary dataset D , and an arbitrary pattern $P \subseteq D$, the *translational equivalence class* of P in D is defined by

$$\text{TEC}(P, D) = \{Q \subseteq D : Q \equiv_{\tau} P\}. \quad (4.6)$$

The *translators* of P in D are given by the set

$$T(P, D) = \{\mathbf{v} \in \mathbb{R}^k : \tau(P, \mathbf{v}) \subseteq D\}. \quad (4.7)$$

■

In the example in Fig. 4.11, two dimensions were considered (ontime and morphetic pitch number). The definitions and pattern discovery algorithms given by Meredith et al. (2003) extend to k dimensions; MIDI note number, duration, and staff are among many possible further dimensions.

The string-based method is not so well suited to Fig. 4.11A. The first step would be voice separation, generating perceptually valid melodies from the texture. Sometimes the scoring of the music makes separation simple (Knopke and Jürgensen, 2009), but even when voicing contains ambiguities, there are algorithms that can manage (Cambouropoulos, 2008; Chiu et al., 2009). Supposing fragments of the pattern in Fig. 4.11A were discovered among separated melodies, these fragments still would have to be correctly reunited. In this instance, even the most sophisticated string-based method (Conklin and Bergeron, 2008) does not compare with the efficiency of the geometric method. The key difference between geometric and string-based approaches is the binding of ontimes to other musical information in the former, and the decoupling of this information in the latter. Both are valid methods for discovering patterns in music.

The reporting of existing intra-opus algorithms often mentions running time (Chiu et al., 2009; Hsu et al., 2001; Meredith, 2006b; Meredith et al., 2003, 2002), occasionally *recall* is given (Meek and Birmingham, 2003; Roland, 2001), and sometimes *precision* (Lartillot, 2005). With the inter-opus

discovery task (Conklin and Bergeron, 2008; Knopke and Jürgensen, 2009) an algorithm's output tends not to be compared with a human benchmark. The justification is that ‘investigations of entire collections require considerable amounts of time and effort on the part of researchers’ (Knopke and Jürgensen, 2009, p. 171). Still, is it not worth knowing how an algorithm performs on a subset of the collection?

4.2 The family of Structure Induction Algorithms

Evidently, analysts are interested in annotating and discussing repeated patterns (as in Figs. 4.1-4.7), so it is worth investigating whether a pattern discovery algorithm can be defined for an analogous task. The family of Structure Induction Algorithms (SIA, Meredith et al., 2002) is of particular interest, because of all existing pattern discovery algorithms, the patterns that it returns are most consistent with the proto-analytical class (cf. Def. 4.1).² For instance, real sequences can be returned when running SIA on a dataset projection including ontine and MNN, tonal sequences can be returned when running SIA on a projection including ontine and MPN, and durational repetitions can be returned when running SIA on a projection including ontine and duration. Exact repetitions and repetitions with interpolation can be returned for any of the above. Alternative pattern discovery algorithms that might have been used were mentioned in relation to Fig. 4.9. These, as well as other candidates (Meek and Birmingham, 2003; Chiu et al., 2009; Knopke

²SIA is a particular algorithm. *SIA family* is a collective term for several algorithms that contain the acronym SIA.

and Jürgensen, 2009) are either not as consistent with the proto-analytical class as the SIA family, or make musical assumptions that do not apply to textures in which additional voices may appear/disappear midway through an excerpt.

In equation (4.2), pattern P from Fig. 4.11 was introduced without explaining how it is discovered. It could be discovered by calculating all the TECs in the dataset D , and then certainly $\text{TEC}(P, D)$ will be among the output. However this approach is tremendously expensive and indiscriminate. It is expensive in terms of computational complexity, as there are 2^n patterns to partition into equivalence classes, where $n = |D|$ is the cardinality of the dataset. Moreover, it is indiscriminate as no attempt is made to restrict the output in terms of musical importance: while P is arguably of importance, not all subsets of D are worth considering, yet they will also be among the output. The set E in Fig. 4.12 represents the output of this expensive and indiscriminate approach.

Therefore Meredith et al. (2002) restrict the focus to a smaller set F , by considering how a pattern like P is *maximal*. Recalling (4.1) and (4.2), the pattern P is maximal in the sense that it contains all datapoints that are translatable in the dataset D by the vector $\mathbf{v} = (3, 3)$. It is called a *maximal translatable pattern*.

Definition 4.4. Maximal translatable pattern (Meredith et al., 2002). Let D be a dataset with k dimensions, and $\mathbf{v} \in \mathbb{R}^k$ be an arbitrary vector. The *maximal translatable pattern* of the vector \mathbf{v} in the dataset D , written $\text{MTP}(\mathbf{v}, D)$, is

$$\text{MTP}(\mathbf{v}, D) = \{\mathbf{d} \in D : \mathbf{d} + \mathbf{v} \in D\}. \quad (4.8)$$

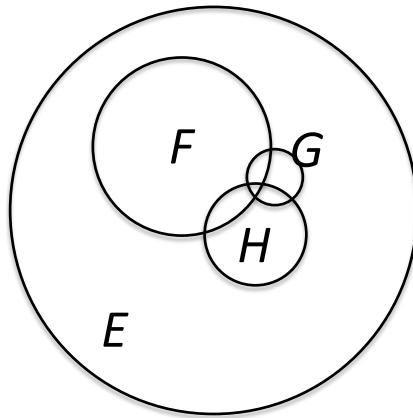


Figure 4.12: A Venn diagram (not to scale) for the number of patterns (up to translational equivalence) in a dataset. The total E is shown relative to the number typically returned by SIATEC (F), COSIATEC (G), and SIACT (H). SIACT is introduced in Sec. 7.1.

As with datasets, maximal translatable patterns are assumed to be in lexicographic order (cf. Def. 2.3), unless stated otherwise. ■

It can be verified that for P in (4.2) and $\mathbf{v} = (3, 3)$, $P = \text{MTP}[(3, 3), D]$. Meredith et al.'s Structure Induction Algorithm (SIA) calculates the set of all pairs $(\mathbf{v}, \text{MTP}(\mathbf{v}, D))$ in a dataset such that $\text{MTP}(\mathbf{v}, D)$ is nonempty, which requires $O(kn^2 \log n)$ calculations.³ While the TEC of each MTP must still be determined to give the set F in Fig. 4.12, this approach is enormously less expensive than partitioning 2^n patterns, and involves a decision about musical importance: ‘In music, MTPs often correspond to the patterns involved in perceptually significant repetitions’ (Meredith et al., 2002, p. 331).

Definition 4.5. SIA and SIATEC (Meredith et al., 2002). Let $D = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n\}$ be a dataset with k dimensions. The first step of SIA is to

³While it is possible to reduce the computational complexity to $O(kn^2)$ by hashing (Meredith, 2006b), doing so relies on prior assumptions about the dataset.

traverse the upper triangle of the similarity array

$$\mathbf{A} = \begin{pmatrix} \mathbf{d}_1 - \mathbf{d}_1 & \mathbf{d}_2 - \mathbf{d}_1 & \cdots & \mathbf{d}_n - \mathbf{d}_1 \\ \mathbf{d}_1 - \mathbf{d}_2 & \mathbf{d}_2 - \mathbf{d}_2 & \cdots & \mathbf{d}_n - \mathbf{d}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{d}_1 - \mathbf{d}_n & \mathbf{d}_2 - \mathbf{d}_n & \cdots & \mathbf{d}_n - \mathbf{d}_n \end{pmatrix} \quad (4.9)$$

by row. If the vector $\mathbf{w} = \mathbf{d}_j - \mathbf{d}_i$ is not equal to a previously calculated vector then a new vector-MTP pair is created, $(\mathbf{w}, \text{MTP}(\mathbf{w}, D))$, with \mathbf{d}_i as the first element of $\text{MTP}(\mathbf{w}, D)$. Otherwise $\mathbf{w} = \mathbf{u}$ for some previously calculated vector \mathbf{u} , in which case \mathbf{d}_i is included as the last element of $\text{MTP}(\mathbf{u}, D)$, in the vector-MTP pair $(\mathbf{u}, \text{MTP}(\mathbf{u}, D))$.

It is possible to determine the set F for a dataset D by first running SIA on the dataset and then calculating the TEC of each MTP. The Structure Induction Algorithm for Translational Equivalence Classes (SIATEC) performs this task, and requires $O(kn^3)$ calculations. ■

To my knowledge, there are two further algorithms that apply the geometric method to intra-opus translational pattern discovery: Meredith et al.'s (2003) COvering Structure Induction Algorithm for Translational Equivalence Classes (COSIATEC) and a variant proposed by Forth and Wiggins (2009). COSIATEC rates patterns according to a heuristic for musical importance, and discards many discovered patterns on each iteration (cf. step 1 in Def. 4.6). As such, COSIATEC tends to return a smaller number of patterns than SIATEC, indicated by the set labelled G in Fig. 4.12. The name COSIATEC derives from the idea of creating a *cover* for the input dataset.

Definition 4.6. COSIATEC (Meredith et al., 2003). Let D be a

dataset with k dimensions.

1. Run SIATEC on $D_0 = D$, rate the discovered patterns using a heuristic for musical importance, and return the pattern P_0 that receives the highest rating.
2. Define a new dataset D_1 by removing from D_0 each datapoint that belongs to an occurrence of P_0 .
3. Repeat step 1 for D_1 to give P_1 , repeat step 2 to define D_2 from D_1 , and so on until the dataset D_{N+1} is empty.
4. The output is

$$G = \{TEC(P_0, D_0), \dots, TEC(P_N, D_N)\}. \quad (4.10)$$

■

Forth and Wiggins's (2009) variant of COSIATEC uses a nonparametric version of the heuristic for musical importance and requires only one run of SIATEC. One run of SIATEC reduces the computational complexity of Forth and Wiggins's (2009) variant of COSIATEC. It does mean, however, that the output is restricted to $F \cap G$ in Fig. 4.12.

4.3 Recall and precision

How does one know when an improved method for pattern discovery has been achieved? For a certain task, there needs to be a collection of musical patterns that are deemed worthy of discovery. This collection is called a

benchmark, and its constituent musical patterns are referred to as *targets*. The formation of a benchmark may involve collating the task responses of several participants, or a benchmark may be formed by a single expert. Either way, there is an assumption underlying the use of a benchmark that the computational method attempts to emulate human task performance. As use of a benchmark enables the task performance of two or more computational methods to be compared, this assumption is generally accepted.⁴

Two common metrics for evaluating the performance of a computational method on a discovery (or retrieval) task are *recall* and *precision* (Manning and Schütze, 1999). If Ω represents the collection of all musical patterns for a certain task, Ψ represents the benchmark of targets, and Λ the patterns returned by a computational method, then

$$\text{recall} = \frac{|\Psi \cap \Lambda|}{|\Psi|}, \quad \text{precision} = \frac{|\Psi \cap \Lambda|}{|\Lambda|}, \quad (4.11)$$

where $|\Psi \cap \Lambda|$ means the number of patterns in the benchmark that are also returned by the computational method (in short, the number of targets discovered), $|\Psi|$ means the number of patterns in the benchmark (the number of targets), and $|\Lambda|$ means the number of patterns returned by the computational method. These collections are depicted in the Venn diagram in Fig. 4.13. Comparing some computational method, A , with another computational method, B , one can say that A is an improved method for a task if the recall and precision values of A are consistently higher than those of B . Other commonly used metrics are the F_1 score (harmonic mean of recall

⁴If a computational method for pattern discovery returns one or more patterns that are not in the benchmark, but for some reason are deemed worthy of discovery, then a different evaluation framework may be required.

and precision) and average precision (Manning and Schütze, 1999).

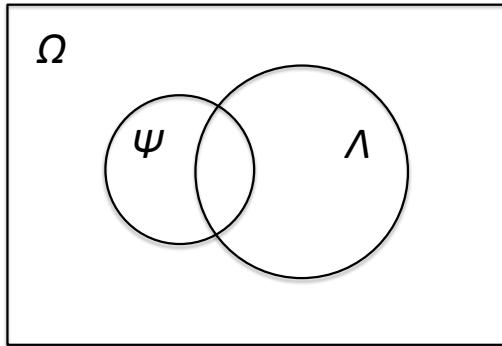


Figure 4.13: A Venn diagram to show different collections of musical patterns for a retrieval task. The collection of all patterns is denoted Ω , the benchmark of targets is denoted Ψ , and the patterns returned by a computational method are denoted Λ .

4.4 The SIA family applied beyond the musical surface

It was mentioned (p. 23) that the output of a chord labelling algorithm—specifically, HarmAn (Pardo and Birmingham, 2002)—can be represented as a dataset E , with the first dimension being the ontime of a chord label, the second being the MNN modulo 12 of the chord root, the third being an integer between 0 and 5 indicating the chord class (cf. Def. B.10), and the fourth being the duration for which the label is valid. A member of the SIA family could be applied to E , therefore, in order to discover repeated patterns in the chord dataset.

There is also a relationship between maximal translatable pattern and a formalisation of *metre* (defined loosely as hierarchical patterns of accents)

called Inner Metric Analysis (Volk, 2008). A *local metre* is defined by Volk (2008) as a set of ‘equally spaced onsets... [that] contains at least three onsets and is maximal, meaning that it is not a subset of any other subset consisting of equally distanced onsets’ (p. 261). The example given by Volk (2008) is reproduced in Fig. 4.14. The notation On is used for the projection of a dataset D on to the dimension of ontine alone. Below the staff, ontimes are shown as asterisks $*$, different local metres are indicated by dots \bullet on different rows A-S, and so-called extensions of local metres are indicated by red triangles \triangle . A local metre is denoted $m_{s,d,\kappa} = \{s + id : i = 0, 1, \dots, \kappa\}$, where s is the starting ontine, d is the period, and κ the length (number of ontimes minus one).

It can be shown (but will not be shown here) that if $m_{s,d,\kappa}$ is a local metre in the dataset of ontimes On , then there exists an interval of time u such that

$$(m_{s,d,\kappa} \setminus \{s + \kappa d\}) \subseteq \text{MTP}(u, On). \quad (4.12)$$

In words, an arbitrary local metre, with its last ontine removed, is a subset of at least one maximal translatable pattern. This relationship between maximal translatable pattern and local metre means that SIA could be used as a step in calculating the set of all local metres of length at least l in a dataset, denoted $M(l)$.

Definition 4.7. General metric weight (Volk, 2008). Let On be the projection of the dataset D on to the dimension of ontine alone, and $M(l)$ be the set of all local metres of length at least l in On . The *general metric*

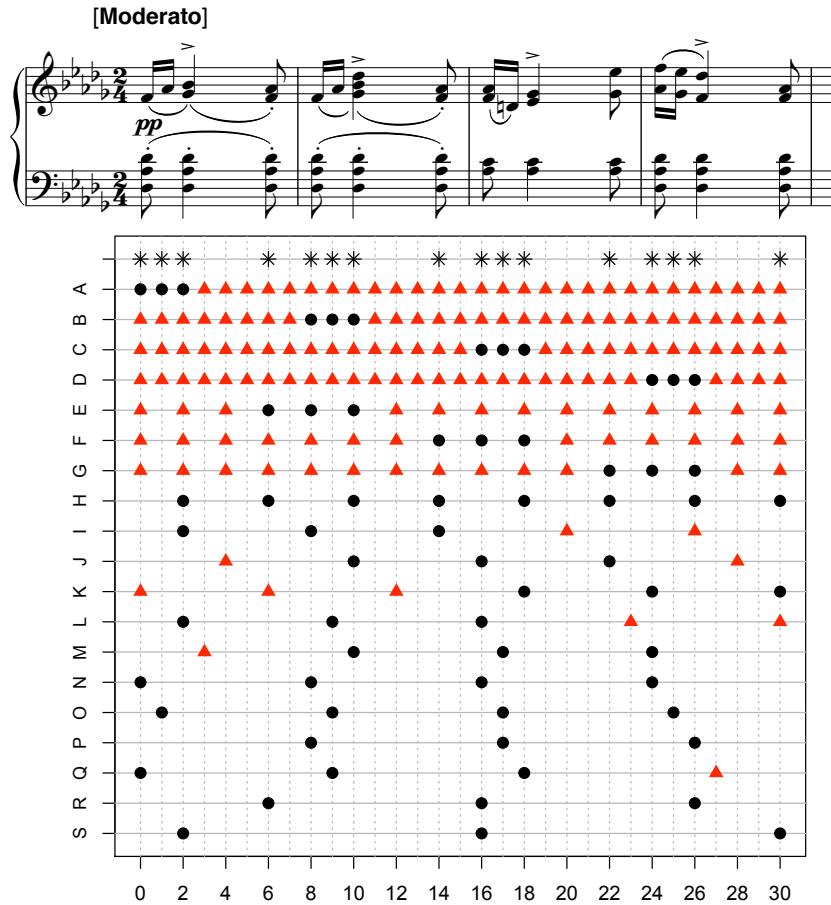


Figure 4.14: Reproduced from Volk (2008). Bars 62–65 of *Moment musical* op.94 no.4 by Schubert. Below the staff, onsets are shown as asterisks *, different local metres are indicated by dots ● on different rows A–S, and extensions of local metres are indicated by red triangles △.

weight of an onset $t \in On$ is defined by

$$W_{l,p}(t) = \sum_{\{m \in M(l): t \in m\}} \kappa^p, \quad (4.13)$$

where l and p are parameters, taken as $l = p = 2$ by Volk (2008). ■

In theory, the larger an ontime's general metric weight, the greater its metric importance. Extensions of local metres, indicated by the triangles in Fig. 4.14, lead to the definition of *spectral weight* (Volk, 2008). A plot of general metric weight against time is not unlike a plot of an empirical distribution, as in Fig. 3.2.

In summary, this chapter began with some of Schenker's (1906/1973) examples of repetition. Other types of repetition were exemplified, and five types of repetition (exact, with interpolation, transposed real, transposed tonal, and durational) were collected together and labelled the *proto-analytical class*. The task of intra-opus discovery of translational patterns was introduced, and string-based and geometric discovery methods were contrasted. Attention was then restricted to three algorithms (SIA, SIATEC, and COSIATEC) from the SIA family (Meredith et al., 2003), as the patterns that these algorithms return are most consistent with the proto-analytical class. A short but important section considered how to determine when an improved method for pattern discovery has been achieved: two metrics, called recall and precision, were defined. The final section, Sec. 4.4, addressed two ways in which discovery algorithms from the SIA family might be applied, beyond the musical surface.

Literature review: Algorithmic composition

5.1 Motivations

This chapter reviews different approaches to algorithmic composition, to situate the models for stylistic composition that are developed over Chapters 8 and 9. Algorithmic composition is a field of great variety and antiquity, which is perhaps unsurprising given the broad definitions of the terms ‘algorithm’—being ‘any well-defined computational procedure that takes some value, or set of values, as *input* and produces some value, or set of values, as *output*’ (Cormen, 2001, p. 5)—and ‘composition’—being ‘[t]he activity or process of creating music, and the product of such an activity’ (Blum, 2001, p. 186). Some aspects of composition that can be described as a process are eminently suited to being turned into algorithms. In a recent summary of algorithmic composition organised by algorithm class, Nierhaus (2009) gives examples ranging from hidden Markov models (Allan, 2002) to genetic algorithms (Gartland-Jones and Copley, 2003), and his introductory historical overview credits Guido of Arezzo (*c* 991-1033) with devising the first system for algorithmic composition.

Pearce, Meredith, and Wiggins (2002) identify four categories of motivation for automating the compositional process, reproduced in Table 5.1.

They give examples of research for each motivational category, and observe a general ‘failure to distinguish between different motivations for the development of computer programs that compose music. . . . As a consequence, researchers often fail to adopt suitable methodologies for the development and evaluation of compositional programs and this, in turn, has compromised the practical or theoretical value of their research’ (p. 120). Table 5.1 suggests that the meaning of the term ‘algorithmic composition’ as used by Pearce et al. (2002), an activity within the domain of composition, differs from the meaning of ‘algorithmic composition’ as used by Nierhaus (2009), whose book of the same name includes examples of compositional tools and computational models of musical style. I do not see this as a major conflict: Pearce et al. (2002) are emphasising that where algorithmic composition is used in the domain of musicology, it is with the specific aim in mind of modelling musical style.

Table 5.1: Reproduced from Pearce et al. (2002). Motivations for developing computer programs which compose music.

Domain	Activity	Motivation
Composition	Algorithmic composition	Expansion of compositional repertoire
Software engineering	Design of compositional tools	Development of tools for composers
Musicology	Computational modelling of musical styles	Proposal and evaluation of theories of musical styles
Cognitive science	Computational modelling of music cognition	Proposal and evaluation of cognitive theories of musical composition

Chapters 8-10 of this thesis fit best into the third category in Table 5.1, computational modelling of musical styles. As such, the scope of the lit-

erature review is limited to those methods of algorithmic composition that might reasonably be expected to be useful for modelling musical style. It is also useful to distinguish between two types of composition:

Stylistic composition. A stylistic composition (or pastiche) is a work in the style of another composer or period.

Free composition. On the other hand, free composition is something of a catchall term for any work that is not a pastiche.

An excerpt of a stylistic composition is shown in Fig. 5.1. Reich (2001) suggests that this excerpt, written by pianist and composer Clara Schuman [née Wieck] (1819-1896), was inspired by the music of Chopin. Chopin's mazurkas began to appear a decade before the piece in Fig. 5.1 was published, and Clara Schumann was among the first pianists to perform his music. The mazurka is a 'Polish folk dance from the Mazovia region [where Chopin spent his childhood]....In his [fifty plus] examples the dance became a highly stylized piece for the fashionable salon of the 19th century' (Downes, 2001, p. 189). An excerpt of a free composition from the sixteenth century is shown in Fig. 5.2. The opening chord sequence (C \sharp major, A minor, B major, G major) is so distinctive that I would be surprised to find instances of other composers using this sequence in the next three hundred or so years. Distinctiveness then, formalised to a degree by Conklin (2010), ought to be added to the definition of free composition. Often the line separating stylistic composition (or pastiche) and free composition is blurred: the excerpt shown in Fig. 5.3 contains stylistic elements associated with Joseph Haydn (1731-1809), as well as elements that situate it in Sergey Prokofiev's (1891-1953) oeuvre. A more credible stance is that most pieces are neither entirely free

nor entirely stylistic, but somewhere in between. At the former extreme, a work is so highly original and lacking in references to existing work that listeners remain perplexed, long after the premiere. No doubt the moderating influence of hindsight plays an important role here: for example, composer and critic Robert Schumann said of Chopin's Piano Sonata in B \flat minor op.35 that 'we listen as if spellbound and without complaint to the very end, yet also without praise, for *music* it is not. Thus the sonata ends as it began, puzzling' (Newman, 1969, p. 490). Arguably, listeners today are less perplexed by this sonata. Nevertheless, it is possible to recognise the historical originality of works such as in Fig. 5.2. At the latter extreme, a piece that is entirely stylistic merely replicates existing work, perhaps even note for note. A 'composer' of such a piece is unlikely to achieve positive recognition, and may even face legal penalties, or artistic/academic isolation.



Figure 5.1: Bars 1-8 of the Mazurka in G minor from *Soirées musicales* op.6 no.3 by Clara Schumann.

1

S. E chi mi può dar vi -

A. Mo - ro, las - so, al mio duo - lo, E chi mi può dar

A. Mo - ro, las - so, al mio duo - lo,

T. Mo - ro, las - so, al mio duo - lo,

B. Mo - ro, las - so, al mio duo - lo,

Figure 5.2: Bars 1-10 of 'Moro, lasso' from Madrigals book 6 by Carlo Gesualdo, Prince of Venosa, Count of Conza (c 1561-1613).

Allegro $\text{J} = 100$

(46)

pianissimo con eleganza

50

mf

ff

pp

Figure 5.3: Bars 46-54 of the first movement from the Symphony no.1 in D major, 'The Classical', op.25 by Sergey Prokofiev (1891-1953).

5.2 Example briefs in stylistic composition

Some example briefs within stylistic composition are as follows:

1. **Chorale harmonisation.** ‘Harmonise [the chorale melody shown in Fig. 5.4 in the style of Johann Sebastian Bach (1685-1750)]... by adding alto, tenor and bass parts’ (AQA, 2009, p. 3).
2. **Ground bass.** ‘Write six four-part variations for string or wind ensemble above the ground bass [shown in Fig. 5.5 by Gottfried Finger (*c* 1660-1730)]... Continuous four-part texture is not required, but some imitative and lively writing should be included’ (Cambridge University Faculty of Music, 2010a, p. 10).
3. **Fugal exposition.** ‘Write a fugal exposition in four parts, for either strings (in open score) or keyboard, on one of the five subjects given in [Fig. 5.6]’ (Cambridge University Faculty of Music, 2010a, p. 8).
4. **Classical string quartet.** ‘Candidates are expected to complete part of a movement of a string quartet [approximately forty bars]. This will allow candidates to demonstrate... the development of thematic ideas... modulation... [and] variety in texture’ (AQA, 2007, p. 21).
5. **Advanced tonal composition.** ‘Candidates are required to submit a portfolio comprising one substantial composition, which should be either an instrumental work in four movements or an extended song cycle... between thirty and forty-five minutes [in duration]... The possible types of composition include (for example) piano sonata, sonata for melody instrument and piano, song cycle for voice and piano, pi-

ano trio, string quartet, clarinet quintet, wind quintet... candidates should demonstrate a detailed understanding of an idiom appropriate to a period and place in Europe between 1820 and 1900' (Cambridge University Faculty of Music, 2010b, pp. 25-26).



Figure 5.4: A melody for harmonisation in the style of J.S. Bach (AQA, 2009). It bears a close resemblance to the hymn tune 'Wenn mein Stündlein vorhanden ist'.



Figure 5.5: A ground bass above which parts for string or wind ensemble are to be added (Cambridge University Faculty of Music, 2010a). The exact source from Finger's work is unknown.

These tasks are in the order of most to least constrained. In task 1, a relatively large number of conditions help the composer respond to the brief: the soprano part is already written, and the number of remaining parts to be composed is specified. A composer who only wrote one note per part per crotchet beat might pass this part of the exam. Supposing that each of the alto, tenor, and bass voices has an octave range, and adopting a one-note-per-part-per-beat approach, the total number of possible compositions is large



Figure 5.6: Five subjects, one to be chosen for development as a fugal exposition (Cambridge University Faculty of Music, 2010a). The exact sources of the subjects are unknown.

but finite, $(13 \cdot 3)^{58} \approx 10^{92}$, where 13 is the octave range, 3 is the number of voices to be added, and 58 is the number of crotchet beats for which material must be composed. Impose supplementary rules that allow only certain types of chords to be composed, and that prohibit the crossing of voices, say, then the total number of possible compositions is reduced considerably (though harder to enumerate). It is worth enquiring as to the origin of these supplementary rules. In this scenario, they are gleaned from the nearly four-hundred chorale harmonisations of J. S. Bach. These harmonisations contain features that complicate the one-note-per-part-per-beat approach, however, such as passing notes and suspensions, as well as exceptions to common rules that might be seen to be observed. That said, the helpful constraints inherent in chorale harmonisation have made this task popular with computational modellers of musical styles.

Allan (2002) uses hidden Markov models (HMM) to harmonise chorale melodies. An HMM consists of *hidden states*, members of the countable set I , and *observed states*, members of the countable set J . A sequence of random variables X_0, X_1, \dots takes values in I , so is called the *hidden sequence*, and another sequence of random variables Y_0, Y_1, \dots takes values in J , so is called the *observed sequence*. In general the following information is known or can be determined empirically from data that one is trying to model:

- The initial distribution $\mathbb{P}(X_0 = i_0)$. That is, the probability that X_0 will take the value $i_0 \in I$.
- The transition probabilities $\mathbb{P}(X_n = i_n \mid X_{n-1} = i_{n-1})$, where $n \geq 1$. That is, the probability that X_n will take the value $i_n \in I$, given X_{n-1} takes the value $i_{n-1} \in I$.

- The emission probabilities $\mathbb{P}(Y_n = j_n \mid X_n = i_n)$, where $n \geq 0$. That is, the probability that the observable random variable Y_n will take the value $j_n \in J$, given the hidden random variable X_n takes the value $i_n \in I$.

The conditional dependence structure shown in Fig. 5.7 is assumed. The probability that Y_n takes the value j_n is conditionally dependent on the value taken by X_n , which in turn is conditionally dependent on the value taken by X_{n-1} , but all other pairs of random variables are conditionally *independent*. The assumptions mean that no more information need be known in order to answer the following question:

Given the known or empirically determined information, and an observed sequence $Y_0 = j_0, Y_1 = j_1, \dots, Y_N = j_N$, what is the most likely hidden sequence $X_0 = i_0, X_1 = i_1, \dots, X_N = i_N$?

The Viterbi algorithm provides the solution to this question (Rabiner, 1989). Allan (2002) treats a chorale melody as the observed sequence and asks, which hidden sequence of harmonic symbols is most likely to underlie this melody? The information about melody notes and harmonic symbols (initial distribution, transition and emission probabilities) is determined empirically by analysing other chorales, referred to as the training set. In effect, the Viterbi algorithm is used to attribute harmonic symbols to the chorale melody. A second HMM is then employed by Allan (2002). ‘The harmonic symbols decided by the previous subtask will now be treated as an observation sequence, and we will generate chords as a sequence of hidden states. This model aims to ‘recover’ the fully filled-out chords for which the harmonic symbols are a shorthand’ (p. 45). A final step introduces ornamentation (e.g.,

passing notes) to what would otherwise be a one-note-per-voice-per-beat harmonisation. Hidden Markov models are appropriate for tasks within stylistic composition if an entire part is provided (such as the melody of a chorale, or bass of a ground bass), but if not, Markov models of the non-hidden variety (introduced in Sec. 3.3, p. 44) are more appropriate.

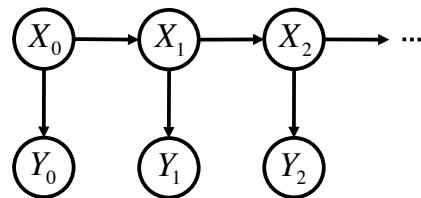


Figure 5.7: A graph showing the typical conditional dependence structure of a hidden Markov model. A sequence $Y_0 = j_0, Y_1 = j_1, \dots$ is observed. The emission probabilities $\mathbb{P}(Y_n = j_n | X_n = i_n)$, where $n \geq 0$, are known, and so are the transition probabilities $\mathbb{P}(X_n = i_n | X_{n-1} = i_{n-1})$, where $n \geq 1$. This knowledge is indicated by the arcs (arrows).

Ebcioğlu (1994) describes a system, CHORAL, also intended for the task of chorale harmonisation. A logic programming language called Backtracking Specification Language (BSL) is used to encode some 350 musical ‘rules’ that the author and other theorists observe in J.S. Bach’s chorale harmonisations, for example ‘rules that enumerate the possible ways of modulating to a new key, the constraints about the preparation and resolution of a seventh in a seventh chord, a constraint about consecutive octaves and fifths’ (Ebcioğlu, 1994, pp. 310-311). Like the HMM of Allan (2002), there are separate chord-skeleton and chord-filling steps. Unlike the HMM of Allan (2002), which consists of probability distributions learnt from a training set of chorale harmonisations, CHORAL is based on the programmer’s hand-coded rules.

This distinction between machine-learning and hand-coding of rules is important though sometimes unclear in other work (cf. the discussion in Sec. 5.6.1 about database construction). While hand-coded, rule-based systems persist (Anders and Miranda, 2010), it is questionable whether such systems alone are applicable beyond relatively restricted tasks in stylistic composition. A similar question might be asked of a neural network model for harmonising chorales called HARMONET (Hild, Feulner, and Menzel, 1992).

Stylistic composition tasks 2 and 3 are noteworthy because they demonstrate different *compositional strategies*, compared to each other and task 1. For example, in composing a four-part chorale, one begins with the soprano (top) part (either borrowing it from an existing hymn tune or creating anew) and then supplies the remaining lower parts. The concern for harmony (the identity of vertical sonorities) dominates the concern for counterpoint (the horizontal, melodic independence of individual voices). Inherent in the name ‘ground bass’ is a different compositional strategy of beginning with the bass (bottom) part, and supplying the upper parts. I am not aware of any existing systems for automated generation of material on a ground bass. Although a system proposed by Eigenfeldt and Pasquier (2010) does allow the user to specify a bass line, it is not intended to model a particular musical style. Whilst it would consume too much space to give a full description of the rules of fugal exposition, suffice it to say that the compositional strategy is different again. One voice is introduced at a time, with the subject. When the second voice enters with the answer (a transposed version of the subject), the first voice begins the countersubject. When the third voice enters with the subject, the second voice takes up the countersubject, and the first

voice is relatively free. Typically, a fugal exposition is said to have finished once the last voice has stated the subject and countersubject. Whilst some fugues begin with voices entering top to bottom, or bottom to top, there are other possibilities (Walker, 2001). In ground bass and fugal exposition, the concern for counterpoint dominates the concern for harmony. A system for generating fugal expositions is outlined by Craft and Cross (2003), and the selected output of a second system (Cope, 2002) is available.

Composers often abstract conventions from the original context of previous musical periods, and use the conventions as devices in their own work, perhaps stretching observed rules. For instance, the Romance in F major op.118 no.5 by Johannes Brahms (1833-1897) contains a ground bass (original context being the Baroque period, *c* 1600-1750). It is also feasible to switch compositional strategy midway through a piece, so that one section may favour harmonic or vertical concerns over contrapuntal or horizontal concerns, and the next section vice versa (an example is given in Fig. 8.2, p. 194).

Stylistic composition tasks 4 and 5 are relatively unconstrained. Hopefully, a composer responding to the brief of the Classical string quartet will produce material that is stylistically similar to the quartets of Haydn or Mozart, say, but there would appear to be less guidance in terms of provided parts or explicit rules. Task 5 is even more open-ended: an appropriate corpus of music, e.g. the songs of Edvard Grieg (1843-1907), must be identified and absorbed by the composer responding to this brief. For the sake of completeness, and without further explanation, here are some tasks that might be classed as free composition:

1. **Soundtrack.** ‘Compose a piece of continuous music for a promotional video to launch a new low-cost airline. You should aim to depict a range of scenes, countries and destinations in the music’ (Edexcel, 2009, p. 4).
2. **Competition test piece.** ‘Compose a competition test piece intended to exploit the playing techniques of an acoustic melody instrument of your choice. This featured instrument should be accompanied by piano or two/three other acoustic instruments’ (Edexcel, 2009, p. 3).
3. **Portfolio of free compositions.** Candidates are encouraged ‘to develop the ability to compose in a manner and style of their own choice....Candidates are required to submit a portfolio of three compositions. One of the compositions should be a setting of words, and one should include fugal elements and/or incorporate the techniques of ground bass and/or chaconne. One piece should be for orchestra (with or without voices) or ensemble of no fewer than ten players. One piece should be no shorter than eight minutes in duration. Normal staff notation will usually be expected, but electro-acoustic submissions are also acceptable’ (Cambridge University Faculty of Music, 2010b, p. 26).

Chapters 8-10 of this thesis are concerned with models that attempt to respond to the following stylistic composition brief:

Chopin mazurka. Compose the opening section (approximately sixteen bars) of a mazurka in the style of Chopin.

The hypothesis that this compositional brief is used to test in Chapter 10 is broad (concerning the application of random generation Markov chains to music from other composers/periods), so it does not make sense to go into too

much detail about the mazurka style (or to try to encode this knowledge). That said, detailed accounts of mazurka style are available (Rosen, 1995; Rink, 1992). For the purposes of Chapters 8-10, Chopin's mazurkas are an appropriate corpus because there are many (approximately fifty) from which to choose, they are relatively stylistically homogeneous, and they contain a mixture of homophonic and polyphonic textures. Evaluation of the models described in Chapter 9 will focus on the third criterion (style) given in Table 5.2. Criteria four and six (instrumentation and notation) are treated as subordinate in this instance, as excerpts for the evaluation will be presented as piano MIDI files. As the computer models do not generate an introspective written review alongside the generated passage of music, criterion seven will be overlooked. I think criteria two and five (imagination and expressivity) are too vague to be evaluated (and wonder about the level of agreement between AQA assessors for these criteria), although I do not doubt that they are important elements of a piece of music. The authorship criterion will be revisited in Secs. 5.3 and 5.4.

5.3 Early models of musical style

Prior to the twentieth century, the system closest to a model of musical style was the *musical dice game*, or *Musikalisches Würfelspiel* (Hedges, 1978; Newman, 1961). Some of the music segments from a dice game attributed to Mozart are shown in Fig. 5.8. To generate the first bar of a new minuet, the game's player rolls a die, observes the outcome $1 \leq m \leq 6$, and consults

Table 5.2: Assessment criteria for composition unit, adapted from AQA (2009).

Criterion	Description
1. Authorship	The submitted composition is the candidate's own work.
2. Imagination	The piece will be stimulating, inventive and imaginative.
3. Style	The piece demonstrates a firm grasp of, and secure handling of, compositional techniques with a clear understanding of the chosen style.
4. Instrumentation	The writing for the chosen instruments/voices will be highly idiomatic.
5. Expressivity	The expressive features of the music will be immediately apparent to the listener.
6. Notation	Notation will be accurate in relation to pitch and rhythm [of recording] and contain detailed performance directions appropriate to the music.
7. Introspection	The candidate's written review provides a detailed and accurate evaluation of the process with an extensive use of technical language.

the m th row, first column of the matrix

$$\begin{array}{c|ccccccccc} \text{Roll/Bar} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \hline 1 & \left(\begin{array}{ccccccccc} v_{1,1} & v_{1,2} & v_{1,3} & v_{1,4} & v_{1,5} & v_{1,6} & v_{1,7} & v_{1,8} \end{array} \right) \\ 2 & \left(\begin{array}{ccccccccc} v_{2,1} & v_{2,2} & v_{2,3} & v_{2,4} & v_{2,5} & v_{2,6} & v_{2,7} & v_{2,8} \end{array} \right) \\ 3 & \left(\begin{array}{ccccccccc} v_{3,1} & v_{3,2} & v_{3,3} & v_{3,4} & v_{3,5} & v_{3,6} & v_{3,7} & v_{3,8} \end{array} \right) \\ 4 & \left(\begin{array}{ccccccccc} v_{4,1} & v_{4,2} & v_{4,3} & v_{4,4} & v_{4,5} & v_{4,6} & v_{4,7} & v_{4,8} \end{array} \right) \\ 5 & \left(\begin{array}{ccccccccc} v_{5,1} & v_{5,2} & v_{5,3} & v_{5,4} & v_{5,5} & v_{5,6} & v_{5,7} & v_{5,8} \end{array} \right) \\ 6 & \left(\begin{array}{ccccccccc} v_{6,1} & v_{6,2} & v_{6,3} & v_{6,4} & v_{6,5} & v_{6,6} & v_{6,7} & v_{6,8} \end{array} \right) \end{array} = \mathbf{V}. \quad (5.1)$$

The segment in Fig. 5.8 bearing this label becomes the first bar of the new minuet. To generate the second bar, the player rolls the die again, observes the outcome $1 \leq m' \leq 6$, and consults the m' th row, second column of \mathbf{V} . The corresponding segment from Fig. 5.8 becomes the second bar of the new minuet. The process continues until eight bars have been generated. In the original dice game, the bar-length segments of music are arranged in a different order to that of Fig. 5.8, so that the equivalent harmonic function of segments in the same column and equality of segments in the eighth column are disguised. The dice game, both matrix \mathbf{V} and music segments, can be represented as a graph, shown in Fig. 5.9. Each vertex represents a segment of music, and an arc from vertex $v_{i,j}$ to $v_{k,l}$ indicates that segment $v_{i,j}$ can be followed by $v_{k,l}$ when the dice game is played. A walk from left to right is shown in black in Fig. 5.9, corresponding to one possible outcome of the dice game. Hedges (1978) suggests that publishers probably used the names of renowned composers to increase sales of the game, although this was hardly necessary in the Age of Reason (*c* 1650-1800)—a period characterised by the

rise of scientific investigation in which ‘a systematic device that would seem to make it possible for anyone to write music was practically guaranteed popularity’ (p. 185).



Figure 5.8: Bar-length segments of music to be used in combination with the matrix \mathbf{V} from (5.1). These segments and the matrix are adapted from a musical dice game attributed to Mozart, K294d.

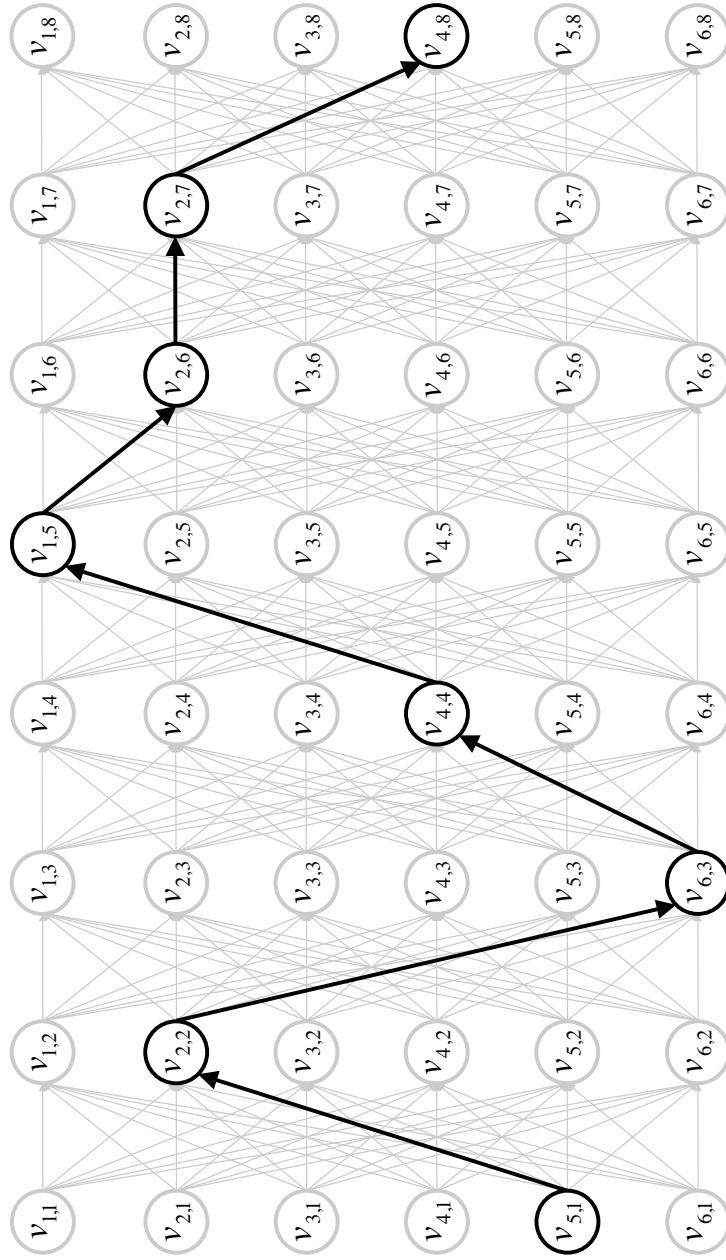


Figure 5.9: A graph with vertices that represent bar-length segments of music from Fig. 5.8. An arc from vertex $v_{i,j}$ to $v_{k,l}$ indicates that segment $v_{i,j}$ can be followed by $v_{k,l}$ when the dice game is played. A walk from left to right is shown in black, corresponding to one possible outcome of the dice game.

A thought experiment in which the musical dice game is played by me, and the resulting passage of music presented to a naïve listener, will help to refine the notion of algorithmic composition as applied to computational modelling of musical style. Suppose I generate an eight-bar passage—the new minuet—using a die, the graph in Fig. 5.9, and the segments in Fig. 5.8. When the passage is presented to the naïve listener, they say it sounds like a Classical minuet and ask who composed it. Surely the credit goes to whoever compiled the graph and the segments (Mozart or an imposter); the person responsible for what might be called *database construction*. The rolling of the die—the *generating mechanism*—influences the content of the generated passage, but has a comparatively negligible effect on the stylistic success of the resulting Classical minuet. Now suppose I encode Fig. 5.8 as MIDI files that can be appended to one another in an order determined by a path through the graph in Fig. 5.9, which in turn is determined by computer-generated random numbers. At the press of a button, I am able to generate another eight-bar passage of music. Again, the naïve listener enquires as to the composer. Is it fair to answer that ‘the passage is computer-generated’? The typical response to such an answer is amazement: ‘How could a computer create such a beautiful passage of music?’ When I explain that pre-existing bars of music are being recombined, and further that someone has selected and marshalled those bars into the database using their musical expertise, the naïve listener cannot help feeling that initially, I overstated the case.

This thought experiment demonstrates that it is all too easy to exaggerate the extent to which a passage of music is computer-generated, as well as the extent to which the process and product are creative (Boden, 1999). The

risk of exaggeration can be reduced by stating separately the extent to which database construction and generating mechanism are algorithmic. The generating mechanism of the dice game was algorithmic (albeit nondeterministic), but the database was not constructed algorithmically. In general, most models' generating mechanisms are algorithmic, but as will be discussed shortly, database construction is not always totally algorithmic.

5.4 Recurring questions of the literature survey

Before reviewing selected computational models of musical style in more detail, it is worth listing the recurring questions of this literature survey, some of which have already been encountered.

1. **Avoidance of replication.** Judging by the authorship criterion for assessment of compositions (Table 5.2), is the model's output ever too similar to works from the intended style? Does the model include any steps to avoid replicating substantial parts of existing work?
2. **Database construction.** How are the stylistic aim and corpus of music selected (for instance, Chopin mazurkas, Classical string quartets, Gesualdo madrigals)? If the model is database-driven, are both database construction and generating mechanism algorithmic, or is only the generating mechanism algorithmic?
3. **Level of disclosure.** To what extent is it possible to reproduce the output of somebody else's model, based on either their description or

published source code?

4. **Rigour and extent of evaluation.** How has the computational model of musical style been evaluated? For which different corpora (different composers, periods, compositional strategies) has the model been evaluated?

5.5 The use of viewpoints to model musical style

Conklin and Witten (1995) describe the theory behind SONG/3 (Stochastically Oriented Note Generator), a system which can be used to predict attributes of the next note in a composition, based on contextual information. Prediction may seem unrelated to algorithmic composition at first glance, but Conklin and Witten (1995) conclude with an application to composition, and this paper forms the theoretical framework for much subsequent research (Conklin, 2003; Pearce, 2005; Pearce and Wiggins, 2007; Whorley, Wiggins, Rhodes, and Pearce, 2010). An example input to SONG/3 might consist of (1) the melody in Fig. 5.4, up to and including the E \flat 5 in bar 4, (2) a collection of other chorale melodies, (3) an attribute in which I, the user, am interested in predicting, such as duration. Given this input, the output of SONG/3 would be a prediction for the duration of the note following the aforementioned E \flat 5. This prediction and the input (1)-(3) can be used to elicit successive predictions from SONG/3 if desired.

If I were asked to predict the duration of the note following E \flat 5 in bar 4 of Fig. 5.4, I could express my confidence about the various possibilities as a

distribution. Letting X be a random variable that represents the duration of the following note, letting i be a member of a countable set I of durations, and setting a crotchet equal to 1, the distribution might be:

$$\mathbb{P}(X = i) = \begin{cases} 1/2, & i = 1, \\ 1/3, & i = 2, \\ 1/6, & i = 1/2, \\ 0, & \text{otherwise.} \end{cases} \quad (5.2)$$

That is, I think there is half a chance that the next note will have a duration of 1 crotchet, a third of a chance that it will be a minim (2 crotchets), a sixth of a chance that it will be a quaver (half a crotchet), and no chance it will be any other duration. Based on this distribution for X , my prediction for the duration of the next note would be 1 crotchet, as this outcome has the largest probability.

Whereas the distribution in (5.2) is based on my intuition, Conklin and Witten (1995) use a *totally algorithmic* method for determining empirical distributions of random variables such as X . Their distributions are calculated using a combination of *viewpoints* (cf. Sec. 2.2), based on a corpus of appropriate melodies. My intuition behind the distribution in (5.2) was that ‘chorale melodies tend to have longer durations (crotchets, minims) on strong beats of the bar (beats 1 and 3 in common time)’. This intuition may or may not be correct, so what Conklin and Witten (1995) do is examine this type of relationship (between duration and beat-of-bar) empirically. Rather than assessing how often the predictions of SONG/3 are correct, the *entropy* (cf. Sec. 3.2 and Def. A.41) of distributions is considered. Recall

that a low value for entropy means that the mass of a distribution is concentrated in a few outcomes, whereas high entropy means the mass is scattered relatively evenly across many outcomes. Low entropy is associated with high confidence in a prediction, and vice versa. By modelling some attribute of a melody as a sequence of random variables, and taking the mean entropy of the corresponding distributions, it is possible to assess the average confidence in the predictions being made. Conklin and Witten (1995) use SONG/3 to generate the MIDI note numbers (MNN) of a chorale melody by selecting successive predicted (most likely) MNNs.

Systems A-D described by Pearce (2005) have the same theoretical foundation as SONG/3. Pearce (2005) suggests that one of the shortcomings of context models such as SONG/3 is ‘a danger of straying into local minima in the space of possible compositions’ (p. 180). That is, in the sample space of all possible MNN sequences of length N , sequences from certain regions of the space are very unlikely to be observed as the output of SONG/3 (local minima). A generating mechanism capable of avoiding such regions may be preferable. Pearce (2005) proposes the Metropolis-Hastings algorithm as a means of addressing this shortcoming. Whereas Conklin and Witten (1995) generate pitches successively, the Metropolis-Hastings algorithm begins with an initial generated sequence i_0, i_1, \dots, i_N . In each iteration of the algorithm, a particular index $0 \leq r \leq N$ is selected ‘at random or based on some ordering of the indices’ (Pearce, 2005, p. 184), and an alternative i'_r to i_r is considered. The alternative i'_r replaces i_r if the probability p' of the sequence $i_0, i_1, \dots, i_{r-1}, i'_r, i_{r+1}, i_{r+2}, \dots, i_N$ is greater than the probability p of the current sequence $i_0, i_1, \dots, i_{r-1}, i_r, i_{r+1}, \dots, i_N$. Otherwise, i'_r replaces i_r .

with probability p'/p . Pearce (2005) uses an existing chorale melody (other than those used to train the model) as an initial generated sequence, as suggested by Conklin (2003). It could be argued, therefore, that Pearce's (2005) method is more appropriate for generating a variation on a theme, than for the unconstrained tasks in stylistic composition (e.g., classical string quartet, advanced tonal composition, and Chopin mazurka).

5.6 Experiments in Musical Intelligence (EMI)

Although Cope (1996, 2001, 2005) has not published details of EMI to the extent that some academics would like (Pearce et al., 2002; Pedersen, 2008), he has proposed key ideas that have influenced several threads of research based on EMI. There has been relatively large demand for a more detailed explanation of the databases and programs referred to collectively as EMI, ranging in tenor from the sanguine—‘I remain an intrigued outsider, and hope and expect that over time, Dave will explain Emmy’s principles ever more lucidly’ (Hofstadter writing in Cope 2001, p. 51)—to the exasperated: ‘I have been unable to find published details (to the extent of reproducibility) of how they [the programs] work—rather, there are imprecise discussions of representations and rules, filled out with examples that sometimes give us an illusion of understanding what the mechanism does’ (Wiggins, 2008, p. 111).

A summary of the databases and programs referred to collectively as EMI is given by Hofstadter (writing in Cope 2001, pp. 44-51), who identifies *recombinancy* (segmenting and re-assembling existing pieces of music) as the

main underlying principle, as well as four related principles:

- Syntactic meshing
- Semantic meshing
- Signatures
- Templagiarism

Each of the four principles will be addressed below, and there will be cause to consider the recurring questions of the literature survey (Sec. 5.4).

5.6.1 Syntactic meshing

Most likely, the bar-length segments shown in Fig. 5.8 for the musical dice game were composed especially. Cope (1996) mentions this and other games in an historical overview, and suggests creating new collections of musical segments from existing works. In the musical dice game, the mechanism for assembling the segments is a die and the matrix \mathbf{V} from (5.1). Both the segments and the matrix are represented as a graph in Fig. 5.9. There is an arc from vertex $v_{i,j}$ to vertex $v_{k,l}$ if and only if it is possible for the segment represented by $v_{i,j}$ to be followed by the segment represented by $v_{k,l}$. When making a new musical dice game from existing works, two questions that arise are:

1. Which segments are allowed to follow which, or more formally, how should a graph analogous to that in Fig. 5.9 be defined?
2. What is an appropriate segment length (one phrase, one bar, one beat, etc.)?

An answer of ‘one bar’ to question 2 will be assumed for the meantime, in order to focus on question 1. Suppose I segment the excerpts shown in Figs. 5.1 and 5.10, so that the anacrusis of Fig. 5.1 is labelled $v_{1,0}$, and bars 1-8 are labelled $v_{1,1}, v_{1,2}, \dots, v_{1,8}$. Similarly, the anacrusis of Fig. 5.10 is labelled $v_{2,0}$, and bars 1-8 are labelled $v_{2,1}, v_{2,2}, \dots, v_{2,8}$. Initially, the graph for the new dice game, shown in Fig. 5.11A, is somewhat restricted. If an odd number is rolled, the result of the dice game is the excerpt from Fig. 5.1, note for note. If an even number is rolled, the result is the excerpt from Fig. 5.10, again note for note. The game is uninteresting because in the graph there are no arcs connecting previously unconnected vertices (representing segments of music). Calling on my musical expertise, I decide that some extra arcs as shown in Fig. 5.11B will not lead to any music-stylistic incongruities. Suddenly, the total number of pieces that might result from the dice game jumps from two to twenty-four. This total is much less than the 8^6 pieces that might result from the dice game in Sec. 5.3, but with a few more existing works included (giving extra vertices), and carefully chosen arcs added to the graph, the total number of pieces grows exponentially.



Figure 5.10: Bars 1-8 of the Mazurka in G minor op.67 no.2 by Chopin.

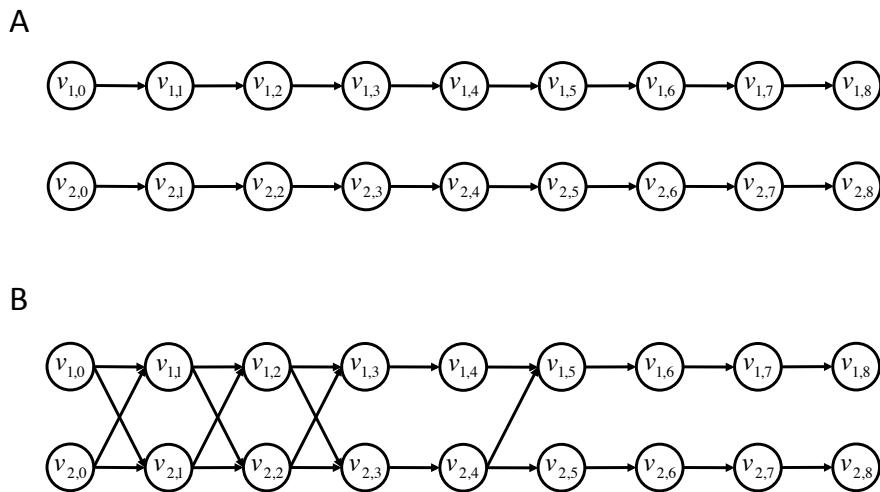


Figure 5.11: Graphs for new dice games based on segments from Figs. 5.1 and 5.10. Bars 1-8 from Fig. 5.1 are represented by the vertices $v_{1,1}, v_{1,2}, \dots, v_{1,8}$, with the anacrusis being represented by $v_{1,0}$. Bars 1-8 from Fig. 5.10 are represented by the vertices $v_{2,1}, v_{2,2}, \dots, v_{2,8}$, with the anacrusis being represented by $v_{2,0}$. (A) A somewhat restricted graph, giving only two possible outcomes to the dice game; (B) New arcs connect previously unconnected verticies, and the total number of outcomes to this dice game is twenty-four.

Syntactic meshing is the process of creating new connections between previously unconnected segments of music. In Fig. 5.11B, new arcs linked vertices $v_{1,j}$ and $v_{2,j+1}$ when, according to musical expertise, there was an equivalency of voice-leading and texture between music segments $v_{1,j}$ and $v_{2,j}$. It would seem that this was how early databases in the EMI collection were constructed (Cope, 1996, p. 136). So the above dice game and such versions of EMI have an algorithmic generating mechanism, but the database construction is not algorithmic. It is acceptable to rely on musical expertise to construct the database (that is, the segments and graph), but one ought not to claim that the output of such a model is computer-generated.

Might it be possible to distil the musical expertise? Can an algorithm be defined that takes segments of existing music as input, determines which segments are allowed to follow which, and returns a graph such as in Fig. 5.11B as output? Then the database construction would be algorithmic. At the core of such an algorithm is a function that takes two segments of music $v_{i,j}$ and $v_{k,j}$ as its arguments, returns ‘true’ if the voice-leading and texture of the two segments are equivalent, and ‘false’ otherwise. Cope (2005) suggests that beat-length segments, rather than bar-length segments, can be used to model the Bach chorale style, and is explicit about the function for determining equivalence of voice-leading and texture: ‘gather these beat groupings into collections of identically voiced beat groupings called lexicons, delineated by the pitches and registers of their entering voices (e.g., C1-G1-C2-C3, [where ‘middle C’ is C2]). . . . To compose, then, this program simply chooses the first beat of any chorale in its database, examines this beat’s voice destination notes, and then selects one of the stored beats with those same

first notes from the appropriate lexicon, assuming enough chorale data has been stored to make more choices than the original following beat grouping possible' (p. 89). In other words, the function for determining equivalence of voice-leading and texture relies on pitch. If two beat segments $v_{i,j}$ and $v_{k,l}$ begin with the same pitches, then the function returns 'true', and 'false' otherwise. Thus, the database construction is algorithmic.

If all of J.S. Bach's chorale settings are transposed to the same major key (or its relative minor), there are likely to be several instances of each beat segment, giving many new connections between previously unconnected segments. It is possible, however, to further increase the number of new connections in a corpus by *clarification* (Cope, 1996, pp. 60-63). Clarification includes (but is not limited to) removal of ornamental figuration. For example, returning to the mazurka excerpt in Fig. 5.10, the acciaccatura C5 on beat 1 of bar 6 would be ignored. The result is that the downbeats of bars 2 and 6 in Fig. 5.10 are now equivalent, and an extra new connection can be considered between $v_{2,2}$ and $v_{2,6}$ in Fig. 5.11B. In my opinion, clarification is justifiable, but as it is not explained exhaustively, the database construction reverts to non-algorithmic status. It also seems that clarification includes ignoring differences between major and minor chords, which is contradictory to an equivalence function that uses pitch, as above. Cope (2001, p. viii) cites different components of EMI (historically and in terms of compositional strategy) as the source of apparent contradictions.

5.6.2 Semantic meshing

Whereas *syntactic meshing* involves consideration of voice-leading and texture, *semantic meshing* in EMI is achieved by SPEAC analysis (Cope, 2005,

pp. 221-243), standing for statement, preparation, extension, antecedent, and consequent. The idea for SPEAC derives from Schenker (1935/1979). SPEAC analysis begins by selecting an existing work (or excerpt thereof). Each beat is given a label ('S', 'P', 'E', 'A', or 'C') and then these are combined to form labels at successively higher levels, corresponding roughly to bar, phrase, section, until a whole piece (or excerpt) is represented by a single letter. Following the guidelines set out by Cope (1996, 2005) where possible, a SPEAC analysis for an excerpt is shown in Fig. 5.12. The guidelines for assignment of labels at the beat level (level 1 in Fig. 5.12) differ from one account based on the scale degrees present (Cope, 1996, p. 68) to another account involving calculation of four types of tension (Cope, 2005, p. 227-237), so I assigned labels using personal experience and judgement. Due to the regular and consistent phrase marks in Bach chorales (denoted by pause marks, cf. Fig. 5.4), Cope (2005, p. 235) is able to jump straight from beat- to phrase-level when assigning labels at level 2. Often phrase lengths in Chopin mazurkas are unbalanced (as in Fig. 5.12), and some notes, as well as longer passages, are left unmarked. So rather than jumping straight from beat- to phrase-level when assigning labels at level 2, I consulted a list of permissible label combinations (Cope, 2005, pp. 237-238). For instance, it appears that 'SEA' at level n can become 'A' at level $n + 1$. According to this list, there is more than one 'correct' labelling for level 2, and it is unclear whether a single letter at level n can remain so at level $n + 1$, or whether it must be combined with some other. In order to assign labels at levels 4 and 5, some unpermitted label combinations were necessary: 'PP' became 'E', and 'SSS' became 'S', indicated by the dashed lines in Fig. 5.12.

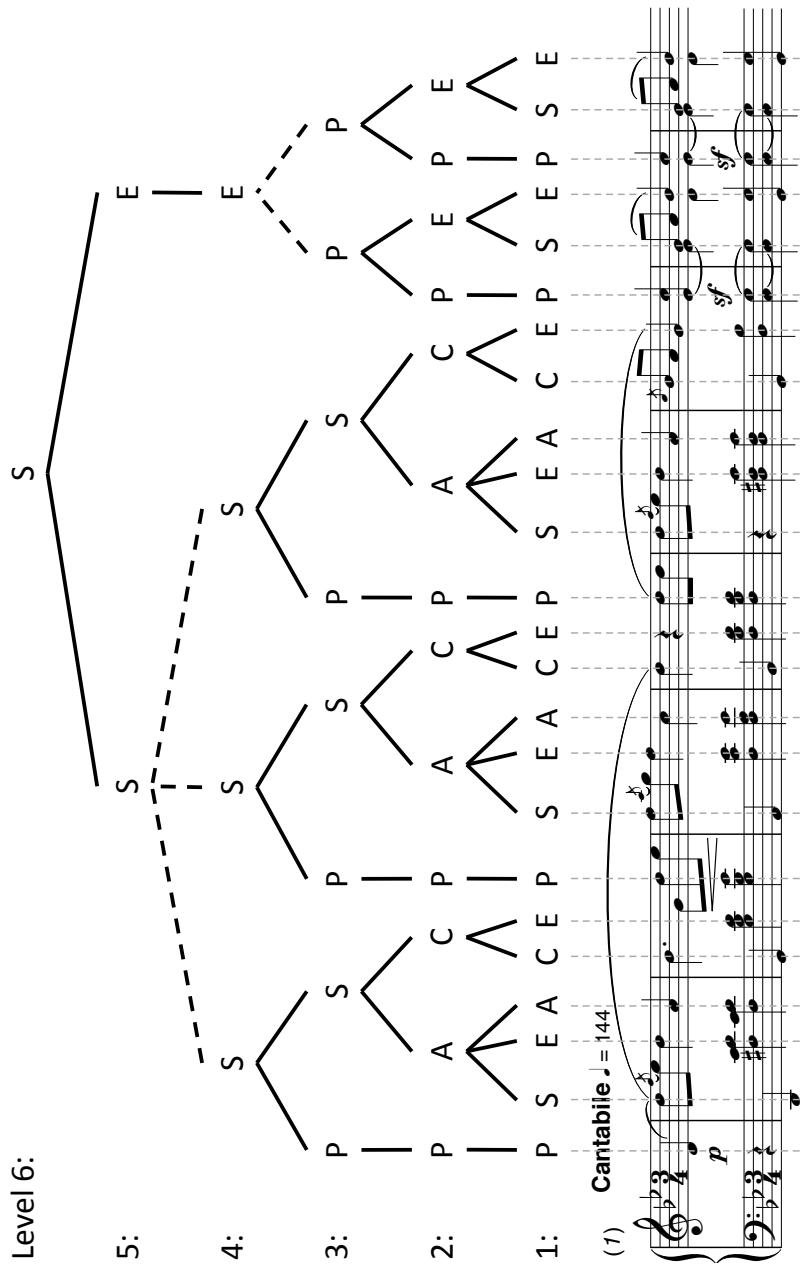


Figure 5.12: Bars 1-8 of the Mazurka in G minor op.67 no.2 by Chopin, annotated with my SPEAC analysis, standing for statement, preparation, extension, antecedent, and consequent (Cope, 1996, 2005). Each beat is given a label ('S', 'P', 'E', 'A', or 'C') and then these are combined to form labels at successively higher levels, until the whole excerpt is represented by a single letter.

Labelling issues aside, the outcome of SPEAC analysis is that each beat of the framework excerpt has an associated SPEAC string, which can be read from the bottom to the top of the hierarchy. For example, taking the excerpt in Fig. 5.12 as our framework, the upbeat to bar 1 has the SPEAC string ‘PPPSSS’, beat 1 of bar 1 has the SPEAC string ‘SASSSS’, etc. This string describes a beat’s (and its constituent notes’) location within a larger musical hierarchy. If this hierarchy—but not the actual notes spawning it—is used to guide the generation of a new passage, then perhaps the generated passage will retain the semantic validity of the framework excerpt. As well as conducting a SPEAC analysis of an appropriate framework piece/excerpt, SPEAC-analysing the other pieces in the corpus is a necessary precursor to semantic meshing. Supposing the initial pitches on a particular beat are being stored in an EMI database as the basis for syntactic meshing, then the SPEAC string corresponding to that beat segment will be stored alongside. That is, for some arbitrary beat segment $v_{i,j}$ from the corpus, its initial pitches and its SPEAC string are known, as well as a list of arcs that connect this segment to others from the corpus. This list, referred to as the *destination list*, consists of other segments $v_{k_1,l_1}, v_{k_2,l_2}, \dots, v_{k_m,l_m}$.

According to Hofstadter (writing in Cope 2001, pp. 46-48), semantic meshing is used within the generating mechanism as follows. Among all initial beat segments in the corpus (those segments that come from the beginning of a piece), attention is restricted to those that have the same SPEAC string as the initial beat segment of the framework excerpt, ‘PPPSSS’ in our example. If there is more than one such segment, one is chosen at random and becomes the first segment of the generated passage. If there are no such seg-

ments, the most global letter of the SPEAC string is removed (so ‘PPPSSS’ becomes ‘PPPSS’) and attention is restricted to those beat segments that carry the latter label. Further letters are removed until there is a choice for the first segment of the generated passage. Supposing $v_{i,1}$ is chosen as the first beat segment of the generated passage, and that $L = (v_{k_1,l_1}, v_{k_2,l_2}, \dots, v_{k_m,l_m})$ is its destination list. Among the beat segments in the destination list L , attention is restricted to those that have the same SPEAC string as the second beat segment of the framework excerpt, ‘SASSSS’ in our example. As before, if there is more than one such segment, one is chosen at random to become the second segment of the generated passage. If not, the process of shortening the label to ‘SASSS’, ‘SASS’, etc. is followed in order to find candidate destinations. Generation continues until the passage has as many beats as the framework excerpt.

This description raises many questions. For instance, semantic meshing is subservient to syntactic meshing (voice-leading and texture matches are ensured and then the best possible SPEAC-string match is sought), but what would be the consequences of inverting this relationship in the generation process? One particularly important question: is the piece/excerpt being used as a framework omitted from the database? Suppose the corpus comprises four Chopin mazurkas, op.68 nos.1-4, and one piece, op.68 no.4, is to be used as a framework. Is the database (or graph) stipulating which segments can follow which constructed over all four pieces, or just op.68 nos.1-3? If the framework piece is not omitted, then the likelihood that the generated passage replicates the framework piece note for note is increased. Several comments (Cope, 2005) suggest that in EMI, the framework piece is

not omitted: the ‘new music develops and releases tension in ways similar to one of the models in the database. . . . The fundamental structure of a new work is inherited from an analyzed work in the database’ (p. 237). This is one possible reason why some of EMI’s output replicates substantial parts of existing pieces. An example is given in Figs. 5.13 and 5.14. The black noteheads in these figures indicate notes that the EMI mazurka and original Chopin mazurka have in common. Furthermore, bars 25-26 of Fig. 5.13 are an exact copy of bars 41-42 of the Mazurka in F minor op.7 no.3 by Chopin. A more detailed analysis of EMI’s output is required to determine whether such substantial replication is the exception or the norm. Current essays on EMI (contributors to Cope 2001) are of a general nature and claim—rather than demonstrate—deep engagement with EMI’s output and the corresponding original corpus: ‘I know all of the Chopin mazurkas well, and yet in many cases, I cannot pinpoint where the fragments of Emmy’s mazurkas are coming from. It is too blurry, because the breakdown is too fine to allow easy traceability’ (Hofstadter writing in Cope 2001, pp. 297-298).

With the principles of *signatures* and *templagiarism* still to be introduced, it is already possible to address two of the recurring questions of my literature survey (Sec. 5.4)—those relating to avoidance of replication and database construction. Figures 5.13 and 5.14 show that on occasion, the databases and programs comprising EMI generate passages that are *too similar* to works from the intended style. If the excerpt/piece selected for use as a framework is not omitted from the database (i.e., the graph stipulating which segments can follows which), then omission should take place to reduce the probability of replicating substantial parts of existing work. With reference

The image shows a musical score for a piano piece, specifically Mazurka no.4 in E minor by David Cope. The score consists of six staves of music, each representing a different voice or layer. The voices are: Treble, Bass, Treble, Bass, Treble, and Bass. The music is in common time and is transposed up a minor second to F minor. The score is divided into measures by vertical bar lines. Measure numbers are indicated above the staff: 1, 6, 11, 16, 21, and 25. The notes are represented by black and white heads, with black heads indicating specific notes from Chopin's Mazurka in F minor op.68 no.4. The piano keys are also shown below the bass staff.

Figure 5.13: Bars 1-28 of the Mazurka no.4 in E minor by David Cope with Experiments in Musical Intelligence. Transposed up a minor second to F minor to aid comparison with Fig. 5.14. The black noteheads indicate that a note with the same ontine and pitch occurs in Chopin's Mazurka in F minor op.68 no.4.



Figure 5.14: Bars 1-28 of the Mazurka in F minor op.68 no.4 by Chopin. Dynamic and other expressive markings have been removed from this figure to aid clarity. The black noteheads indicate that a note with the same ontine and pitch occurs in EMI's Mazurka no.4 in E minor (Fig. 5.13).

to the second recurring question about database construction, it is unclear whether this component of EMI is totally or partially algorithmic. Recalling the thought experiment from Sec. 5.3, unless database construction is algorithmic, to call the output of a model computer-generated is to overstate the case. An unambiguous statement of the segment length and the function(s) used by Cope (1996, 2001, 2005) to determine when two segments of music are equivalent would lend weight to the argument that database construction is *totally*, not *partially*, algorithmic. Arguably, the research value of partially algorithmic processes is limited by the extent to which each decision relying on musical expertise is logged and fully explained.

5.6.3 Signatures

If a student presented the excerpt shown in Fig. 5.13 to me as their own work, their answer to the Chopin mazurka brief stated in Sec. 5.1 (p. 94), I would point out with reference to Fig. 5.14 that it was not their own work, and as such it fails the first assessment criterion in Table 5.2. If I found or was shown other examples bearing the same degree of resemblance to Fig. 5.14 as borne by Fig. 5.13, from elsewhere in Chopin's oeuvre or that of another composer, I would change my opinion: bars 1-21 of Chopin's op.68 no.4 would no longer be specific to a single piece, but a general indicator of Chopin's (or the period's) style. This is the intuition behind *signatures*, ‘contiguous note patterns that recur in two or more works of a composer and that serve in some way to characterize this composer’s musical style. Signatures typically extend over one to three measures, and often consist of a combination of melody, harmony, and rhythm’ (Cope, 2005, p. 95). Having reviewed the

discovery of patterns in music in Chapter 4, I am wary of the vague definition of signatures and of ‘controllers’ that ‘allow variations of patterns to count as matches’ (Cope, 2001, p. 111). Whereas Cope (1996) claims that determination of a corpus’ signatures is algorithmic in later versions of EMI—‘One more recent version of EMI incorporates a reflexive pattern matcher that identifies signatures without user input’ (p. 218)—Wiggins (2008) questions the efficacy and scalability of a published pattern matcher: ‘Examination of the implementation shows that all this program does is compare the pieces notewise; it’s not surprising, therefore, that when run on large pieces, or large databases of pieces, it becomes far too slow: one is forced to restrict the maximum length of the sequence. As far as I can tell from the undocumented code, gaps in allusions are not allowed, so that under this definition, variations on a theme would not count as ‘allusions’’ (pp. 112-113).

In order to explain the effect of signatures on the database construction and generating mechanism of EMI, I will overlook these claims and counter-claims, and assume that (1) signatures are well-defined, (2) they can be identified algorithmically across a corpus. Let us suppose that the algorithm for signature identification is applied to a corpus of music, and among its output is a signature corresponding to segments of music labelled $v_{i,j}, v_{i,j+1}, \dots, v_{i,m}$. Taken together, segments $v_{i,j}, v_{i,j+1}, \dots, v_{i,m}$ are several beats or bars that constitute one occurrence of the signature. In the database, which I have been representing as a graph (such as in Figs. 5.9 and 5.11), each connection from $v_{i,j}$ to $v_{k,l}$ is removed, where $k \neq i$ and $l \neq j + 1$. This means that segment $v_{i,j}$ must be followed by segment $v_{i,j+1}$. Similarly, each connection between $v_{i,j+1}$ and $v_{k,l}$ is removed, where $k \neq i$ and $l \neq j + 2$. And so on for

$v_{i,j+2}, v_{i,j+3}, \dots, v_{i,m-1}$. The last segment of the signature, $v_{i,m}$, is permitted to remain connected to segments other than $v_{i,m+1}$. Overall, the effect is that signatures are protected ‘from being fragmented into smaller groupings, thus ensuring that these signatures will survive the recombination process [syntactic and semantic meshing]’ (Cope, 2005, p. 97).

5.6.4 Templagiarism

Templagiarism is a term coined by Hofstadter (writing in Cope 2001, p. 49), to describe borrowing from an existing piece/excerpt on an abstract or template level. Suppose that in the piece selected for use as a framework, bars 1-4 are repeated at bars 9-12, and then again at bars 63-66. There may be further elements of repetition in the framework piece (including transposed or inexact repetition of bars 1-4, and repetition of other motives), but for the sake of simplicity, focus is restricted to bars 1-4, labelled A_1 , and the two subsequent occurrences of A_1 , labelled A_2 and A_3 respectively. The positions in terms of temporal and pitch displacement—but not the actual notes—of these occurrences are recorded and used to guide EMI’s generating mechanism. For instance, material is generated for bars 1-4 first, and then ‘copied and pasted’ to bars 9-12, and bars 63-66. Now material for intervening bars, bars 5-8 and 13-62, is generated, as well as from bars 67-90, the length of the framework piece, say. Thus, the generated passage contains a collection of notes in bars 1-4, which I label B_1 , and this collection repeats at bars 9-12 (label B_2) and 63-66 (label B_3). The collections A_1 and B_1 may not share a note in common, but on the more abstract level of relative temporal and pitch displacement, the sets $\{A_1, A_2, A_3\}$ and $\{B_1, B_2, B_3\}$ are equiva-

lent. ‘[I]n order to quote the template, you need to supplement it with a new “low-level” ingredient—a new motive—and so the quotation, though exact on the *template* level, sounds truly novel on the *note* level, even if one is intimately familiar with the input piece from which the template was drawn’ (Hofstadter writing in Cope 2001, p. 50). An explanation of templagiarism is conspicuous by its absence in Cope (1996, 2001, 2005), although there are passing references (Cope 2001, p. 175; Cope 2005, p. 245). With only Hofstadter’s description (cited above) on which to rely, my own explanation may be inaccurate.

While critical of the type of borrowing shown between Figs. 5.13 and 5.14, I see templagiarism as an important component of stylistic composition. The caveats are that a slightly more flexible approach would be preferable (e.g., do the temporal and pitch displacements between A_1 , A_2 , and A_3 have to be retained exactly by B_1 , B_2 , and B_3 ?), that borrowed patterns ought to be over one bar, say, in duration (last ontine minus first ontine), and that the framework piece ought to be omitted from the database to reduce the probability of note-for-note replication (as argued in Sec. 5.6.2, p. 114). To write a sonata-form movement, for instance, the first *group* (or theme) must be stated in the exposition (perhaps more than once), and restated in the recapitulation, much like A_1 and A_3 in the above example, with A_2 being the extra statement in the exposition. There are other aspects to a sonata form, but templagairising A_1 , A_2 , and A_3 is one way to begin such a stylistic composition (Czerny, 1848). Even if the listener/analyst/assessor notices that the bar numbers of the repetitions coincide with those of the first movement of the String Quartet in E♭ major, ‘The Joke’, op.33 no.2 by

Haydn—as they do in the case of A_1 , A_2 , and A_3 —there can be little cause for rebuke, as the borrowing is on such an abstract level.

5.7 Issues of evaluation

At the beginning of this chapter, I mentioned Pearce et al.'s (2002) observation that generally there was inadequate evaluation of systems for automated composition. Nearly a decade on, the situation is changing: a recent review of evaluation of systems for algorithmic composition (Ariza, 2009) and a framework for investigating computational creativity (Wiggins, 2006) are indicative of increased concern for matters of evaluation. In the first of two paradigmatic listening experiments (Pearce and Wiggins, 2001), participants were asked to distinguish between human-composed and computer-generated drum loops. In the second experiment (Pearce and Wiggins, 2007), participants were asked to rate excerpts in terms of *stylistic success* as a chorale melody, blind to the source of the excerpt (human or computer).

The second of these listening experiments (Pearce and Wiggins, 2007) is based on Amabile's (1996) Consensual Assessment Technique (CAT). The CAT is designed to evaluate the *creativity* of a set of artistic products. For example, in an early study Amabile (1982) asked girls aged 7-11 to produce artistic designs by glueing shapes on paper. These artistic designs (products) were later shown to judges who, working individually, rated the creativity of the products on a five-point scale, employing their own definition of creativity. Judgements along other dimensions, such as aesthetic appeal and neatness, were also elicited. Analysis of results for the CAT focuses on interjudge reliability, for creativity ratings and other rated dimensions: ‘these studies

have shown that it is possible to obtain high levels of agreement in subjective judgements of creativity, even when the judges are working independently and have not been trained to agree in any way' (Amabile, 1996, p. 60). The CAT has been applied to music by Hickey (2001), who investigated judgements of the creativity of children's compositions.

When using the CAT to evaluate their models of music composition, Pearce and Wiggins (2007) make some fundamental changes. First, the products consist of chorale melodies and computer-generated melodies that are meant to be in the chorale style. Second, ratings of stylistic success are elicited from judges, not creativity. Pearce and Wiggins (2007) show that the stylistic success ratings for melodies from different computer models are significantly lower than the stylistic success ratings for actual chorale melodies. As such, the CAT is being used to evaluate models of composition. In an effort to address the shortcomings of their models, Pearce and Wiggins (2007) quantify certain musical attributes, such as the *pitch centre* of a melody, and regress these attributes against mean stylistic success. Those attributes that emerge from the regression with significant negative coefficients might be responsible for reducing the mean stylistic success of a melody. Therefore, altering the melody-generating model to take (more) account of these attributes is one possible source of future improvements.

One criticism of the CAT—in both its original and adapted formats—is that the judges are not incentivised to think when rating the stylistic success (or any other dimension) of a product. A judge could rate an excerpt of music without thinking whether it conforms to the stylistic traits of the intended style. The earlier framework for evaluating computer-generated

music (Pearce and Wiggins, 2001) incentivised judges to a greater degree, by challenging them to distinguish between human-composed and computer-generated music. The disadvantage of distinguishing alone is that the resultant data is less rich: a judge might decide that an excerpt is computer-generated, but it is not possible to tell why, or whether the judge thought the excerpt a stylistic success (Moffat and Kelly, 2006).

In summary, this chapter began with definitions of algorithm and composition. Five example briefs in stylistic composition were given, leading to a discussion of existing algorithmic approaches to some of these tasks. I selected the brief of composing the opening section of a maurka in the style of Chopin as being an appropriate test for the computational models of musical style developed in Chapters 8 and 9. Three important contributions to computational models of musical style—the musical dice game (as described by Hedges, 1978), SONG/3 (Conklin and Witten, 1995), and Experiments in Musical Intelligence (Cope, 1996, 2001, 2005)—were reviewed in some detail, in relation to recurring questions about avoidance of replication, database construction, level of disclosure, and rigour/extent of evaluation. Finally, I have given some indication of appropriate frameworks for evaluating computational models of musical style.

6

Rating discovered patterns: An improved method

This chapter describes an experiment that attempts to answer the following question. Given some information about a discovered pattern (such as the number of times it occurs in a piece), is it possible to predict the extent to which it will be perceived as important by a listener? While it is legitimate to distinguish as Cross (1998) does between music analysis (a largely conscious and voluntary process undertaken by experts) and what might be called ordinary listening (mostly unconscious and involuntary—what tends to be studied in music psychology), both analysis and ordinary listening involve the discovery of patterns, and there is clearly some common ground between them. My aim is to explore how this pattern discovery process works. I asked students to rate already-discovered patterns, according to which patterns they would give priority to mentioning in an analysis essay. Attributes of these patterns and the excerpts in which they occur were quantified, and inferences made of the form

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p, \quad (6.1)$$

where y is the rating given to a pattern, x_1, x_2, \dots, x_p are attributes of the

pattern and excerpt, and $\alpha, \beta_1, \beta_2, \dots, \beta_p$ are coefficients to be estimated from the data (cf. Example A.47, p. 319). I am not suggesting for a moment that music analysts operate according to a formula such as (6.1), or that music analysis would benefit from doing so. Rather, I am claiming that some aspect of the pattern discovery process could be modelled by a weighted sum of pattern attributes. It is hoped that testing this claim will shed some light on how both ordinary listening and expert analysis work, and might therefore be of interest to music psychologists.

A second field where this work can have an impact is music information retrieval (MIR). Indeed, many of the pattern attributes x_1, x_2, \dots, x_p from (6.1) that are considered below come from this domain. In MIR there is an unsolved problem of how to order (and even discard some of) the output of a pattern discovery system (cf. Fig. 4.9). Figures 4.8 and 4.9 (the former a framework for the task of pattern matching) should be contrasted because in pattern matching, there is an obvious way of ordering the output matches: *rate them by relevance or proximity to the original query*, using an appropriate relevance metric. With pattern discovery, it seems less obvious how the analogous step should work. Suppose that an algorithm has discovered hundreds of patterns within a piece of music. Now these must be presented to the user, but in what order? Unlike with pattern matching, there is no original query to compare with discovered patterns.

Researchers have addressed this unsolved problem by defining various concepts and formulae. Some of these will be presented in Sec. 6.1, some are deferred to Appendix B, and I introduce a few now. To my knowledge, none of these formulae were derived empirically, and only two (Eerola and North,

2000) have been validated empirically. Hence, statistically derived models of the form (6.1) would constitute a methodological improvement.

Meredith et al. (2003) define the concepts of *coverage*, *compactness* and *compression ratio* and combine them in a multiplicative formula. Forth and Wiggins (2009) also combine them multiplicatively. It is claimed that these measures help to identify ‘perceptually salient patterns’ (Forth and Wiggins, 2009, p. 44) that would be ‘considered musically interesting by an analyst or expert listener’ (Meredith et al., 2003, p. 7). Conklin and Bergeron (2008) put forward a formula for the *interest* of a discovered pattern, and Conklin and Anagnostopoulou (2001) define something similar called a pattern’s *score*. Both of these formulae are based on the concept of the number of times one *expects* to hear/see a given pattern in a piece or excerpt. There is an analogy to be made here with bioinformatics, in terms of the expected number of occurrences of a subsequence in a DNA string (Ewens and Grant, 2001, pp. 166-177). There is also the related concept of a *distinctive* pattern (Huron, 2001b; Conklin, 2008, 2010). Cambouropoulos (2006) defines a formula for the *prominence* of a discovered pattern. ‘The patterns that score highest should be the most significant’ (Cambouropoulos, 2006, p. 254). Only fifteen patterns are discovered in the example provided by Lartillot (2004), so it hardly seems necessary to rate them. Consequently no formula is suggested, which is a shame since this is the only research that claims explicitly to be founded on ‘modeling of listening strategies’ (Lartillot, 2004, p. 53).

In summary, I focus on five kinds of repetition that are labelled collectively as the proto-analytical class (cf. Def. 4.1). In the experiment, analysis students were asked to rate already-discovered patterns, according to which

patterns they would give priority to mentioning in a music analysis essay. The model in (6.1) gives the general form of inference to be drawn from these ratings. The primary contribution of my experiment is that it tests the conjecture that some aspect of the pattern discovery process can be modelled by a weighted sum of pattern attributes. As such, it should shed some light on how both ordinary listening and expert analysis work, and therefore be of interest to music psychologists. The work is also relevant to MIR and the unsolved problem of how to arrange the output of a pattern discovery system.

6.1 Method

6.1.1 Participants and instructions

Music undergraduates (7 males and 5 females) from the University of Cambridge were paid £10 for an hour for their time, during which they were asked to rate already-discovered patterns.¹ Participants were returning for their second or third year (mean age = 20.83 years, $SD = 0.95$) and had attended music analytical lectures and written music analytical essays as part of their studies. The instructions began by alluding to these essays, and the preparatory work of identifying recurring patterns—the restatement of material, the appearance of themes, motifs, gestures. Then the main task was set out:

‘In the following exercise such recurring patterns *have been*

¹The accompanying CD (or <http://www.tomcollinsresearch.net>) includes a copy of the instructions for participants, as well as the music stimuli used in the study.

identified and will be presented for you to *rate* according to how noticeable and/or important you think they are.

- High ratings should be given to the most noticeable and/or important patterns. Even if they might be ‘obvious’, these are the kind of patterns that deserve at least a mention in a standard analytical essay.
- Low ratings should be given to patterns that are difficult to see or hear and are of little musical importance. One would struggle to justify mentioning them in an essay.
- Middling ratings apply to any other patterns—quite important but not that noticeable, or vice versa. Something will be lacking in such patterns that prevents them receiving the highest ratings, yet they are more readily perceived than low-rating patterns.’

As there is considerable variety in the terminology used to qualify ratings, participants were invited to rate patterns according to what *they* would mention in—or omit from—an analysis essay. The term *noticeable and/or important* covers as much of the terminological variety as possible, but arguably it is not as meaningful to participants as the reference to writing an analysis essay.

Participants were asked to rate patterns on a scale of 1 to 10 (least to most noticeable and/or important), giving their ratings to one decimal place. The decimal place was helpful for distinguishing between two patterns if both received the same integer rating initially. The instructions also indicated that a noticeable pattern was not necessarily an important pattern and vice versa.

A darker font for pattern noteheads than for nonpattern material was used to identify the patterns to participants, as in Fig. 6.1. Participants had access to a digital piano and a recording (Biret, 1990) of each excerpt throughout. This arrangement was intended to be typical of the environment in which an undergraduate begins analysing a piece of music. Participants were able to ask questions of clarification at any point, they were able to revise ratings, and were assured that they were not sitting a test. They were encouraged to form responses on the basis of their musicality and not by concocting some formula.

A balanced incomplete block design was used ($v = 9$, $b = 12$, $r = 4$, $k = 3$, $\lambda = 1$). This means that $v = 9$ excerpts of music were prepared, and a different combination of $k = 3$ excerpts was given to each of the $b = 12$ participants, such that each excerpt appeared in exactly $r = 4$ combinations, and each pair of excerpts appeared in exactly $\lambda = 1$ of the combinations (Mathon and Rosa, 1996). Ten patterns per excerpt were presented, so that each participant had $3 \times 10 = 30$ patterns to rate in total. The order of presentation of excerpts, and the order of patterns within excerpts, were randomised to allow for any ordering effects. Immediately prior to this task, each participant completed the same short warm up task, rating five patterns. The warm up task was intended to help participants to familiarise themselves with the format of presentation, answer sheet, and the rating scale. It also gave them an opportunity to ask questions. The format had been tested in a pilot study and adjusted accordingly for ease of understanding and use.

Lento Pattern A,
 1st occurrence

Pattern A,
2nd occurrence

Pattern A,
3rd occurrence

10

16

Figure 6.1: Bars 1-20 from the Mazurka in G \sharp minor op.33 no.1 by Chopin. Occurrences of pattern A are indicated by black noteheads. Dynamic and other expressive markings have been removed from this and subsequent figures to aid clarity.

6.1.2 Selection of excerpts and patterns

In any study such as this, the selection of stimuli influences the results. Our excerpts were selected from Paderewski's (1953) edition of mazurkas by Chopin, using a different mazurka for each excerpt.² With an eye on appropriate material for the participants, music from nineteenth-century Europe was chosen, though students may well not have met the mazurkas before. One of the selected mazurkas (op.7 no.5) was short enough to be presented in its entirety, but for other mazurkas a substantial section was chosen, not always from the beginning. Relatively speaking, Chopin's mazurkas are texturally and stylistically homogeneous, but still rich enough to contain examples of all the types of repetition from the proto-analytical class.

Approximately half of the discovered patterns were selected by me, such as patterns *A* and *B* in Figs. 6.1 and 6.2. The remaining patterns were chosen randomly from the output of Meredith et al.'s (2002) Structure Induction Algorithm for Translational Equivalence Classes (SIATEC, cf. Def. 4.5) when applied to each excerpt, such as pattern *C* in Fig. 6.3. Participants were not told of the composer or the source of the discoveries. This method of selection (half handpicked and half chosen at random from a large set) was used because I wanted to elicit a full range of judgements, whereas an entirely handpicked set of stimuli might all be relatively noticeable and/or important. On the other hand, Cook (1987) claims that he 'can 'hear' the most preposterous analytical relationships if [he] choose[s] to' (p. 57). I felt that the inclusion of some preposterous patterns—for example, pattern *C*—

²Op.7 no.5 bars 1-20; op.24 no.1 bars 17-32; op.24 no.3 bars 1-24; op.30 no.1 bars 17-36; op.33 no.1 bars 1-20; op.33 no.4 bars 1-24; op.50 no.1 bars 25-48; op.56 no.2 bars 45-68; op.67 no.3 bars 1-16.

was necessary to see what kind of ratings they received from participants. When handpicking five of the ten patterns for an excerpt, I tried first to select noticeable/important motifs such as pattern *A*. Second, I tried to select longer sections that support Schenker's (1906/1973) notion of repetition as 'creator of form', such as (approximately) bars 5-8 in Fig. 6.1, repeating at bars 9-12. Third, an attempt was made to represent each type of repetition from the proto-analytical class (cf. Def. 4.1). Finally, on occasion a pattern was chosen (nonrandomly) from the SIATEC output for its resemblance to a handpicked pattern. Thus, I tried to avoid participants realising that half of the patterns were handpicked and half discovered algorithmically.

As discussed (p. 70), SIATEC was used in preference to other algorithms because the patterns that it returns are most consistent with the proto-analytical class. Furthermore, while some of its results 'correspond to the patterns involved in perceptually significant repetitions' (Meredith et al., 2002, p. 331), the sheer number of output patterns per excerpt means that at least some fall under the heading of Cook's 'preposterous analytical relationships'.

6.1.3 Explanatory variables

I consider linear regression models for rating discovered patterns in music, as in (6.1) and Example A.47 (p. 319). The ratings given to patterns form the response variable: the explanatory variables quantify attributes of a pattern and the excerpt in which it appears. Other common methods, such as principal component analysis or a support vector machine, do not address my specific suggestion that a formula such as (6.1) could be involved at some

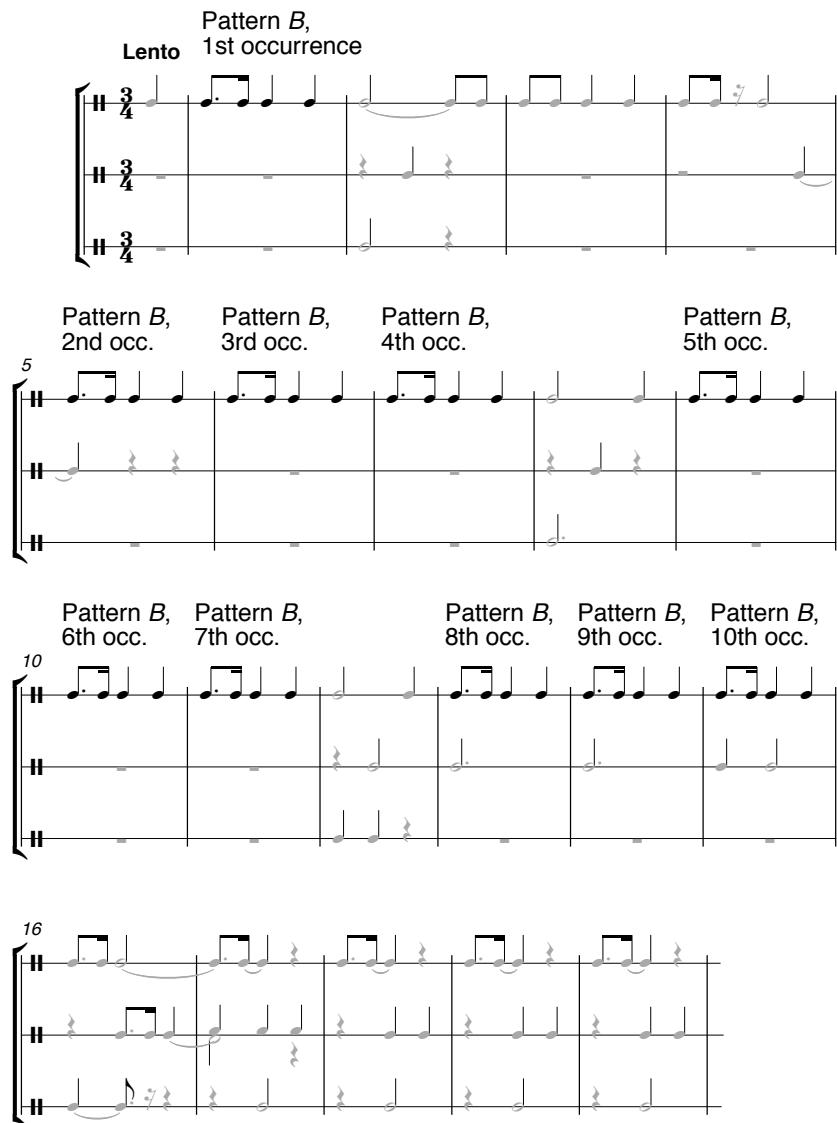


Figure 6.2: A rhythmic representation of bars 1-20 from the Mazurka in G \sharp minor op.33 no.1 by Chopin. Occurrences of pattern *B* are indicated by black noteheads.

Lento

5

Pattern C,
1st occurrence

10

Pattern C,
2nd occurrence

16

Figure 6.3: Bars 1-20 from the Mazurka in G \sharp Minor op.33 no.1 by Chopin. The first occurrence of pattern *C* contains three notes, as indicated by the bounding lines and black noteheads.

stage of the pattern discovery process. It should be recalled that linear means linear in the coefficients. That is, linear models can contain explanatory variables that are quite complex, nonlinear functions of simpler variables, and this is true of some of the pattern attributes considered below. Forward selection, backward elimination and cross-validation were used to select which explanatory variables should be used in the regression models. Eighteen of the twenty-nine explanatory variables included in the regression are formulae from existing work, and eleven variables are my suggestions. Occasionally an existing formula had to be adapted, if originally it was defined only for melodic material. Below is a list introducing the explanatory variables that emerged as being of most importance in this study. More details of these variables can be found in Appendix C, along with definitions of the remaining explanatory variables. The models that were fitted also included factors for participants and excerpts, to allow for fixed differences between participants in their ratings.

Cardinality is the number of notes contained in one occurrence of a pattern.

Occurrences refers to the number of times that a pattern occurs in an excerpt.

Coverage: The *coverage* of a pattern in a dataset is ‘the number of data-points in the dataset that are members of occurrences of the pattern’ (Meredith et al., 2003, p. 7). Recall that a *dataset* is the set of all data-points representing an excerpt of music. If no occurrences of a pattern overlap (in the sense of sharing notes) then the *coverage* of a pattern is the product of its *cardinality* and *occurrences*.

Compactness: Meredith et al. (2003) define the *compactness* of a pattern in a dataset to be ‘the ratio of the number of points in the pattern to the total number of points in the dataset that occur within the region spanned by the pattern within a particular representation’ (p. 8). There are several plausible definitions of region. I employ two and use whichever results in the maximum compactness.

Compression ratio is equal to *coverage* divided by the sum of *cardinality* and the number of nonzero translators (*occurrences* minus 1). It is the amount of compression that ‘can be achieved by representing the set of points covered by all occurrences of the pattern by specifying simply one occurrence of the pattern and all the vectors by which the pattern can be translated’ (Meredith et al., 2003, p. 8).

Expected occurrences: Conklin and Bergeron (2008) give a formula for calculating the expected number of occurrences of a pattern in a dataset. The intuition is that patterns less likely to arise by chance—because they involve less common pitches or rhythms—should be more noticeable. The calculation of *expected occurrences* involves the empirical distribution (the relative frequency of occurrence of pitches and/or other musical events in an excerpt, as discussed in Secs. 3.1-3.2). Whereas Conklin and Bergeron’s (2008) formula handles melodic material with no overlapping patterns allowed, the formula used here (and defined fully in Appendix C, p. 369) can handle textures where overlapping patterns and patterns with interpolation are allowed. Models based on relative frequency of occurrence are liable to criticism for being over-

simple,³ but I prefer to include a variable *expected occurrences* in the regression, and then assess its credentials. From the empirical distribution, it is possible to calculate the likelihood of the event that a given pattern occurs. Multiplying this likelihood by the number of places in which the pattern can occur gives the *expected occurrences*.

Interest: The interest of a pattern in a dataset is defined to be ‘the ratio of observed to expected counts’, the rationale being that ‘large differences between observed and expected counts indicate potentially interesting patterns’ (Conklin and Bergeron, 2008, p. 64).

Score: In earlier work, Conklin and Anagnostopoulou (2001) formulated essentially the same concept but in a different way, calling it the *score* of a pattern. This is the squared difference between observed and expected occurrences, divided by expected occurrences.

Rhythm only: If a pattern consists of rhythms only, then the variable *rhythm only* takes the value 1, and 0 otherwise. The intuition is that rhythm-only patterns are less noticeable than patterns that involve pitch.

Transposed repetition: The repetition of a pattern may be at the same pitch as the first occurrence, or transposed. The variable *transposed repetitions* counts the number of transposed repetitions of a pattern. Patterns with a high number of transposed repetitions could highlight real or tonal sequences, and these are likely to be noticeable/important.

³For instance, in relation to one of his models, Temperley (2007) observes that such a ‘proposal may seem wholly implausible as a model of the compositional process. But it is not intended as a model of the compositional process, only as a model of how listeners might represent the compositional process for...[a particular] purpose’ (p. 83).

6.2 Results

The explanatory variables to include in the regression were chosen by forward selection and also by backward elimination. A .05 significance level was used as the cut-off criteria for entering/removing variables. Forward selection begins with a model consisting of the participant variables (denoted by par_2 , $\text{par}_3, \dots, \text{par}_{12}$). These are protected from removal during model selection, as a blocking factor should generally be retained in models. The first step is to include each of the pattern attributes in this model, individually, and determine which of these attributes most reduces the residual sum of squares (RSS). The results of these individual fittings are shown in Table 6.1. It can be seen that *compactness* most reduces the RSS, as its value for r^2 is greatest. The coefficient for *compactness* is significant at the .05 level, so now a model is being considered that consists of the participant variables and *compactness*. The second step is similar to the first: take the new model, and include each of the remaining pattern attributes individually. To determine which attribute most reduces the RSS, a table similar in format to Table 6.1 can be constructed. The order of attributes in that table may be completely different, however, due to the effect of including *compactness*. For instance, there is no guarantee that *rhythmic density* will most reduce the RSS—in fact, *expected occurrences* is the next attribute to be appended. Variables continue to be appended in this fashion while the corresponding coefficients

are significant at the .05 level. The resulting *forward model* is

$$\begin{aligned} \text{rating} = & 4.79 + 0.01 \cdot \text{par}_2 - 1.19 \cdot \text{par}_3 - 0.96 \cdot \text{par}_4 - 0.60 \cdot \text{par}_5 \\ & - 1.18 \cdot \text{par}_6 - 0.90 \cdot \text{par}_7 - 0.27 \cdot \text{par}_8 - 0.70 \cdot \text{par}_9 - 0.62 \cdot \text{par}_{10} \\ & - 0.49 \cdot \text{par}_{11} + 0.73 \cdot \text{par}_{12} + 3.42 \cdot \text{compactness} \\ & - 0.04 \cdot \text{expected_occurrences} + 0.65 \cdot \text{compression_ratio}, \end{aligned} \tag{6.2}$$

with test statistic $F(14, 345) = 59.12$, $p < 2.2 \times 10^{-16}$, and $s = 1.67$ as the error standard deviation.

Backward elimination works in an analogous fashion. It begins with a full model, consisting of variables for participants, excerpts and pattern attributes. At each step the variable whose exclusion least increases the RSS is removed. Variables continue to be removed in this way while the corresponding coefficients are not significant at the .05 level. The *backward model* that resulted was

$$\begin{aligned} \text{rating} = & 3.88 - 0.13 \cdot \text{par}_2 - 1.28 \cdot \text{par}_3 - 0.88 \cdot \text{par}_4 - 0.74 \cdot \text{par}_5 \\ & - 1.34 \cdot \text{par}_6 - 0.97 \cdot \text{par}_7 - 0.35 \cdot \text{par}_8 - 0.78 \cdot \text{par}_9 - 0.70 \cdot \text{par}_{10} \\ & - 0.59 \cdot \text{par}_{11} + 0.68 \cdot \text{par}_{12} + 0.07 \cdot \text{cardinality} + 0.89 \cdot \text{occurrences} \\ & - 0.04 \cdot \text{coverage} + 3.39 \cdot \text{compactness} + 1.48 \cdot \text{compression_ratio} \\ & - 0.53 \cdot \text{expected_occurrences} - 0.99 \cdot \text{interest} + 0.50 \cdot \text{score} \\ & + 0.94 \cdot \text{rhythm_only} + 0.15 \cdot \text{transposed_repetitions}, \end{aligned} \tag{6.3}$$

Table 6.1: Each row in this table represents an individual fitting. For example the first row contains the results of fitting a model including the participant (block) variables *and compactness*. The standard error (s.e.) relates to the width of the confidence interval about the coefficient estimate. The ratio of explained sum of squares to total sum of squares is given by r^2 , also known as the coefficient of determination.

Variable	Coefficient	s.e.	t-value	p-value	r^2
compactness	6.12	0.26	23.87	$< 2.0 \times 10^{-16}$.63
rhythmic density	1.68	0.08	21.61	$< 2.0 \times 10^{-16}$.58
expected occurrences	-0.07	3.7×10^{-3}	-19.73	$< 2.0 \times 10^{-16}$.54
coverage	0.04	2.6×10^{-3}	14.14	$< 2.0 \times 10^{-16}$.38
threeCs	9.03	0.71	12.80	$< 2.0 \times 10^{-16}$.33
rhythmic variability	6.56	0.56	11.72	$< 2.0 \times 10^{-16}$.30
signed pitch range	0.12	0.01	11.18	$< 2.0 \times 10^{-16}$.28
prominence	7.3×10^{-3}	6.5×10^{-4}	11.16	$< 2.0 \times 10^{-16}$.28
cadential	3.50	0.32	10.89	$< 2.0 \times 10^{-16}$.27
cardinality	0.06	6.3×10^{-3}	9.96	$< 2.0 \times 10^{-16}$.24
compression ratio	1.68	0.18	9.10	$< 2.0 \times 10^{-16}$.21
alt. prominence	0.07	7.6×10^{-3}	8.61	2.7×10^{-16}	.19
phrasal	2.17	0.26	8.20	4.6×10^{-15}	.18
small intervals	0.18	0.02	8.15	6.6×10^{-15}	.18
max. pitch centre	-0.17	0.02	-6.94	1.9×10^{-11}	.14
score	-0.02	3.5×10^{-3}	-6.48	3.1×10^{-10}	.13
m.c. card. \times occ.	-0.02	2.6×10^{-3}	-6.31	8.3×10^{-10}	.12
chromatic	0.39	0.06	6.07	3.3×10^{-9}	.11
metric syncopation	2.59	0.51	5.13	4.9×10^{-7}	.09
transposed repetition	-0.29	0.07	-4.21	3.3×10^{-5}	.07
unsigned pitch range	0.07	0.02	3.35	9.0×10^{-4}	.05
interest	0.02	7.6×10^{-3}	2.85	4.7×10^{-3}	.04
intervallic leaps	0.09	0.04	2.15	.03	.04
tempo fluctuation	0.31	0.28	1.09	.28	.02
occurrences	-0.05	0.05	-0.90	.37	.02
rhythm only	0.29	0.44	0.66	.51	.02
unsigned dyn. level	0.02	0.06	0.43	.67	.02
signed dyn. level	0.03	0.07	0.42	.68	.02
geom. mean likelihd.	0.23	1.50	0.16	.88	.02

with test statistic $F(21, 338) = 42.01$, $p < 2.2 \times 10^{-16}$, and $s = 1.64$ as the error standard deviation. For the forward model, $r^2 = .71$, meaning that this model explains 71% of the variation in the ratings. For the backward model, $r^2 = .72$. Hence both models explain a substantial proportion of the variation in ratings and the difference in the amount they explain is minimal. The forward model is more parsimonious than the backward model—it is built on just three explanatory variables (apart from the between-participant factor) while the backward model uses ten.

The sign of some of the coefficients in the backward model (6.3) is concerning. For example, *coverage* and *interest* have negative coefficients but, by definition, these variables would be expected to contribute positively towards a pattern being rated as noticeable/important. In defence of the backward model it should be recalled that some variables are constituents of other variables. For example, *occurrences* is a constituent of *coverage* and *interest*. Hence it is an over-simplification to say that the backward model contains counter-intuitive coefficients without examining the overall contribution of variables such as *occurrences*.

Partitioning the design matrix according to the nine mazurka excerpts, I performed nine-fold cross-validation for the forward and backward models, comparing their mean prediction for each pattern rating with the observed mean rating. That is, I kept a mazurka to one side, estimated regression parameters with data from the other eight mazurkas, and used the resulting regression models to predict mean ratings for patterns in the mazurka kept to one side. This process was repeated for each mazurka in turn. The result is that, on average, the forward model's mean predictions are much closer

to the observed mean ratings ($MSE = 0.96$) than are the backward model's mean predictions ($MSE = 2.37$). Therefore the forward model outperforms the backward model and there is evidence that the forward model in (6.2) gives better predictions than the backward model in (6.3).⁴

For the forward model, in Fig. 6.4 the mean predictions from the cross-validation are plotted against the observed mean ratings. Figure 6.5 is the analogous plot for the backward model. There are acceptable straight-line fits in each plot and, in the main, there is little to choose between the two models. However, it can be seen from Fig. 6.5 that one of the backward model's mean predictions is particularly large (the point at approximately (9, 20) in the split plot). This poor prediction is the reason that the backward model was out-performed by the forward model in cross-validation, so this item was investigated further. Figure 6.6 is a plot of ratings for patterns 1-20 (from the first two excerpts). For a given pattern, the four participant ratings are plotted as dots, joined by a line to give an indication of the range of the response. If fewer than four dots are visible then this is due to coincident ratings. The observed mean rating—the mean of the four participant ratings—is plotted as a cross, the forward model's mean prediction is plotted as an asterisk, and the backward model's mean prediction as a diamond. The backward model's poor prediction is for pattern eleven. This pattern has a higher *score* variable (= 396.01) than any other pattern and this is the cause of the large predicted rating. The forward model does not suffer from the same waywardness and, moreover, this will typically be the case—the forward model contains several fewer parameters than the backward model,

⁴Plots made to check model assumptions and to check for outliers did not lead to any model revisions or any outlying data being removed.

making it more robust.

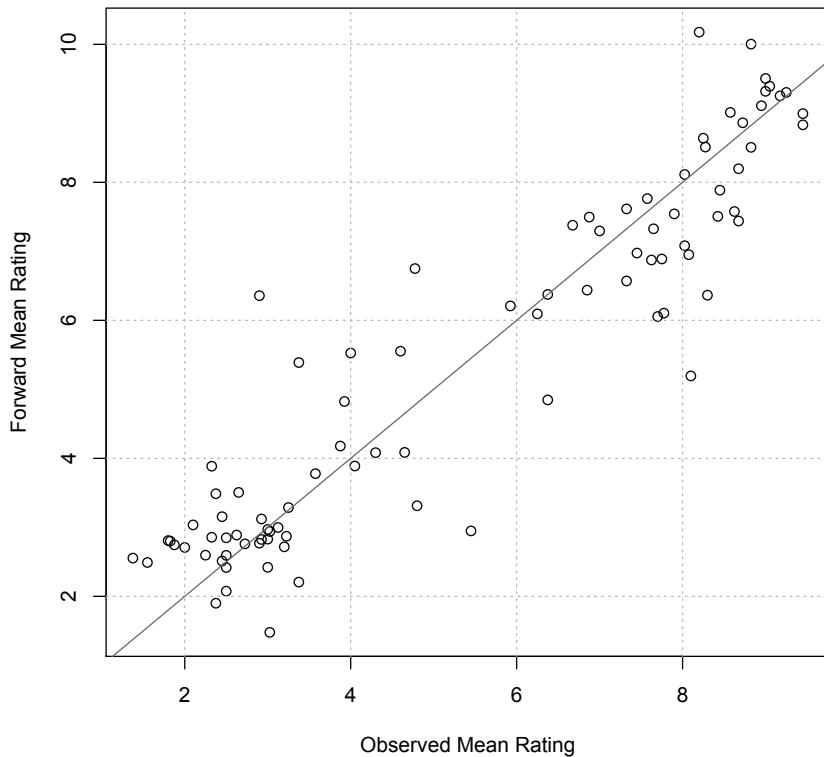


Figure 6.4: A plot of the forward model's mean prediction against the observed mean prediction for each of the ninety patterns.

An aim of this study was to address an unsolved problem in MIR, discussed in relation to Fig. 4.9, of producing a formula for predicting ratings that could be applied to *unseen* excerpts/pieces of music. To this end, the forward model in (6.2) was adapted so as not to include $\text{par}_2, \text{par}_3, \dots, \text{par}_{12}$, which relate to the individual participants and only apply in this study. Specifically the mean of the relevant coefficients, $(0.00 + 0.01 - 1.19 + \dots + 0.73)/12 \approx -0.52$, was added to the constant term 4.79, changing it to 4.28. So my formula for rating the extent to which a discovered musical pattern is

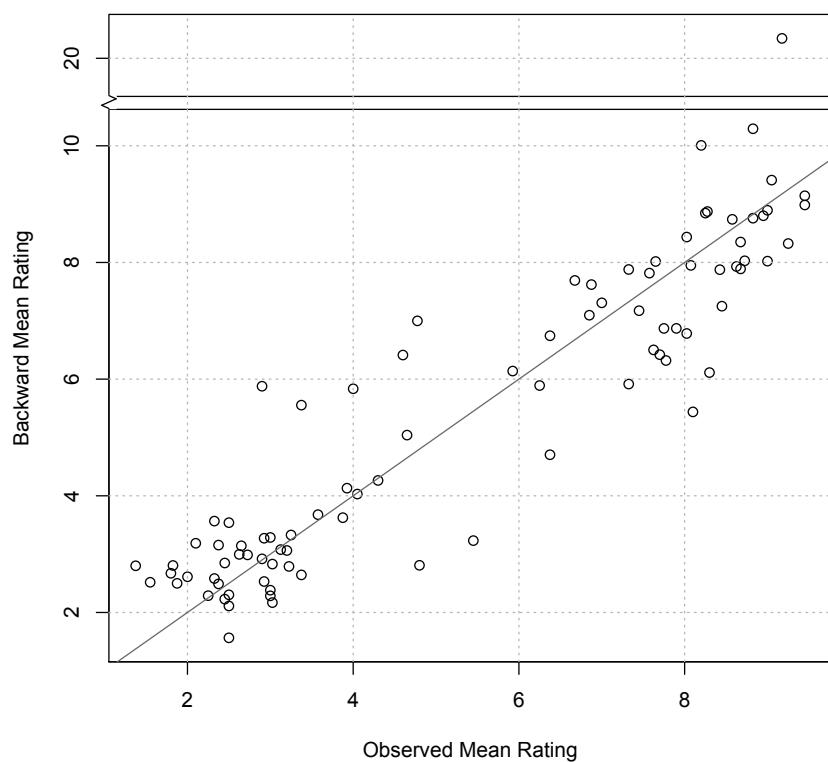


Figure 6.5: A plot of the backward model's mean prediction against the observed mean prediction for each of the ninety patterns.

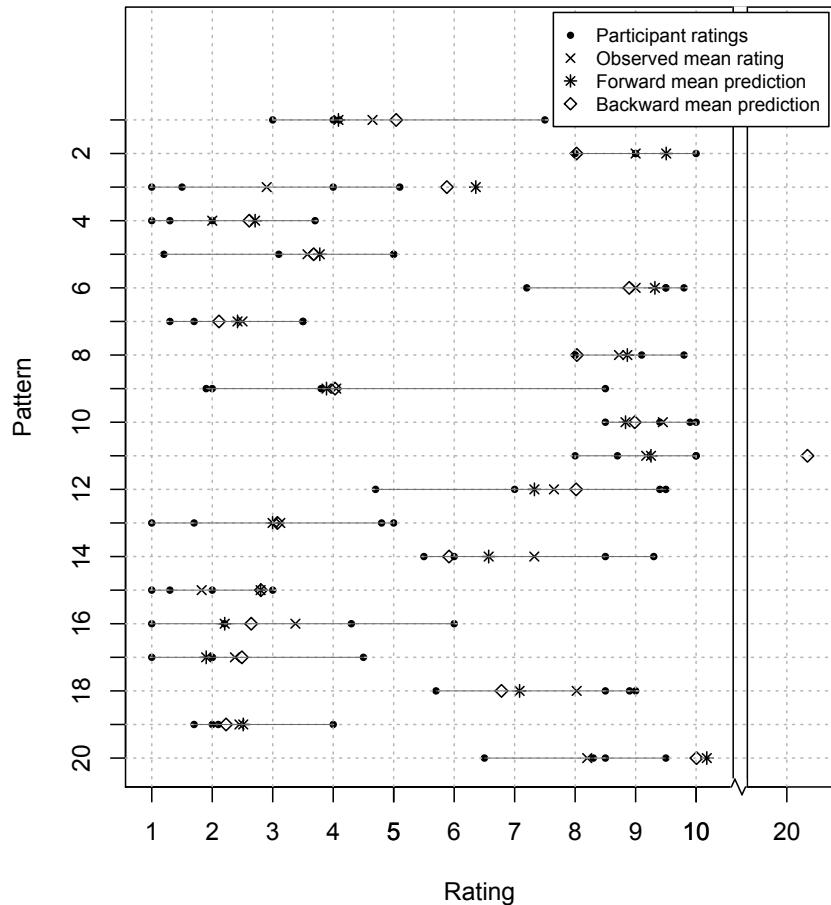


Figure 6.6: Observed and predicted ratings for patterns 1-20 (from the first two excerpts). If fewer than four dots (participant ratings) are visible per pattern then this is due to coincident ratings.

noticeable and/or important is

$$\begin{aligned} \text{rating} = & 4.28 + 3.42 \cdot \text{compactness} - 0.04 \cdot \text{expected_occurrences} \\ & + 0.65 \cdot \text{compression_ratio}. \end{aligned} \quad (6.4)$$

6.2.1 Predictive value of individual variables

Several of the explanatory variables that are in neither my forward nor backward models have been proposed by others as useful for predicting the salience of a pattern. The fact that a variable was not in these models does not imply it has no predictive ability. It is just that correlations between the explanatory variables mean that adding more variables to a model does not significantly improve the predictions of ratings after the model already contains certain variables. Other researchers may wish to construct other models or devise other explanatory variables, so further information about the predictive ability of variables is important. Table 6.1 is useful in this respect, as it shows the results of individual fittings. The rows are ordered by r^2 , the proportion of variability in the data that is explained by the model. The line just below *intervallic leaps* indicates a cut-off point in this table. Above this line, there is evidence to suggest that participants used a particular variable to form their ratings ($p < .05$). For all but six of the explanatory variables, there is evidence that the variable was useful for predicting participants' ratings. However, *maximum pitch centre* has a negative coefficient, which is contrary to the intuition given in Appendix C, and the same is true of *score*, *transposed repetitions* and the interaction *mean-centred cardinality* \times *occurrences*. Hence a cut-off point between $r^2 = .18$ and $r^2 = .14$ may give a better distinction between the variables that seem useful for pre-

dicting salience and those that do not.

6.2.2 Predictive ability of participants compared with the formula

The present work has developed a model for evaluating the salience of a pattern. It accounted for just over 70% of the variation in participants' ratings, which looks useful, but raises the question of whether it is easy to predict their ratings. Is the model useful or could a person give ratings effortlessly and more effectively? One approach to examining this question is to determine if participants in the experiment could predict the ratings that other participants gave. Does the formula that I have proposed in (6.4) give predictions that are closer to the consensus than any one music undergraduate can get?

For instance, the first participant rated patterns from three excerpts. Each pattern in each of these excerpts was also rated by three other participants.

For each pattern, the mean of these other ratings is called a *consensus*. Now, on average, are the first participant's ratings or the formula's predictions closer to this consensus? Accuracy was evaluated by calculating the mean squared error $\sum(\text{observed_value} - \text{prediction})^2/N$, where N is the number of observations (patterns that the first participant did not examine are ignored).

It turns out that the formula in (6.4) out-performs the first participant in terms of mean squared error (MSE). Analogous consensus tests for participants 2-12 found that the formula in (6.4) out-performed every participant.

The MSE for the participants and the formula are given in Table 6.2.

The table shows that ratings from participant 9 were closest to the con-

Table 6.2: Consensus test results. Mean squared error (MSE) when one participant's ratings estimate the consensus rating given by other participants, and when the formula in (6.4) is used to predict the consensus.

Participant	Participant MSE	Formula MSE
1	4.34	0.87
2	2.65	1.22
3	2.40	1.13
4	2.97	1.15
5	2.68	0.67
6	4.23	1.47
7	3.09	1.17
8	4.00	0.71
9	1.64	1.20
10	3.57	1.78
11	3.89	1.28
12	7.69	1.37
Average	3.60	1.17

sensus, but even this participant had a substantially larger MSE than the formula in (6.4). In predicting the consensus rating, the MSE of a participant was between 37% and 461% larger than the model and, on average, it was more than 200% larger. The extent to which the formula in (6.4) improved on participants' predictions was surprising. Judging by these results, it would be much better to use the formula to rate the salience of a pattern than to use ratings of any one of the participants.

6.3 Discussion

6.3.1 Conclusions

The primary aim of the work reported here was to investigate the claim that a formula such as (6.1) could model some aspect of the pattern discovery process. The formula in (6.4) was derived empirically using a linear model (6.2) that emerged as the stronger performer on cross-validation, and accounted for just over 70% of the variability in the participants' responses.⁵ I am cautious about drawing general conclusions from the results of one participant study, but can say that the above results do nothing to undermine the claim. A secondary aim of this chapter was to address an unsolved problem in MIR, of arranging the output of a pattern discovery system. For this purpose also, the formula in (6.4) was derived for rating patterns as relatively noticeable and/or important, based on variables that quantify attributes of a pattern and the piece of music in which it appears. I hope that MIR researchers will find the formula useful, especially when arranging the output of their pattern discovery systems. My review of existing work suggests that, up to this point, researchers have proposed formulae for rating discovered patterns with little foundation of empirical evidence. This chapter seems to be the first attempt to adopt an empirical method in the context of rating discovered patterns. The value of $r^2 = .71$ from the forward model in (6.2) is the proportion of variability in the ratings that is explained by the forward model, and it is greater (of course) than any of the r^2 values given in Table 6.1: hence the

⁵ Interested in estimating the maximum value of r^2 that could be achieved for this dataset, a model was fitted consisting of eleven participant indicator variables and 89 pattern indicator variables. For this model $r^2 = .80$, $s = 1.60$.

empirical method leads to a formula that offers a better explanation of the ratings than any of the proposed formulae do individually.

The results suggest that the formula in (6.4) can be used with confidence. Further, using the formula offers certain advantages. First, it can be used to filter or screen a large amount of data in a way that a human cannot. Second, the formula's rating for a certain pattern in a given piece is not subject to change, whereas a human may become tired or alter their preferences over time. I hope that this work will act as a springboard for other researchers wishing to build their own models. The model put forward in (6.2) is appealing due to its performance on cross-validation and also because of its parsimony, but there is the potential to test other models. In this respect, Table 6.1 gives some idea of which existing variables might lead to plausible alternative models.

In the introduction it was suggested that this chapter would be of interest to those working in music psychology. Can any more general conclusions be drawn that are pertinent to this field? First, forward selection, backward elimination and cross-validation could be valuable for testing hypotheses in other areas of music perception. Second, one can imagine situating each of the twenty-nine explanatory variables included in the regression on a line, its position on that line determined by how likely it is that an average music undergraduate is familiar with the variable's meaning. For example, contrast *intervallic leaps* with *compression ratio*; most music undergraduates would be able to furnish a definition of *intervallic leaps*, but even if the definition of *compression ratio* were given, few would acknowledge it as musically relevant. It is surprising and telling that the variables that appear in the formula

in (6.4)—*compactness*, *expected occurrences*, and *compression ratio*—are not those that one associates as being particularly familiar to music undergraduates. Perhaps these variables do not have much currency in music psychology and music analysis because they are relatively recent, or because their definitions (intuitive or mathematical) are somewhat unmusical. However, I have found evidence for their perceptual validity—in that they have emerged as predictors for participant ratings—and therefore they deserve a more prominent place in music psychology and music analysis. In particular, it would be worth attempting to situate the concepts of *compactness*, *expected occurrences*, and *compression ratio* in relation to music Gestalt principles, such as in voice-leading (Huron, 2001a) and stream segregation (Bregman, 1990).

Now that a rating formula (6.4) has been proposed, it would perhaps be helpful to discuss some of its components. This is done with reference to Figs. 6.7 and 6.8, and Table 6.3, which contains ratings and attributes for the patterns shown in these figures. Pattern *E* (Fig. 6.7) is a real sequence with three occurrences. The same figure contains pattern *F*, a predominantly scalic motif that passes between right and left hands in an imitative fashion, and has eight occurrences overall. Lastly, pattern *G* (Fig. 6.8) consists of the durations of pattern *F*, with an extra first note. From Table 6.3 it can be seen that patterns *E* and *F* are similar in terms of *cardinality* (the number of notes contained in one occurrence), as well as in terms of the significant components *compactness* and *expected occurrences*. They differ in how they *account* for the excerpt. For instance, a listener that comprehends pattern *F* as consisting of 16 notes that repeat $8 - 1 = 7$ times, is able to encode approximately sixteen bars of music using $16 + 7 = 23$ pieces of information.

On the other hand, a listener that does not comprehend pattern F just hears $16 \times 8 = 128$ notes. That is, they must try to encode the same sixteen bars of music using 128 pieces of information. The parsimony of comprehending pattern F is quantified by the *compression ratio* $128/23 \approx 5.57$. Pattern F accounts for approximately twice as many bars of music as does pattern E , which accordingly has a lower *compression ratio* of $45/(15+3-1) \approx 2.65$. So pattern F is rated higher than E (10.1 versus 7.9) by the formula in (6.4).⁶

Table 6.3: Ratings given by participants 3, 6, 7, and 11 to patterns E , F , G , and I shown in Figs. 6.7, 6.8, and 6.10. The observed means, ratings according to the formula in (6.4), and various attributes are also given.

Ratings and attributes	Pattern E	Pattern F	Pattern G	Pattern I
participant 3	8.0	10.0	6.0	9.6
participant 6	8.5	8.5	5.0	9.5
participant 7	10.0	9.6	1.0	7.5
participant 11	8.0	7.2	4.0	6.5
observed mean	8.6	8.8	4.0	8.3
rating formula	7.9	10.1	5.8	8.9
compactness	.94	1	.81	.99
expected occurrences	33.96	32.95	83.33	0.72
compression ratio	2.65	5.57	3.00	1.98
cardinality	15	16	17	80
occurrences	3	8	5	2
coverage	45	128	63	160

It has been suggested that listeners use parsimonious encodings where available to aid memorisation of note sequences (Deutsch, 1980). Although participants in my study were not asked to memorise passages, I too suggest that the availability of parsimonious encodings is intimately linked to the per-

⁶As an aside, there is a link to be made here between a music excerpt with a high compression ratio and a higher-order Markov model (cf. Sec. 8.2) where a small number of transitions are observed a large number of times.

[Vivace]

45 Pattern *E*, 1st occurrence

Pattern *E*, 2nd occurrence

Pattern *E*, 3rd occurrence

51 Pattern *F*, 1st occurrence

Pattern *F*, 2nd occurrence

57 Pattern *F*, 3rd occurrence

Pattern *F*, 4th occurrence

Pattern *F*, 5th occ.

62 Pattern *F*, 6th occurrence

Pattern *F*, 7th occurrence

Pattern *F*, 8th occ.

67

Figure 6.7: Bars 45-68 from the Mazurka in C major op.56 no.2 by Chopin. Occurrences of patterns *E* and *F* are indicated by black noteheads. Boxes demarcate the first two occurrences of pattern *F*.

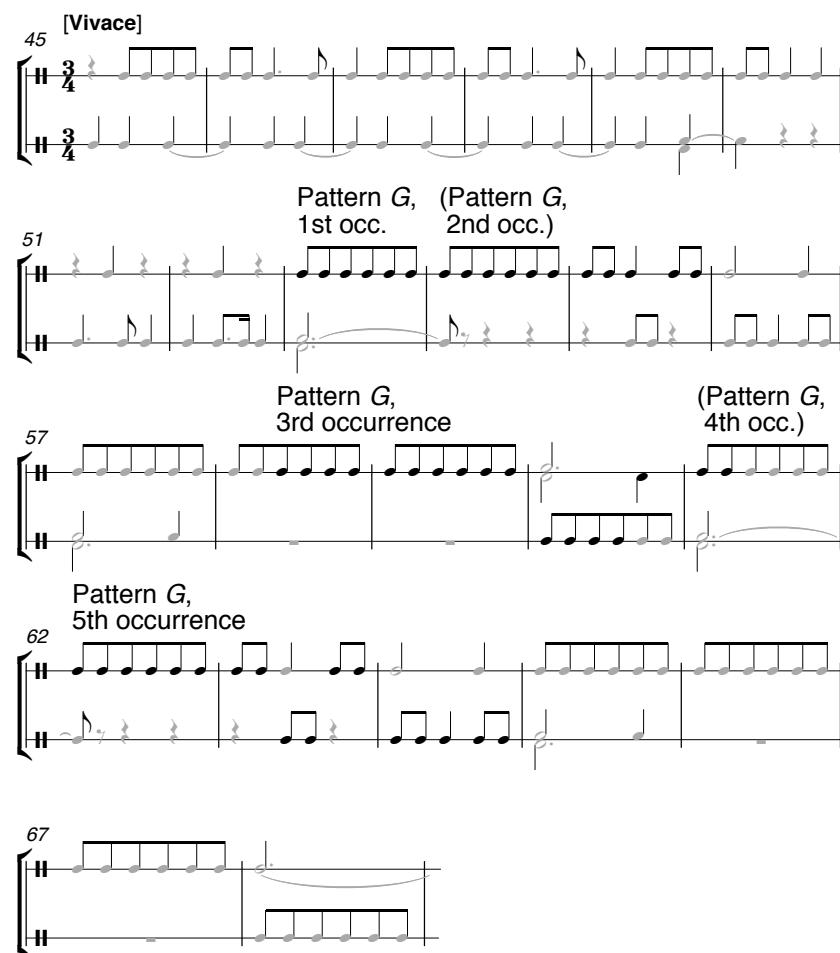


Figure 6.8: A rhythmic representation of bars 45-68 from the Mazurka in C major op.56 no.2 by Chopin. Due to overlapping notes, two occurrences (second and fourth) of pattern *G* are not shown.

ception of musical structure. Comprehending a pattern and its occurrences may also confirm or undermine what the listener perceives as the established meter. For example, pattern *B* in Fig. 6.2 lasts for three beats and often repeats immediately, confirming the prevailing triple meter of the mazurka. However, pattern *H* in Fig. 6.9 lasts for two beats and repeats immediately twice. Hence, the listener might hear three bars in duple meter, rather than two bars in triple meter (as written), thus undermining the prevailing triple meter of the mazurka.

[Maestoso]

H, H, H,
1st occ. 2nd occ. 3rd occ.

45

50

Figure 6.9: Bars 40-52 from the Mazurka in C \sharp minor op.41 no.1 by Chopin. Occurrences of pattern *H* are indicated by black noteheads.

Like patterns *E* and *F*, patterns *F* and *G* (Fig. 6.8) have similar *cardinalities*. Although pattern *G* fares worse than *F* in terms of the significant components *compactness* and *compression ratio*, the most striking difference

is in their *expected occurrences*. One way of interpreting these values is to say that a pattern like G , consisting of fourteen consecutive quavers, a crotchet, and two more quavers, is not that unexpected in the context of a mazurka. In fact, it could be said that pattern G is $83.33/32.95 \approx 2.5$ times more likely to occur than pattern F , which is essentially the same pattern but with a specific pitch configuration. Recalling that *expected occurrences* in (6.4) has a negative coefficient, the high value for pattern G contributes $-0.04 \times 83.33 \approx -3$ to the rating, whereas the lower value for pattern F contributes only $-0.04 \times 32.95 \approx -1$. Hence, the *expected occurrences* component accounts for approximately half of the difference between the rating for pattern G of 5.8, and that of 10.1 for pattern F .

Pattern I (Fig. 6.10 and Table 6.3) and pattern J (Fig. 6.11) highlight two ways in which the formula in (6.4) might be improved. All but one of the participants, as well as the rating formula, agree that pattern F (formula rating 10.1) should be rated above pattern I (formula rating 8.9). Pattern I defines a small section, with an original statement at bars 53-60 being repeated immediately at bars 61-68. However, if instead the listener hears bars 53-68 as consisting of eight occurrences of pattern F , then the two occurrences of pattern I are more or less implied. Therefore, pattern F rather than pattern I would be mentioned in an analysis essay, so the rating of 8.9 for I is too high. Augmenting the rating formula in (6.4) to adjust for these kind of implications could lead to improved performance. Pattern J (Fig. 6.11) provides an instance of high participant ratings (9.1, 7.8, 8.3, and 8.0) being at odds with a lower formula rating (6.3). First, the formula's performance could be improved if the concept of octave equivalence was incorporated in

the representation: each occurrence of pattern J is followed immediately by the pitch classes G and B. Participants may have heard these other notes as part of the pattern and this could have inflated the ratings. Second, the formula's performance could be improved if the concept of harmonic function was incorporated—a more ambitious aim. One of the reasons why pattern J receives a relatively low formula rating is because there are three nonpattern notes among the first occurrence at bar 29, reducing the *compactness*. The chord on beat 3 of bar 29 is an augmented sixth chord (German or French depending on whether the E \flat 5 or D5 is counted), whereas the chord on beat 3 of bars 30-32 is a diminished seventh or dominant chord (again depending on whether the E \flat 5 or D5 is counted) above a G3 pedal. Hence the different chords, augmented sixth versus dominant, make legitimate the omission of nonpattern notes in the left hand of bar 29. However, the harmonic function in each case is similar—moving towards G major (or G dominant seventh)—so in this sense the omitted notes are part of a more abstract pattern.

What does it mean if a variable appears below the line of statistical significance in Table 6.1? It could mean that the concept giving rise to this variable is music-perceptually and analytically obsolete. More likely, however, it means that I have failed to capture the concept adequately in the variable's definition. For instance the *signed dynamic level* of a pattern is calculated by summing over scores given to dynamic markings, but perhaps it would be better captured by analysing the amplitude of waveform segments. Another possibility is that the variable does capture the concept, but that participants did not apply this concept in a consistent manner when forming ratings. Although the forward model in (6.2) does reveal an underlying con-

[Vivace]

45

Pattern I,
1st occurrence

51

57

Pattern I,
2nd occ.

62

67

Figure 6.10: Bars 45-68 from the Mazurka in C major op.56 no.2 by Chopin. Occurrences of pattern I are indicated by black noteheads.

[Allegretto non tanto]

22

Pattern J,
1st occ.

Pattern J,
2nd occ.

Pattern J,
3rd occ.

Pattern J,
4th occ.

28

33

Figure 6.11: Bars 17-36 from the Mazurka in C minor op.30 no.1 by Chopin. Occurrences of pattern *J* are indicated by black noteheads.

sistent explanation ($r^2 = .71$), there is still considerable leeway. This leeway is perhaps where the music analyst makes his or her mark, by interpreting a piece of music in a novel way, yet within the realm of feasibility. To reiterate a point from the introduction, I am not suggesting that music analysts should use a formula, just that the process of rating musical patterns as more or less noticeable and/or important reflects the practice of deciding what to mention and what not to mention in a music analytical essay. A pedagogical outcome of this chapter is that the results could form part of a tool to help students with the discovery of patterns in music, so fostering the ‘desire to encounter a piece of music more closely, to submit to it at length, and to be deeply engaged by it, in the hope of thereby understanding more fully how it makes its effect’ (Pople, 2004, p. 127).

6.3.2 Future work

An outstanding issue to address is whether the formula in (6.4) can be applied to translational patterns in mazurkas other than those included in the participant study. Does the formula scale up to longer excerpts/entire pieces? And does it generalise to music by other composers, for different instrumental forces, from different periods, genres etc? Answering the second question is beyond the scope of this chapter, but a tentative answer to the scaling question follows. Box-and-whisker plots of the absolute errors between observed mean ratings and forward mean ratings against excerpt length are shown in Fig. 6.12. Two of the nine excerpts are 16 bars long, three are 20 bars long, and four are 24 bars long. Any trends in these box-and-whisker plots—for instance if the median (thick black) line increased with excerpt length—might

suggest that the formula in (6.4) does not scale up to longer excerpts. Looking at the plots, there is no evidence to suggest that the forward model suffers from scaling problems: neither the median nor the interquartile ranges appear to be a function of excerpt length; and whereas there are outlying values for the 16- and 20-bar excerpts (points more than $1\frac{1}{2}$ times the interquartile range from the box), the errors for the 24-bar excerpts contain no outliers.

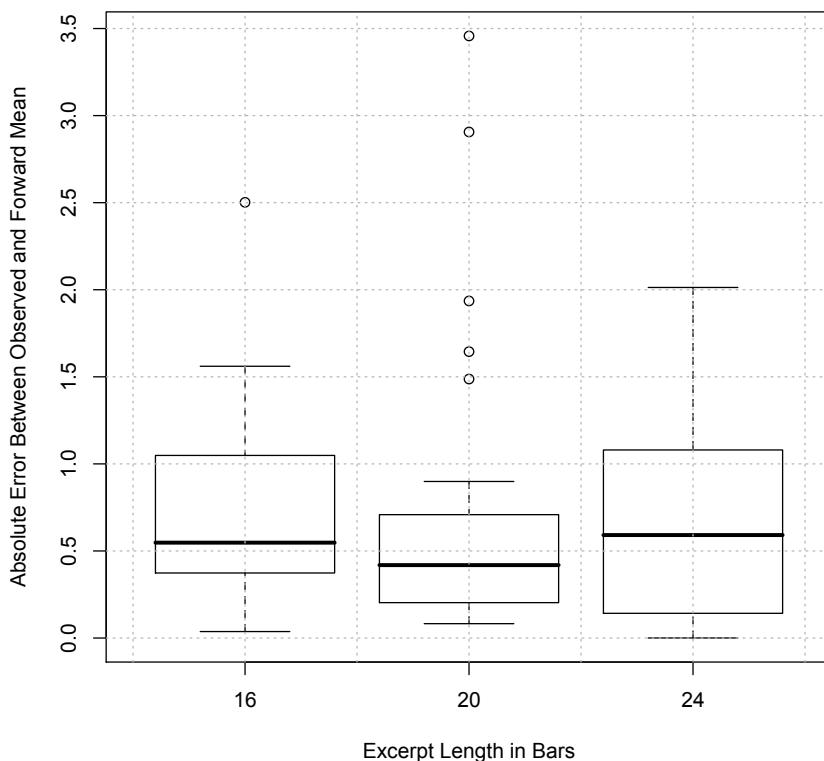


Figure 6.12: Box-and-whisker plots to explore the relationship between model performance and excerpt length. For each of the ninety patterns investigated, the absolute error between the observed mean rating and forward mean rating is calculated. This data is then split into three categories depending on the length of excerpt in which a pattern occurs.

There are several worthwhile directions in which this research could be taken. First, the participants in the study described above were twelve music

undergraduates. But music listeners, expert and nonexpert alike, might be able to rate discovered patterns. With music undergraduates, it was possible to assume a substantial amount of knowledge and expertise. Music undergraduates at the University of Cambridge prepare for exams in which they analyse and compose whole pieces of music without recourse to recordings or a means of playing through passages. Therefore, it was not deemed necessary to isolate patterns aurally for participants. Further, it did not surprise me that none of the participants made substantial use of the digital piano, and that two participants did not want to listen to the recordings of excerpts. It could be said that a particular performance of a piece can have an undue influence on the perception of musical structure. On the other hand, such an approach seems to neglect that music is heard primarily rather than seen, and this reliance on the score has been criticised before, and labelled *scriptism* (Cook, 1994, p. 79). In short, with a greater number of participants and considerable amendments to the design, a similar trial could be conducted with nonexpert listeners. Second, a previous participant study (Tan, Spackman, and Peaslee, 1980) investigated how listeners' judgements of music were affected by repeated exposure, by conducting a trial with the same participants on two occasions, separated by two days. Neither repeated exposure nor time were considered as factors in my participant study, yet there is much anecdotal evidence to suggest that comprehension of a piece varies with exposure, and in particular that a listener discovers new patterns in a piece over time. This acknowledgement could be cause for concern, for how can the performance of a pattern discovery system be evaluated by comparison with a human benchmark performance, if this benchmark is not an absolute but

instead depends on exposure or time? The definition of a human benchmark merits further attention, although different definitions may be appropriate to different situations.

Third, it is possible to argue that different occurrences of the same pattern ought to be rated individually. With reference to Fig. 6.1, arguably the first occurrence of pattern *A* is more noticeable than the second occurrence. The first occurrence is at the excerpt's very beginning, isolated to some extent, whereas the second occurrence dovetails with preceding and proceeding phrases. In the study described above, participants were asked to give a pattern one overall rating, taking all occurrences of the pattern into account. Both the issues of ratings affecting one another and of pattern occurrences being rated individually merit further investigation. Finally, aspects of my analysis have focused on mean ratings. However, there was marked disagreement between participant ratings over some patterns. For example pattern 9 in Fig. 6.6 received ratings with a standard deviation of 3.09, whereas the standard deviation of ratings for pattern 10, say, was only 0.99. Although the mean rating for pattern 9 is lower than that for pattern 10, some might argue that pattern 9 is the more important of the two: it has polarised the participants for some reason. Identifying factors that cause participant polarisation is another worthy topic for future work.

The recall of pattern discovery algorithms: An improved method

The main aim of the present chapter is to improve the recall (4.11) of the Structure Induction Algorithm (SIA, Meredith et al., 2002). This involves two key ideas: the problem of isolated membership, and compactness trawling. Chapter 4 introduced the topic of discovering repeated patterns in music. I justified focusing on five types of repetition—exact, with interpolation, transposed real, transposed tonal, and durational—labelling them the proto-analytical class (cf. Def. 4.1). Definition 4.2 stated the task of intra-opus discovery of translational patterns, and related concepts (Defs. 4.3 and 4.4) and algorithms (Defs. 4.5 and 4.6) were introduced. After addressing the ideas of isolated membership and compactness trawling, the present chapter contains an evaluation of my proposed improvements to SIA, and finishes by bringing together a new algorithm, SIACT, with the rating formula for pattern importance from (6.4).

7.1 The problem of isolated membership

To begin making improvements to SIA, I revisit an excerpt by D. Scarlatti (Fig. 4.11A), and expand the dataset representation D from (4.1) to include

more datapoints. In Sec. 4.2 (p. 72), it was noted that pattern P from (4.2) could be discovered by running SIA on the dataset D from (4.1). This is because P is the MTP (cf. Def. 4.4) for the vector $\mathbf{v} = (3, 3)$ and SIA returns all such patterns in a dataset. However, D is a conveniently chosen example consisting only of bars 13-15 of Fig. 4.11A. How might an MTP be affected if the dataset is enlarged to include bar 16? Letting

$$D^+ = \{\mathbf{d}_1, \dots, \mathbf{d}_{35}\}, \quad \mathbf{v} = (3, 3), \quad (7.1)$$

it can be verified that

$$P^+ = \text{MTP}(\mathbf{v}, D^+) = \{\mathbf{d}_1, \dots, \mathbf{d}_8, \mathbf{d}_{18}, \mathbf{d}_{19}, \mathbf{d}_{22}\}. \quad (7.2)$$

Unfortunately P^+ , the new version of P , contains three more datapoints, \mathbf{d}_{18} , \mathbf{d}_{19} , \mathbf{d}_{22} , that are isolated temporally from the rest of the pattern. This is an instance of what I call the *problem of isolated membership*. It refers to a situation where one or more musically important patterns are contained *within* an MTP, along with other temporally isolated members that may or may not be musically important. Intuitively, the larger the dataset, the more likely it is that the problem will occur. Isolated membership affects all existing algorithms in the SIA family, and could prevent them from discovering some translational patterns that a music analyst considers noticeable or important (see Sec. 7.2 for further evidence in support of this claim).

Based on the findings of the last chapter, where *compactness* emerged as the most significant explanatory variable for pattern importance, my proposed solution to the problem of isolated membership is to take the SIA

output and *trawl inside* each MTP from beginning to end, returning subsets that have a compactness greater than some threshold a and that contain at least b points.

Definition 7.1. Compactness (Meredith et al., 2003). The *compactness* of a pattern is the ratio of the number of points in a pattern to the number of points in the region of the dataset in which the pattern occurs. The *compactness* of a pattern $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_l\}$ in a dataset $D = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n\}$ is defined by

$$c(P, D) = l / |\{\mathbf{d}_i \in D : \mathbf{p}_1 \preceq \mathbf{d}_i \preceq \mathbf{p}_l\}|. \quad (7.3)$$

■

Different interpretations of *region* lead to different versions of compactness. The version employed in (7.3) is of least computational complexity, worst case $O(kn)$, where k is the number of dimensions and n is the number of points in the dataset. If the lexicographical ordering (cf. Def. 2.3 and Fig. 4.11B) is known, this computational complexity can be reduced.

For example, the compactness of pattern Q in Fig. 4.11C is $8/9$, as there are 8 points in the pattern and 9 in the dataset region $\{\mathbf{d}_9, \mathbf{d}_{10}, \dots, \mathbf{d}_{17}\}$ in which the pattern occurs.

One of Meredith et al.'s (2002) suggestions for improving/extending the SIA family is to 'develop an algorithm that searches the MTP TECs generated by SIATEC and selects all and only those TECs that contain convex-hull compact patterns' [p. 341]. The way in which my proposed solution is crucially different to this suggestion is to trawl inside MTPs. It will not suffice

to calculate the compactness of an entire MTP, since we know it is likely to contain isolated members. Other potential solutions to the problem of isolated membership are to:

- Separate an MTP into distinct sets wherever the inter-onset interval between consecutive datapoints exceeds one beat (Wiggins, 2007, p. 323). This solution, though elegant in its simplicity, may cause unnecessary segmentation of large-scale repetitions that contain rests.
- Segment the dataset before discovering patterns. The issue is how to segment appropriately—usually the discovery of patterns *guides* segmentation (Cambouropoulos, 2006), not the other way round.
- Apply SIA with a sliding window of size r . Approximately, this is equivalent to traversing only the elements on the first r superdiagonals of \mathbf{A} in (4.9). The issue is that the sliding window could prevent the discovery of very noticeable or important patterns, if their generating vectors lie beyond the first r superdiagonals.
- Consider the set of all patterns that can be expressed as an intersection of MTPs, which may not be as susceptible to the problem of isolated membership. The issue with this larger class is that it is more computationally complex to calculate, and does not aim specifically at tackling isolated membership.

The algorithmic form of my solution is called a *compactness trawler*. It may be helpful to apply it to the example of P^+ in (7.2), using a compactness threshold of $a = 2/3$ and points threshold of $b = 3$. The compactness of successive subsets $\{\mathbf{d}_1\}, \{\mathbf{d}_1, \mathbf{d}_2\}, \dots, \{\mathbf{d}_1, \dots, \mathbf{d}_8\}$ of P^+ remains above the

threshold of $2/3$ but then falls below, to $9/18$, for $\{\mathbf{d}_1, \dots, \mathbf{d}_8, \mathbf{d}_{18}\}$. So we return to $\{\mathbf{d}_1, \dots, \mathbf{d}_8\}$, and it is output as it contains $8 \geq 3 = b$ points. The process restarts with subsets $\{\mathbf{d}_{18}\}$, $\{\mathbf{d}_{18}, \mathbf{d}_{19}\}$, and then the compactness falls below $2/3$ to $3/5$ for $\{\mathbf{d}_{18}, \mathbf{d}_{19}, \mathbf{d}_{22}\}$. So we return to $\{\mathbf{d}_{18}, \mathbf{d}_{19}\}$, but it is discarded as it contains fewer than 3 points. The process restarts with subset $\{\mathbf{d}_{22}\}$ but this also gets discarded for having too few points. The whole of P^+ has now been trawled. The formal definition follows and has computational complexity $O(kn)$.

Definition 7.2. Compactness trawler and SIACT. Two parameters are a compactness threshold of $0 < a \leq 1$ and a points (or cardinality) threshold of $b \geq 1$.

1. Let $P = \{\mathbf{p}_1, \dots, \mathbf{p}_l\}$ be a pattern in a dataset D and $i = 1$.
2. Let j be the smallest integer such that $i \leq j < l$ and $c(P_{j+1}, D) < a$, where $P_{j+1} = \{\mathbf{p}_i, \dots, \mathbf{p}_{j+1}\}$. If no such integer exists then put $P' = P$, otherwise let $P' = \{\mathbf{p}_i, \dots, \mathbf{p}_j\}$.
3. Return P' if it contains at least b points, otherwise discard it.
4. If j exists in step 2, re-define P in step 1 to equal $\{\mathbf{p}_{j+1}, \dots, \mathbf{p}_l\}$, set $i = j + 1$, and repeat steps 2 and 3. Otherwise re-define P as empty.
5. After a certain number of iterations P will be empty and the output can be labelled P'_1, \dots, P'_N , that is N subsets of the original P , where $0 \leq N \leq l$.

I give the name *Structure Induction Algorithm and Compactness Trawler* (SIACT) to the process of calculating all MTPs in a dataset (SIA), followed by the application of the compactness trawler to each MTP. ■

The compactness-trawling stage in SIACT requires $O(kmn)$ calculations, where m is the number of MTPs returned by SIA. If desired, it is then possible to take the output of SIACT and calculate the TECs. These TECs are represented by the set H in Fig. 4.12. To my knowledge, this newest member of the SIA family is the only algorithm intended to solve the problem of isolated membership.

7.2 Evaluation

A music analyst analysed the Sonata in C major L1 and the Sonata in C minor L10 by D. Scarlatti, and the Prelude in C \sharp minor BWV849 and the Prelude in E major BWV854 by J.S. Bach. The analyst's task was similar to the intra-opus discovery task (Def. 4.2, p. 61): given a piece of music in staff notation, discover translational patterns that occur within the piece. Thus, a benchmark of translational patterns was formed for each piece, the criteria for benchmark membership being left largely to the analyst's discretion. One criterion that was stipulated was to think in terms of an analytical essay: if a pattern would be mentioned in prose or as part of a diagram, then it should be included in the benchmark. Figure 7.1 contains some of the analyst's annotations for bars 1-19 of the Sonata in C minor L10 by D. Scarlatti.¹ The analyst is referred to as independent because of the relative freedom of the task and because they were not aware of the details of the SIA family, or my new algorithm. The analyst was also asked to report where aspects of musical interest had little or nothing to do with translational patterns, as

¹The analyst's complete annotations and a parallel commentary can be found on the accompanying CD (or at <http://www.tomcollinsresearch.net>)

these occasions will have implications for future work.

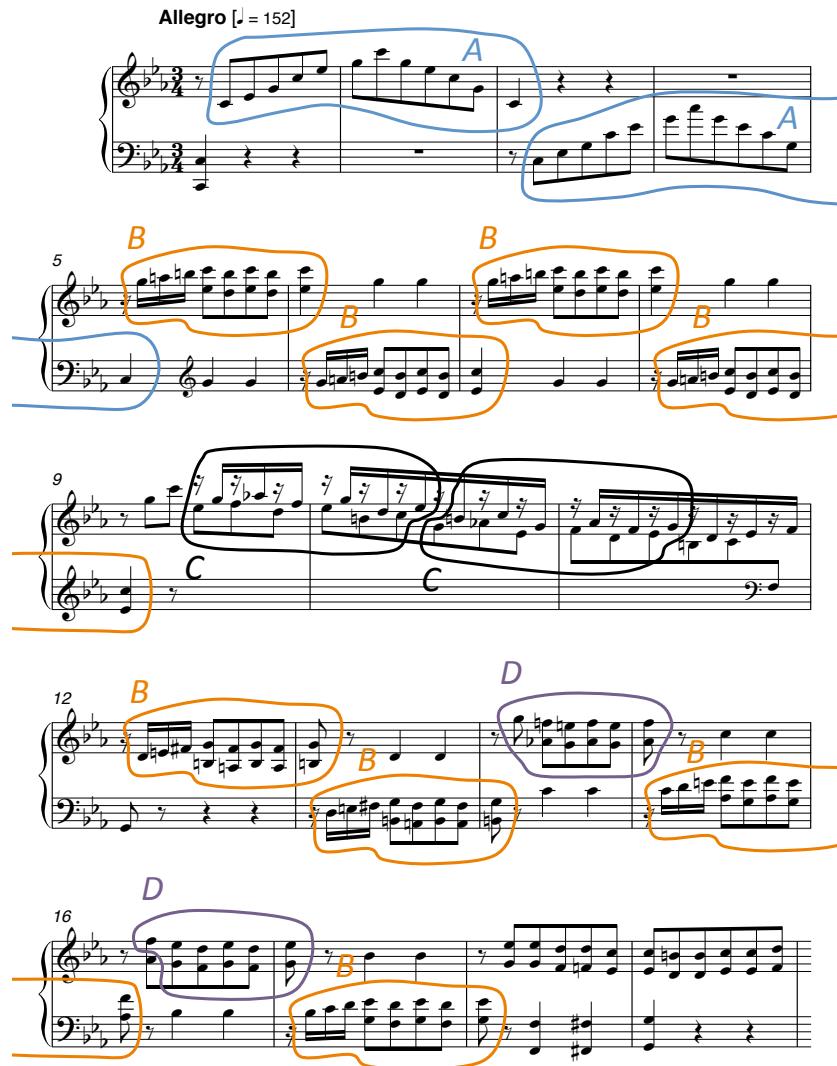


Figure 7.1: Bars 1-19 from the Sonata in C minor L10 by D. Scarlatti. Bounding lines indicate some of the analyst’s annotations for this excerpt.

Three algorithms—SIA (Meredith et al., 2002), COSIATEC (Meredith et al., 2003) and my own, SIACT—were run on datasets that represented L1, L10, BWV849, and BWV854. For COSIATEC the non-parametric version of the rating heuristic was used (Forth and Wiggins, 2009) and for SIACT

I used a compactness threshold of $a = 2/3$ and a points threshold of $b = 3$. The choice of $a = 2/3$ means that at the beginning of an input pattern, the compactness trawler will tolerate one non-pattern point between the first and second pattern points, which seems like a sensible threshold. The choice of $b = 3$ means that a pattern must contain at least three points to avoid being discarded. This is an arbitrary choice and may seem a little low to some. Each point in a dataset consisted of an ontime, MIDI note number (MNN), morphetic pitch number (MPN), and duration (voicing was omitted for simplicity on this occasion). Nine combinations of these four dimensions were used to produce projections of datasets (cf. Def. 2.2), on which the algorithms were run. These projections always included ontime, bound to: MNN and duration; MNN; MPN and duration; MPN; duration; MNN mod 12 and duration; MNN mod 12; MPN mod 7 and duration; MPN mod 7.² For the first time, the use of pitch modulo 7 and 12 enabled the concept of octave equivalence to be incorporated into the geometric method as discussed here.

If a pattern is in the benchmark, it is referred to as a target; otherwise it is a nontarget. An algorithm is judged to have discovered a target if a member of the algorithm's output is equal to the target pattern or a translation of that pattern. In the case of COSIATEC the output consists of TECs, not patterns. So I will say it has discovered a target if that target is a member of one of the output TECs. Table 7.1 shows the recall and precision of the three algorithms for each of the four pieces. Often COSIATEC did not discover any target patterns, so for these pieces it has zero recall and precision. This

²These combinations are not exhaustive, but it was not felt necessary to run the algorithms on a projection of ontime, MNN, and MPN, say, having run the algorithms on projections for ontime and MNN, and ontime and MPN.

Table 7.1: Results for three algorithms on the intra-opus pattern discovery task, applied to four pieces of music. Recall is the number of targets discovered, divided by the sum of targets discovered and targets not discovered. Precision is the number of targets discovered, divided by the sum of targets discovered and nontargets discovered.

Piece →	L1	L10	BWV849	BWV854
Algorithm ↓	Recall			
SIA	.29	.22	.28	.22
COSIATEC	.00	.17	.00	.00
SIACT	.50	.65	.56	.61
	Precision			
SIA	1.5 e^{-5}	1.1 e^{-5}	1.3 e^{-5}	1.8 e^{-5}
COSIATEC	.00	.02	.00	.00
SIACT	2.6 e^{-3}	1.5 e^{-3}	7.8 e^{-4}	2.0 e^{-3}

is in contrast to the parametric version’s quite encouraging results for J.S. Bach’s two-part inventions (Meredith, 2006b; Meredith et al., 2003). When it did discover some target patterns in L10, COSIATEC achieved a better precision than the other algorithms, as it tends to return far fewer patterns per piece (168 on average compared with 8,284 for SIACT and 385,299 for SIA). Hence the two remaining contenders are SIA and SIACT. SIACT, defined in Def. 7.2, out-performs SIA in terms of both recall and precision. Having examined cases in which SIA and COSIATEC fail to discover targets, I ascribe the relative success of SIACT to its being intended to solve the problem of isolated membership. Across the four pieces, the running times of SIA and SIACT are comparable (the latter is always slightly greater since the first stage of SIACT is SIA).

7.3 Discussion

7.3.1 Conclusions

This chapter has discussed and evaluated algorithms for the intra-opus discovery of translational patterns. One of my motivations was the prospect of improving upon current solutions to this open MIR problem. A comparative evaluation was conducted, including two existing algorithms and one of my own, SIACT. For the pieces of music considered, it was found that SIACT out-performs the existing algorithms considerably with regard to recall and, more often than not, it is more precise. Therefore, my aim of improving upon the best current solution has been achieved. Central to this achievement was the formalisation of the problem of isolated membership. It was shown that for a small and conveniently chosen excerpt of music, a maximal translatable pattern corresponded exactly to a perceptually salient pattern. When the excerpt was enlarged by just one bar, however, the MTP gained some temporally isolated members, and the salient pattern was lost inside the MTP. My proposed solution, to trawl inside an MTP, returning compact subsets, led to the definition of SIACT.

I am now in a position to combine knowledge elicited about the attributes of a pattern that matter to human analysts (Chapter 6) with the improved pattern discovery algorithm SIACT, so as to rate output patterns. When SIACT is run on three dataset representations of bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin, the ten top-rated output patterns according to the formula in (6.4) are as shown in Appendix D, Figs. D.1-D.14.³ The

³The *compactness threshold* was $a = 4/5$, and the *points threshold* was $b = 5$. These parameters are both slightly larger than the parameters used for the Baroque keyboard

three dataset representations are projections on to ontime and MIDI note number (MNN), ontime and morphetic pitch number (MPN), and ontime and duration.⁴ Patterns *A-J*, depicted in Figs. D.1-D.14 prompt the following observations:

1. The first occurrence of the top-rated pattern, pattern *A* (Fig. D.1), overlaps with its second occurrence (Fig. D.2). There is also some overlap between the first and second occurrences of pattern *B* (Figs. D.3 and D.4 respectively), but less so than with *A*. The occurrences of *A* occupy the time intervals [0, 9] and [6, 15], which are overlapping intervals, whereas the occurrences of *B* occupy the time intervals [12, 24] and [24, 36], which merely touch.
2. Patterns *A, B, E*, and *F*, which were discovered in the ontime-MNN projection, have their approximate equivalents in the ontime-MPN projection—patterns *C, D, G*, and *H* respectively. Although the necessity for separate MNN and MPN projections is clear (otherwise one of real or tonal sequences will not be discoverable), browsing through near-duplicates of discovered patterns is tiresome.
3. The second occurrence of pattern *F* (Fig. D.10) is a subset of the first occurrence of pattern *E* (Fig. D.9), and *F* has one more occurrence than *E* overall. The same can be said of patterns *B* and *E*.
4. Patterns *I* and *J* (Figs. D.13 and D.14 respectively) were discovered in the ontime-duration projection. Durational patterns tend not to

works in Sec. 7.2. The justification is that Chopin’s mazurkas tend to have thicker textures.

⁴For the sake of simplicity, this is a smaller number of projections than the nine considered in Sec. 7.2.

be among the very top-rated patterns, as comparable patterns with specific pitch profiles have lower *expected occurrences* and hence higher ratings.

To overcome what may be seen as shortcomings in points 1, 2, and 4 above, I recommend the following simplifications:

- (a) Run SIACT on one projection of the dataset—ontime, MNN, and MPN—and rate the output according to (6.4).
- (b) Let discovered pattern P have first onttime ω and last onttime ω' . Filter out P if $\omega' - \omega$ is less than the number of beats in one bar.
- (c) Filter out overlapping occurrences of the same pattern. That is, if Q , with first onttime ω_Q , is a later occurrence of P , with last onttime ω'_P , then filter out Q if $\omega_Q < \omega'_P$. If this results in only one remaining occurrence of P in the dataset, then filter out the entire discovered translational equivalence class (TEC). For instance, pattern A (Fig. D.1) would be filtered out, but pattern B (Fig. D.3) would not.
- (d) If two different TECs, $\text{TEC}(P, D)$ and $\text{TEC}(Q, D)$, are such that P is rated above Q , they have the same translators (e.g. $T(P, D) = T(Q, D)$), and Q is a subset of P , then filter out $\text{TEC}(Q, D)$.

Recommendations (a)-(d) are observed when SIACT and the rating formula (6.4) reappear in Chapter 9. Recommendation (a) addresses points 2 and 4 in the previous list, but it goes against the spirit of considering several projections of the same dataset. Forth and Wiggins (2009) have made a recommendation that might address point 2 as well, which involves grouping

discovered TECs together according to so-called primary and secondary patterns. Recommendation (c) addresses point 1 about patterns with properly overlapping occurrences. Recommendations (b) and (d) act as helpful simplifications when discovered patterns are used as the template for generating a stylistic composition (cf. Chapter 9).

When the recommended steps (a)-(d) are applied to bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin, three discovered patterns remain, and are shown in Fig. 7.2. Pattern $P_{1,1}$ (indicated by the solid blue line) is rated higher than pattern $P_{2,1}$ (indicated by the dashed green line), which in turn is rated higher than $P_{3,1}$ (indicated by the dotted red line). Patterns are rated by the perceptually validated formula (6.4) and labelled in order of rank, so that pattern $P_{i,j}$ is rated higher than $P_{k,l}$ if and only if $i < k$. The second subscript denotes occurrences of the same pattern in lexicographic order. That is, pattern $P_{i,j}$ occurs before $P_{i,l}$ if and only if $j < l$.

Point 3 in the previous list (about patterns being subsets of one another) leads to two theoretical considerations. First, it is possible to represent discovered patterns as a digraph, with an arc leading from the vertex for pattern $P_{i,j}$ to the vertex for $P_{k,l}$ if and only if $P_{i,j} \subset P_{k,l}$. (Digraphs were introduced on p. 17.) The corresponding graph for the discovered patterns shown in Fig. 7.2 is given in Fig. 7.3. The position of each vertex is immaterial, but it is helpful to place each vertex horizontally at the ontime where the corresponding pattern begins, and vertically by pattern ranking. The total number of arcs emanating from a pattern's vertex is defined as that pattern's *subset score*, denoted \S . For instance, pattern $P_{3,3}$ has a subset score of $\S(P_{3,3}) = 2$, whereas pattern $P_{3,1}$ has a subset score of $\S(P_{3,1}) = 0$.

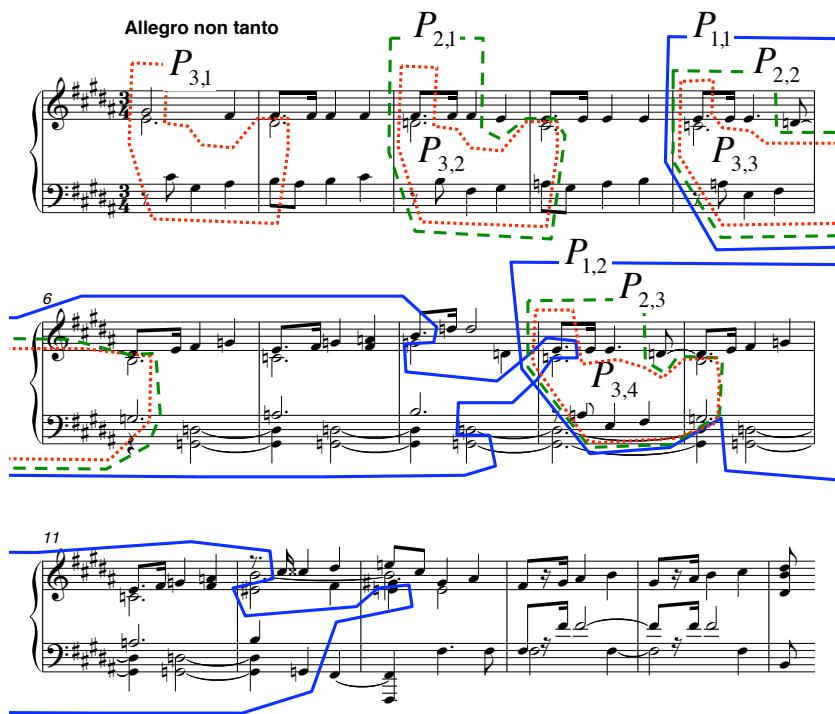


Figure 7.2: SIACT was applied to a representation of bars 1-16 of the Mazurka in B major op.56 no.1 by Chopin, and the results were filtered and rated. Occurrences of the top three patterns are shown.

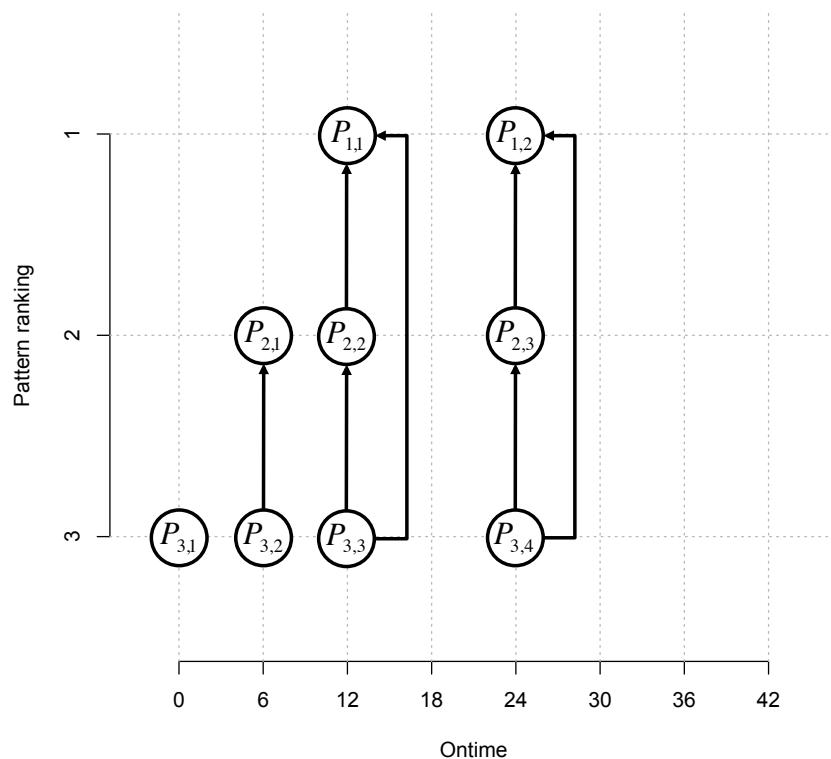


Figure 7.3: In this digraph, each vertex represents the pattern of the same name from Fig. 7.2. An arc is drawn from the vertex representing pattern $P_{i,j}$ to the vertex representing $P_{k,l}$ if and only if $P_{i,j} \subset P_{k,l}$. For the sake of clarity, vertices are placed horizontally at the ontimer of the corresponding pattern, and vertically by pattern ranking.

The second theoretical consideration might be called *hierarchy of patterns*, or *pattern implication*. For example, putting Figs. 7.2 and 7.3 to one side, suppose that in another piece and corresponding dataset there are patterns as indicated by the digraph in Fig. 7.4. In this dataset, pattern $P_{1,1}$ has a second occurrence, pattern $P_{1,2}$. Also, pattern $P_{2,1}$ has a second occurrence, $P_{2,2}$, and both $P_{2,1}$ and $P_{2,2}$ are subsets of $P_{1,1}$. The existence of the subsequent occurrences $P_{2,3}, P_{2,4}$ is implied by the existence of $P_{1,2}$. Or, one could say that there is a hierarchy of patterns established by $\text{TEC}(P_{1,1}, D)$ and $\text{TEC}(P_{2,1}, D)$. In general, for a pattern P in a dataset D , such a hierarchy is evident when the *translators themselves* $T(P, D)$ contain a translational pattern of cardinality two or more. Suppose for instance that $\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3, \mathbf{t}_4$ are the translators of $P_{2,1}$ from Fig. 7.4, so \mathbf{t}_1 is the zero vector mapping $P_{2,1}$ to itself, the vector \mathbf{t}_2 translates $P_{2,1}$ to $P_{2,2}$, the vector \mathbf{t}_3 translates $P_{2,1}$ to $P_{2,3}$, and the vector \mathbf{t}_4 translates $P_{2,1}$ to $P_{2,4}$. Then a hierarchy of patterns is evident, as $\{\mathbf{t}_1, \mathbf{t}_2\}$ is a translational pattern of cardinality two, being a translation of $\{\mathbf{t}_3, \mathbf{t}_4\}$.

7.3.2 Future work

The weight placed on the improved results reported in this chapter is limited somewhat by the extent of the evaluation, which includes only four pieces, all from the Baroque period, and all analysed by one expert. Extending and altering these conditions and assessing their effect on the performance of the three algorithms is a clear candidate for future work. There are also more sophisticated versions of compactness and the compactness trawler algorithm that could be explored, and alternative values for the compactness and points

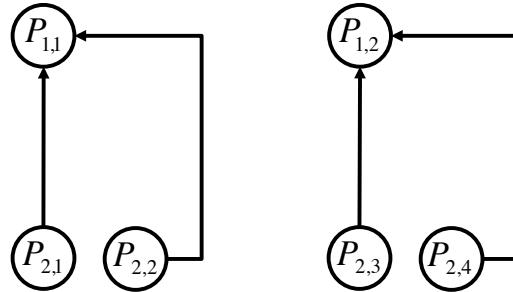


Figure 7.4: Each vertex in this digraph represents a pattern in a dataset. An arc is drawn from the vertex representing pattern $P_{i,j}$ to the vertex representing $P_{k,l}$ if and only if $P_{i,j} \subset P_{k,l}$. The term *hierarchy of patterns* refers to the way in which existence of patterns $P_{2,3}$ and $P_{2,4}$ is implied by the existence of $P_{1,2}$.

thresholds, a and b . The discovery of patterns from the proto-analytical class (cf. Def. 4.1) has provided a sensible starting point for this research, but extending definitions such as maximal translatable pattern (4.4) might allow other perceptually salient classes of pattern to be discovered, and so is an important and challenging next step. Cases of failure, where SIACT does not discover targets, will be investigated. Perhaps some of these cases share characteristics that can be addressed in a future version of the algorithm. Although SIA has been presented before as the sorting of matrix elements (Meredith et al., 2002), the connection that \mathbf{A} in (4.9) makes with similarity matrices (Peeters, 2007; Ren et al., 2004) may lead to new insights or efficiency gains.

Another important question is: could one focused algorithm encompass the many and diverse classes of musical pattern? It seems improbable, and the discussion of Figs. 4.10 and 4.11 in Sec. 4.1 could be interpreted as a

counterexample. Hence, given the improved voice separation algorithms, and string-based and geometric methods that now exist, another worthy topic for future work would be the unification of a select number of algorithms within a single user interface. This would bring me closer to achieving an aim stated on p. 61, of enabling music analysts, listeners, and students to engage with pieces of their choice in a novel and rewarding manner. To this end, the work reported here clearly merits further development.

8

Application: Markov models of stylistic composition

One component of a Markov model is the state space. This chapter begins by discussing different options for state spaces, and the musical implications. By the end of the chapter, a Markov chain has been defined that is at the heart of the models described in Chapter 9. A review of methods for automating the compositional process was given in Chapter 5. It seems that Markov chains (introduced in Sec. 3.3) are appropriate for the more open-ended tasks in stylistic composition (e.g., briefs 4 and 5 on p. 86, or the Chopin mazurka brief on p. 94). This is despite Markov chains, in their simplest form, being unable to model global musical structure. One of the potential applications of a pattern discovery algorithm such as SIACT (Chapter 7) is to inform the generation of stylistic compositions, remedying the aforementioned structural myopia. Accordingly, the next two chapters develop two models for stylistic composition, Racchman-Oct2010 and Racchmaninof-Oct2010 (acronyms explained in due course). In the former model, global structures can only occur by chance, but the latter model incorporates the results of a pattern discovery algorithm, thus ensuring that generated passages contain certain types of pattern. The development of the models addresses general issues in stylistic composition, for instance how to avoid generating a passage of music

that replicates too much of an existing piece in the intended style. Justification for decisions concerning this and similar issues are provided, but in any work such as this, different choices lead to different models. Evaluation of the two models on the Chopin mazurka brief is deferred until Chapter 10.

8.1 Realisation and musical context

Two issues with the randomly generated melodies (3.21)-(3.23) from Chapter 3 (p. 47) is that we do not know when the pitch classes begin and how long they last. (They could also be distributed across different octaves/instruments but I will ignore this for the time being.) The lack of ontimes and durations is not necessarily a weakness, as a composer might welcome the challenge of furnishing these melodies with a rhythmic profile. Alternatively, there are models that generate pitches only, making use of the rhythmic profile of an existing melody (Pearce, 2005). There are further possibilities for ensuring that the output of a model has a rhythmic profile. One possibility is to broaden the state space I , so that it includes both a ‘rest’ state and durations. Setting a crotchet equal to 1, such a state space for the material in Fig. 8.1 would be

$$I' = \{(rest, 1), (F, \frac{1}{2}), (F, 1\frac{1}{2}), (G, \frac{1}{2}), (G, 1), (G, 2), (A, \frac{1}{2}), (A, 2), (B\flat, \frac{1}{2}), (B, \frac{1}{2}), (B, 2), (C, \frac{1}{2}), (C, 1), (C, 4), (D, \frac{1}{2}), (E, 1)\}. \quad (8.1)$$

Definitions of the transition matrix \mathbf{P} and the initial distribution \mathbf{a} would also change. The former would be a 16×16 matrix (as $|I'| = 16$), and it would appear more sparse (with more zero entries) than \mathbf{P} in (3.19). The

increase in sparsity increases the probability that a melody generated from this model will replicate the original. Already we are skating on thin ice, as $(I, \mathbf{P}, \mathbf{a})$ from (3.18)-(3.20) results in a melody (3.21) whose first nine notes (A, G, F, G, F, G, A, B, G) differ from the first nine of Fig. 8.1 in only two places. Replication-avoidance strategies are revisited in Section 9.1.1.

Figure 8.1: Reproduction of Fig. 3.7. Bars 3-10 of the melody from ‘Lydia’ op.4 no.2 by Gabriel Fauré (1845-1924).

Another possibility, instead of broadening the state space to include duration, is to retain some musical context when analysing transitions between states.¹ For instance, in Fig. 8.1 there are four transitions from F to another pitch class: three to pitch-class G, which all last a quaver; and one to A, which lasts a minim. A *transition list* is more appropriate than a transition matrix for recording this information. The first three elements of the

¹Please be aware of a distinction between the terms *musical context* and *temporal context* (p. 188).

transition list L for the material in Fig. 8.1 are shown,

$$L = \left(\left(F, (G, \frac{1}{2}), (G, \frac{1}{2}), (G, \frac{1}{2}), (A, 2) \right), \right. \\ \left(G, (A, 2), (F, \frac{1}{2}), (A, \frac{1}{2}), (A, \frac{1}{2}), (B\flat, \frac{1}{2}), (F, 1\frac{1}{2}), (A, \frac{1}{2}) \right), \\ \left. \left(A, \underbrace{(G, \frac{1}{2}), (B, 2), (B, \frac{1}{2}), (C, 1), (G, 2), (G, 1\frac{1}{2}), (F, \frac{1}{2}), (G, 1)}_{(\dagger)} \right), \dots \right). \quad (8.2)$$

Suppose that when generating a melody using L , the first generated pitch-class i_0 is A. For the next random variable X_1 to assume a value i_1 , we look to the third element of the transition list L (as A is the third element of the state space I) and make a random equiprobable choice between the elements labelled (\dagger) in (8.2). In terms of pitch class, this is equivalent to looking at the third row of the transition matrix \mathbf{P} in (3.19) and choosing between $X_1 = F$, $X_1 = G$, $X_1 = B$, $X_1 = C$, with respective probabilities $\frac{1}{8}, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}$. The difference between transition matrix and transition list is that L in (8.2) retains some musical context (in this case durations), which can be used to furnish the generated pitch classes with a rhythmic profile. Compared with broadening the state space, as in (8.1), using a transition list reduces the probability of replicating original material from Fig. 8.1. As an example, the pitch-class-duration pairs $(A, 2)$ and $(B, \frac{1}{2})$ could never result consecutively from the model with the broadened state space, as a minim A is never followed by a quaver B in Fig. 8.1. However, these pairs could result consecutively from the model with the transition list as *state* and *musical context* are to some extent dissociated. Further possibilities for handling multiple dimensions are discussed by Whorley et al. (2010).

Concept 8.1. Realisation. Sometimes the *output* generated by formation of a Markov model does not consist of ontime-pitch-duration triples, which might be considered the bare minimum for having generated a passage of music. The term *realisation* refers to the process of converting output that lacks one or more of the dimensions of ontime, pitch, duration, into ontime-pitch-duration triples. ■

For example, the model $(I, \mathbf{P}, \mathbf{a})$ from (3.18)-(3.20) is used to generate pitch classes (3.21) that could be *realised* by assigning the corresponding on-times, octave numbers, and durations from the original (Fig. 8.1). This would give $(0, A4, \frac{1}{2})$, $(\frac{1}{2}, G4, \frac{1}{2})$, $(1, F4, 2)$, etc. Pearce (2005) also realises generated pitch material by using a pre-existing rhythmic profile. With reference to equation 8.2, it was shown how realisation is possible by retaining relevant musical context in a transition list. It is conceivable to avoid the process of realisation by defining a state space from which ontime-pitch-duration triples can be generated directly, although in the example given, a broadened state space made the replication of original material more likely. The process of realisation arises again below, when a choice is discussed between state spaces that consist of music *sets*, such as pitch classes, and state spaces that consist of music *groups*, such as pitch-class intervals (representing music as sets and groups was discussed in Chapter 2, pp. 13-19).

8.2 Different orders of Markov models and different state spaces

Loy (2005) gives an accessible introduction to *m*th-order Markov chains in

the context of monophonic music. In Secs. 3.3 and 8.1, 1st-order Markov chains were discussed. Markov chains of higher order take into account more temporal context. In a 2nd-order Markov chain, the probability that X_{n+1} takes the value i_{n+1} depends not just on X_n , but on two random variables, X_n and X_{n-1} . This is what more temporal context means. For example, in Fig. 8.1 there are four transitions from pitch-classes G to A, and one is followed by G, two by B, one by F. So

$$\mathbb{P}[X_{n+1} = G \mid (X_n, X_{n-1}) = (G, A)] = \frac{1}{4}, \quad (8.3)$$

$$\mathbb{P}[X_{n+1} = B \mid (X_n, X_{n-1}) = (G, A)] = \frac{1}{2}, \quad (8.4)$$

$$\mathbb{P}[X_{n+1} = F \mid (X_n, X_{n-1}) = (G, A)] = \frac{1}{4}. \quad (8.5)$$

These probabilities would appear in the row of the transition matrix corresponding to the pair (G, A). The merit of a 2nd-order Markov chain is that there is a dependency between X_{n+1} and X_{n-1} , which is not true for a 1st-order chain. When these random variables assume values i_{n+1}, i_{n-1} in the 1st-order case, the musical effect could be incongruous, if i_{n-1} and i_{n+1} never appeared so close together in the original data. The disadvantage of more temporal context is an increased probability of replicating the original (Loy, 2005).

Thus far the focus has been on state spaces consisting of music sets (e.g., pitch classes), but it would have been acceptable to use music groups (e.g., directed semitone intervals) instead. The transition F, G is similar perceptually (equivalent, some would argue) to the transition C, D, so why not call the directed semitone interval of 2 a state, and count both F, G and C, D as

instances of this state? Using directed semitone intervals, a plausible state space for the material shown in Fig. 8.1 is

$$I'' = \{-5, -4, -3, -2, -1, 0, 1, 2, 3, 4\}, \quad (8.6)$$

and the transition matrix is

$$\begin{array}{cccccccccc} & -5 & -4 & -3 & -2 & -1 & 0 & 1 & 2 & 3 & 4 \\ \begin{matrix} -5 \\ -4 \\ -3 \\ -2 \\ -1 \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \left(\begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1/7 & 2/7 & 1/7 & 0 & 0 & 2/7 & 1/7 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2/3 & 0 & 1/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix} \right) & = \mathbf{P}''. \quad (8.7) \end{array}$$

For example, in Fig. 8.1 there are three directed semitone intervals of size three, the ninth element of the state space, hence the denominator 3 in the nonzero entries of the ninth row of \mathbf{P}'' in (8.7). Of the three transitions, two are followed by a directed semitone interval of -2 , the fourth element of the state space, hence $p_{9,4}'' = 2/3$, and one is followed by a directed semitone interval of 0 , the sixth element, hence $p_{9,6}'' = 1/3$. For the sake of an example, let the initial distribution \mathbf{a}'' always choose the directed semitone interval 2. So $(I'', \mathbf{P}'', \mathbf{a}'')$ defines a Markov model. A plausible list of directed semitone

intervals generated by this Markov model is

$$(2, 2, 1, 2, -4, 2, -4). \quad (8.8)$$

The process of realisation arises again: how should these intervals be converted back into pitches? Pitch F4 could be stipulated as the first of the generated melody, as it is the first pitch of the original melody (Fig. 8.1), and then the intervals in (8.8) would give the pitches

$$(F4, G4, A4, B\flat 4, C5, A\flat 4, B\flat 4, G\flat 4). \quad (8.9)$$

So deviation from the original pitch material can occur if the state space consists of intervals, but not if it consists of pitches. One could be forgiven for assuming that a first-order Markov model over a state space of directed semitone intervals is equivalent to a second-order Markov model over a pitch state space. There is a difference, however: the former can deviate from the original material whereas the latter cannot.

Concept 8.2. Deviation. Let $(I, \mathbf{P}, \mathbf{a})$ be a Markov model formed using a certain piece, or certain pieces, of music, and let i_0, i_1, \dots be output generated from $(I, \mathbf{P}, \mathbf{a})$. When this output is realised, an aspect of it (such as a pitch or perhaps the beat of the bar on which a note begins) may never have occurred in the original piece(s) of music. In which case $(I, \mathbf{P}, \mathbf{a})$ is said to *deviate*. ■

Above it was shown that $(I'', \mathbf{P}'', \mathbf{a}'')$, the model formed using directed semitone intervals, deviated with respect to pitch. It is also worth mentioning that $(I', \mathbf{P}', \mathbf{a}')$ from (8.1), the model with the broadened state space of pitch class and duration, deviates with respect to the beat of the bar. For instance,

$i_0 = (F, \frac{1}{2})$ and $i_1 = (A, 2)$ could result from this model, giving a minim beginning on beat $1\frac{1}{2}$, which never happens in Fig. 8.1. Deviation is meant to be a neutral term, although intuitively, the more a generated passage deviates, the less likely it is to be stylistically successful. Generally, there is a balance to strike between avoidance of deviation and avoidance of replication, as the former tends to increase the sparsity of the transition matrix/list, which increases the probability that a generated passage will replicate the original. I prize avoidance of replication above avoidance of deviation, so my proposed model allows deviation. In Chapter 9, strategies are presented for constraining output generated from a Markov model; strategies that can also have the effect of limiting deviation.

On the matter of state spaces over pitches or over intervals, there are several musical questions that arise if a pitch state space is chosen:

1. What if the model is constructed over several pieces in different keys?
Transposition of each piece to C major is a sensible solution (Cope, 2005, p. 89), but do pieces in G \flat major/E \flat minor get transposed up or down?
2. If transposition is the solution, what about pieces with ambiguous keys, or without keys? This question might seem only to apply to pre- and post-tonal works, but even among my chosen corpus (Chopin's mazurkas), there are examples of ambiguous keys (op.17 no.4, op.7 no.5).
3. A state space formed over pitches will preserve relations of key to a greater extent than will one formed over intervals, but is this preservation more important than constructing a nonsparse transition matrix?

Over a pitch state space, repeated melodies in different octaves/keys are assigned to different regions of the transition matrix, increasing its sparsity. I have cautioned against doing this where possible.

Given the three points above, my proposed model uses a state space that consists in part of intervals. The drawback of using intervals is a model that deviates, as discussed. I argue that the alternative, of using a pitch state space with the transposition solution suggested above and by Cope (2005), can be more problematic.

8.3 A beat-spacing state space

8.3.1 Partition point, minimal segment, and semitone spacing

Sections 8.1 and 8.2 were limited to consideration of melody. This was useful for exemplifying the definitions of Markov model and chain, and the concepts of sparsity, replication, realisation, and deviation, but monophony constitutes a very small proportion of textures in Western classical music. That is not to say models for generation of melodies contribute little to an understanding of musical style. Since a common compositional strategy is to begin by composing a melody (followed by harmonic or contrapuntal development), models that generate stylistically successful melodies are useful for modelling the first step of this particular strategy.

My proposed model assumes a different compositional strategy; one that begins with the full texture, predominantly harmonic (or vertical) but with scope for generated passages to contain contrapuntal (or horizontal) elements.

A clear candidate for future work is a model of stylistic composition that has several broad compositional strategies at its disposal (e.g., melody followed by harmonic/contrapuntal development, harmonic with elements of counterpoint, contrapuntal with elements of harmony, etc.). Within genres, and even within single movements, composers move from one texture to another. For instance, the archetypal texture of a mazurka by Chopin is homophonic, but a handful (op.50 no.3, op.56 no.2) emphasise independent melodic lines. As a further example, there is a striking contrast between homophonic writing (bars 1-4) and polyphonic writing (bars 5-13) in the excerpt of Tallis shown in Fig. 8.2.

Figure 8.2 will be used to demonstrate the state space of my proposed Markov model. A state in the state space consists of two elements: the beat of the bar on which a particular *minimal segment* begins; and the spacing in semitone intervals of the sounding set of pitches.

Definition 8.3. Partition point and minimal segment (Pardo and Birmingham, 2002, pp. 28-9). A *partition point* occurs where the set of pitches currently sounding in the music changes due to the ontime or offtime of one or more notes. A *minimal segment* is the set of notes that sound between two consecutive partition points. ■

The partition points for the excerpt from Fig. 8.2 are shown beneath the stave in Fig. 8.3. The units are crotchet beats, starting from zero. So the first partition point is $t_0 = 0$, the second is $t_1 = 3$, and the third is $t_2 = 4$, coinciding with the beginning of bar 2, and so on. The first minimal segment S_0 consists of the notes sounding in the top-left box in Fig. 8.3. Representing

Soprano If ye love me,____ keep my com mand- ments, and I will

Alto If ye love me,____ keep my com mand- ments,____

Tenor If ye love me,____ keep my com mand- ments,____ and

Bass If ye love me,____ keep my com mand- ments,____

6

pray the Fa - ther, and he shall

and I will pray the Fa - ther, and

I will pray the Fa - ther,____ and

and I will pray the Fa - ther,

10

give you a - no - ther com - fort - er,

he shall give you a - no - ther com - fort - er,

he shall give you a - no - ther com - fort - er, that

and he shall give you a - no - ther com - fort - er,

Figure 8.2: Bars 1-13 of 'If ye love me' by Thomas Tallis (c 1505-1585).

these notes as ontime-pitch-duration triples,

$$S_0 = \{(0, F3, 3), (0, A3, 3), (0, C4, 3), (0, F4, 3)\}. \quad (8.10)$$

The second minimal segment

$$S_1 = \{(3, D3, 1), (3, F3, 1), (3, D4, 1), (3, F4, 1)\}, \quad (8.11)$$

and the third minimal segment

$$S_2 = \{(4, C3, 2), (4, C4, 2), (4, E4, 2), (4, G4, 2)\}, \quad (8.12)$$

and so on. Conventionally, beats of the bar are counted from one, not zero. So the first minimal segment S_0 has ontime 0, and begins on beat 1 of the bar. The next segment S_1 begins on beat 4, and S_2 begins on beat 1 of the bar.

The second element of a state—the spacing in semitone intervals of the sounding set of pitches—is considered now. In Chapter 2 (p. 11) I discussed a bijection between the pitch of a note and a pair consisting of a MIDI note number and morphetic pitch number (also addressed by Meredith, 2006a). So each pitch s in a sounding set of pitches S can be mapped to a MIDI note number y .

Definition 8.4. Semitone spacing. Let $y_1 < y_2 < \dots < y_m$ be MIDI note numbers. The spacing in semitone intervals is the vector

$$(y_2 - y_1, y_3 - y_2, \dots, y_m - y_{m-1}). \quad (8.13)$$



Figure 8.3: Bars 1-13 of the 'If ye love me' by Tallis, annotated with partition points and minimal segments (cf. Def. 8.3). The partition points are shown beneath the stave. The units are crotchet beats, starting from zero. So the first partition point is $t_0 = 0$, the second is $t_1 = 3$, and the third is $t_2 = 4$, and so on. Minimal segments are indicated by grey boxes. The first minimal segment, S_0 in (8.10), consists of the notes sounding in the top-left box.

For $m = 1$, the spacing of the chord is the empty set. For $m = 0$, a symbol for ‘rest’ replaces the vector in (8.13). ■

The first minimal segment S_0 consists of the pitches F3, A3, C4, F4, which map to the MIDI note numbers 53, 57, 60, 65, giving a spacing in semitone intervals of

$$(57 - 53, 60 - 57, 65 - 60) = (4, 3, 5). \quad (8.14)$$

The next segment S_1 has spacing (3, 9, 3), and S_2 has spacing (12, 4, 3).

Definition 8.5. Beat-spacing state space. Let $I^{(3)}$ denote a state space where each state is a pair: the first element of the pair is the beat of the bar on which a minimal segment begins (cf. Def. 8.3); the second element of the pair is the semitone spacing of that minimal segment (cf. Def. 8.4). ■

If a Markov model is constructed over the excerpt in Fig. 8.3, using the beat-spacing state space $I^{(3)}$, then the first state encountered is

$$i = (1, (4, 3, 5)), \quad (8.15)$$

a list consisting of two elements: the beat of the bar on which the first minimal segment begins; and a vector containing the spacing in semitone intervals of the sounding set of pitches. The whole state space $I^{(3)}$ for this excerpt is shown below. For the sake of clarity, there is a new line for each new bar, and repeated states are shown in bold (repeated states should really be removed, as $I^{(3)}$ is a set).

$$\begin{aligned}
I^{(3)} = \Big\{ & \left(1, (4, 3, 5)\right), \left(4, (3, 9, 3)\right), \\
& \left(1, (12, 4, 3)\right), \left(3, (7, 5, 4)\right), \\
& \left(2, (7, 5, 4)\right), \left(3, (12, 4, 3)\right), \left(4, (7, 5, 3)\right), \\
& \left(1, (15, 3, 5)\right), \left(\mathbf{3}, (\mathbf{7}, \mathbf{5}, \mathbf{4})\right), \\
& \left(1, (7, 5)\right), \left(\mathbf{2}, (\mathbf{7}, \mathbf{5}, \mathbf{4})\right), \left(3, (16)\right), \left(4, (4, 12)\right), \\
& \left(1, (15)\right), \left(2, (15)\right), \left(3, (8)\right), \left(4, (7)\right), \left(4\frac{1}{2}, (9)\right), \\
& \left(1, (11)\right), \left(1\frac{1}{2}, (12)\right), \left(2, (4, 5)\right), \left(3, (7, 5, 7)\right), \left(4, (7, 5, 7)\right), \quad (8.16) \\
& \left(1, (7, 8)\right), \left(2, (7, 8)\right), \left(3, (9)\right), \left(4, (7)\right), \left(4\frac{1}{2}, (9)\right), \\
& \left(1, (10)\right), \left(2, (8, 7)\right), \left(3, (12, 3)\right), \left(4, (9, 4, 3)\right), \\
& \left(1, (3, 12)\right), \left(2, (3, 12)\right), \left(3, (12, 3, 5)\right), \left(4, (12, 3, 4)\right), \\
& \left(1, (7, 5, 4)\right), \left(2, (4, 3, 5)\right), \left(\mathbf{3}, (\mathbf{7}, \mathbf{5}, \mathbf{4})\right), \left(4, (12, 4, 3)\right), \\
& \left(1, (8, 4, 5)\right), \left(1\frac{1}{2}, (7, 5, 5)\right), \left(2, (3, 5, 7)\right), \left(3, (7, 5, 5)\right), \dots \\
& \left(4, (\mathbf{7}, \mathbf{5}, \mathbf{3})\right), \left(4\frac{1}{2}, (7, 5, 3)\right), \\
& \left(\mathbf{1}, (\mathbf{7}, \mathbf{5})\right), \left(3, \emptyset\right) \Big\}.
\end{aligned}$$

Up to and including bar 4 in Fig. 8.3, the texture is homophonic, so hopefully following the beat-spacing states in lines 1-4 of (8.16) is relatively straightforward. Bar 4 ends with the state $(3, (7, 5, 4))$, and on the first beat of bar 5 the soprano rests, giving the state $(1, (7, 5))$. From this point the writing becomes more polyphonic. Evidently a single note can belong to more than one state, but the choice of state space does not encode whether such a note

is held over to (from) a next (previous) state. The musical context, rather than the state space, is used to retain this information (cf. Sec. 8.3.2 for more details). The tenor part in bar 6 of Fig. 8.3 is interesting, as it contains the first offbeat note, a C4 on beat $4\frac{1}{2}$, creating an interval of 9 semitones with the soprano's A4. Therefore, the state is $(4\frac{1}{2}, (9))$, and can be seen on line 6 of (8.16).

Is the inclusion of *beat of the bar* in the state space justified? When Tallis was writing, for instance, barlines were not in widespread use. This raises the question of whether it makes sense to represent the chord setting ‘me’ in bar 2 of Fig. 8.2 and that setting ‘keep’ in bar 3 as different states, when they are the same chord: F3, C4, F4, A4. The first occurrence of the chord is on beat 3 of the bar, and the second on beat 2, which could influence what happens next, so representing the two occurrences as different states is justified. Compared with a model that did not use any metrical information in the state space (Collins, Laney, Willis, and Garthwaite, 2010), it would appear that including *beat of the bar* in the state space leads to generated output with more stylistically successful rhythms.

Minimal segments that last longer than a bar are potentially problematic. For example, suppose $(1, (12))$ and $(1, (7))$ are consecutive states in a piece. It can be inferred that the first chord spans an octave (12 semitones) and that the second chord spans a perfect fifth (7 semitones), but it is unclear whether the second chord begins a bar after the first, or two bars, three bars etc. In the current model, it is assumed that the second chord begins one bar after the first. It is possible to recover the actual answer from the information retained as musical context, but only very rarely in Chopin’s mazurkas do minimal

segments last longer than a bar. A similar problem is addressed by Cope (1996), when ties that cross barlines are removed: ‘Ties, however, especially when they cross bar lines and thus fall out of the data of a single object in the database, must be altered’ (p. 61). Thus Cope prevents any minimal segments lasting longer than a bar, but proposes reinstating some ties ‘at appropriate junctures in the final output by a user-controlled variable in the performance section of the user interface’ (*ibid.*). In the current model, ties may cross barlines. For instance, state $(3, (7, 5, 4))$ is followed by $(2, (7, 5, 4))$ in bars 2 and 3 of Fig. 8.3, and there is no new state on the downbeat of bar 3, as no notes end and no new notes begin. The only occasion on which ties are removed is between enharmonically equivalent notes. The E \flat 5 is tied to D \sharp 5 in bars 24-25 of Fig. 8.4, for example. This type of tie occurs only very rarely in Chopin’s mazurkas, but it is mentioned for the sake of completeness.

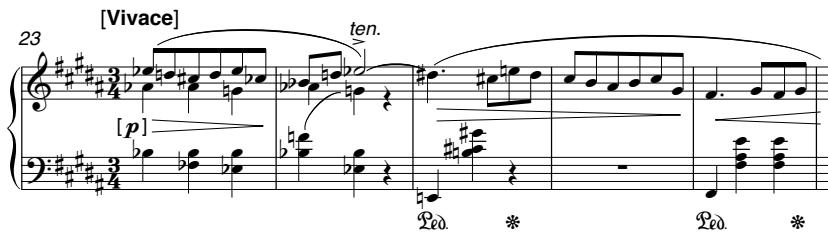


Figure 8.4: Bars 23-27 of the Mazurka in B major op.63 no.1 by Chopin.

While on the subject of notational curiosities, Chopin—as well as many other composers—often notates music that is impossible to perform on a piano, due to a regard for proper voice-leading. For instance, the D3 from bar 15 in Fig. 8.5A is still being held in bar 18 when, on beat 3, another D3 appears. It is impossible both to continue holding a note and play that same note on the piano without the use of the sustain pedal, which Chopin

indicates should not be depressed at this point. The choice of notation discloses a wish to display four independent voices, but Fig. 8.5B shows what would actually be played. If a corpus for an ensemble was being used, such as Joseph Haydn's minuets for string quartet, then there might be an argument for representing doubled notes such as these in chord spacings (as a zero), but I have removed doubled notes as indicated by Fig. 8.5B, as they are impossible to play on a piano and they increase the sparsity of the state space. Pedalling directions have also been ignored, as often they are left to the discretion of the editor or performer.

The figure consists of two musical staves, A and B, both labeled [Moderato] and starting at measure 15. Staff A shows four voices with complex note heads and stems. Staff B shows the same music but with simplified notation, likely representing how it would actually be played, with fewer note heads and stems.

Figure 8.5: (A) Bars 15-19 of the Mazurka in C minor op.56 no.3 by Chopin; (B) The same excerpt, but how it would actually be played.

8.3.2 Details of musical context to be retained

The concept of retaining some musical context when analysing transitions between states was discussed in relation to the transition list L in (8.2). In general, for a state space I with n elements, the form of the transition list is

$$\begin{aligned}
L = & \left((i_1, (j_{1,1}, c_{1,1}), (j_{1,2}, c_{1,2}), \dots, (j_{1,l_1}, c_{1,l_1})), \right. \\
& (i_2, (j_{2,1}, c_{2,1}), (j_{2,2}, c_{2,2}), \dots, (j_{2,l_2}, c_{2,l_2})), \\
& \dots, \\
& \left. (i_n, (j_{n,1}, c_{n,1}), (j_{n,2}, c_{n,2}), \dots, (j_{n,l_n}, c_{n,l_n})) \right). \tag{8.17}
\end{aligned}$$

Fixing $k \in (1, 2, \dots, n)$, let us look at an arbitrary element of this transition list,

$$L_k = (i_k, (j_{k,1}, c_{k,1}), (j_{k,2}, c_{k,2}), \dots, (j_{k,l_k}, c_{k,l_k})). \tag{8.18}$$

The first element i_k is a state in the state space I . In (8.2) i_k was a pitch class. In the current model, $i_k \in I^{(3)}$ is a beat-spacing state as discussed above. Each of $j_{k,1}, j_{k,2}, \dots, j_{k,l_k}$ is also an element of the state space. In (8.2) these were other pitch classes. In the current model, which uses a beat-spacing state space, they are the beat-spacing states for which there exists a transition from i_k , over one or more pieces of music. Each of $j_{k,1}, j_{k,2}, \dots, j_{k,l_k}$ has a corresponding musical context $c_{k,1}, c_{k,2}, \dots, c_{k,l_k}$, which is considered now in more detail. Attention is restricted to the first context $c_{k,1}$ to avoid introducing further subscripts. In (8.2), $c_{k,1}$ was a positive rational number, indicating the duration of the pitch-class $j_{k,1}$. For the current model, $c_{k,1}$ is itself a list:

$$c_{k,1} = (\gamma_1, \gamma_2, s, D), \tag{8.19}$$

where γ_1, γ_2 are integers, s is a string, and D is a dataset.² The dataset $D \in c_{k,1}$ contains datapoints that determine the beat-spacing state $j_{k,1}$. In

²Recall from Def. 2.2 that a dataset is a set of points in multidimensional space that represents a collection of notes (Meredith et al., 2002).

the original piece, the state $j_{k,1}$ will be preceded by the state $i_{k,1}$, which was determined by some set D' of datapoints. For the lowest-sounding note in each dataset D and D' , γ_1 is the interval in semitones and γ_2 is the interval in scale steps. For example, the interval in semitones between bass notes of the asterisked chords shown in Fig. 8.6 is $\gamma_1 = -5$, and the interval in scale steps is $\gamma_2 = -3$. If either of the datasets is empty, because it represents a ‘rest’ state, then the interval between their lowest-sounding notes is defined as \emptyset . The string s is a piece identifier. For instance, $s = \text{‘C-56-3’}$ means that the beat-spacing state $j_{k,1}$ was observed in Chopin’s op.56 no.3. At present, the reasons for retaining this particular information in the format $c_{k,1}$ may be unclear. As already discussed, retaining musical context can help with realisation (Concept 8.1) whilst reducing the probability of replicating original material. This does not explain, however, why the interval between lowest-sounding notes is retained, or why a piece identifier s or a dataset D is useful. These matters are revisited in Secs. 8.4 and 9.1.

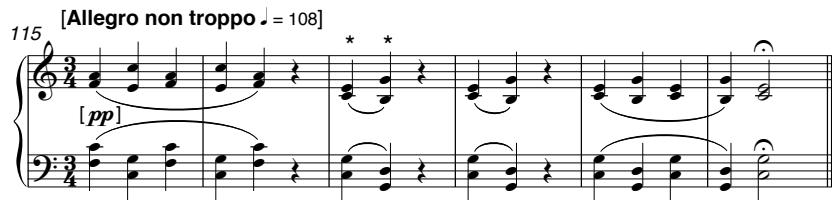


Figure 8.6: Bars 115–120 of the Mazurka in C major op.24 no.2 by Chopin.

8.4 Random generation Markov chain

Definition 8.6. Random generation Markov chain (RGMC). Let (I, L, A) be an m th-order Markov chain, where I is the state space, L is

the transition list of the form (8.17), and A is a list containing possible initial state-context pairs. I use the term *random generation Markov chain* (RGMC) to mean that:

1. A pseudo-random number is used to select an element of the initial distribution list A .
2. More pseudo-random numbers ($N - 1$ in total) are used to select elements of the transition list L , dependent on the previous selections.
3. The result is a list of state-context pairs

$$H = ((i_0, c_0), (i_1, c_1), \dots, (i_{N-1}, c_{N-1})), \quad (8.20)$$

referred to as the *generated output*. ■

I will be concerned with the steps involved in realising the generated output H of an RGMC (cf. Concept 8.1). In this section, a mapping from H to D will be described, where D is a dataset consisting of ontime-pitch-duration triples, which might be considered the bare minimum for having generated a passage of music. In the next section, the musical characteristics of such generated passages will be discussed. Readers familiar with the term *Markov chain Monte Carlo* (MCMC) may wonder how this differs from the definition of random generation Markov chain. Typically, the main application of MCMC is estimation of a Markov chain's invariant distribution, in a scenario where theoretical calculation is infeasible (Norris, 1997, pp. 206-216). Although the definitions of RGMC and MCMC are equivalent, I have used a different abbreviation to emphasise that the concern here is realising generated output, not estimation of an invariant distribution.

Definition 8.7. Markov model for Chopin mazukas. Let $I^{(4)}$ denote the state space for a first-order Markov model, containing all beat-spacing states (cf. 8.3) found over thirty-nine Chopin mazurkas.³ Let the model have a transition list $L^{(4)}$ with the same structure as L in (8.17), and let it retain musical context as in (8.19). The model's initial distribution list $A^{(4)}$ contains the first beat-spacing state and musical context for each of the thirty-nine mazurkas, and selections made from $A^{(4)}$ are equiprobable. ■

³Data from Kern Scores, <http://kern.ccarh.org>. Only thirty-nine mazurkas are used, out of an encoded forty-nine, because some of the others feature as stimuli in a later evaluation, so also including them in the state space of the model would be inappropriate. The thirty-nine are op.6 nos.1, 3, & 4, op.7 nos.1-3 & 5, op.17 nos.1-4, op.24 nos.2 & 3, op.30 nos.1-4, op.33 nos.1-3, op.41 nos.1-3, op.50 nos.1-3, op.56 nos.2 & 3, op.59 nos.1 & 2, op.63 nos.1 & 2, op.67 nos.2-4, and op.68 nos.1-4.

An RGMC for the model $(I^{(4)}, L^{(4)}, A^{(4)})$ generated the output

$$\begin{aligned}
H' = & \left(\left(\underbrace{\left((1, (7, 5, 4)), \right)}_{=i'_0}, \right. \right. \\
& \left. \left. \underbrace{\left(\emptyset, \emptyset, \text{'C-24-2'}, \{(0, 48, 53, 1), (0, 55, 57, 1), (0, 60, 60, 1), (0, 64, 62, 1)\} \right) }_{=c'_0} \right), \\
& \left((2, (7, 9, 8)), \right. \\
& \left(-5, -3, \text{'C-24-2'}, \right. \\
& \left. \left. \{(349, 43, 50, 1), (349, 50, 54, 1), (349, 59, 59, 1), (349, 67, 64, 1)\} \right) \right), \\
& \left((3, (7, 9, 5)), \right. \\
& \left(0, 0, \text{'C-17-4'}, \right. \\
& \left. \left. \{(206, 45, 51, 1), (206, 52, 55, 1), (206, 61, 60, 1), (206, 66, 63, 1)\} \right) \right), \\
& \left((1, (7, 10, 2)), \right. \\
& \left(0, 0, \text{'C-17-4'}, \right. \\
& \left. \left. \{(231, 45, 51, 1), (231, 52, 55, 1), (231, 62, 61, \frac{1}{2}), (231, 64, 62, 3)\} \right) \right), \\
& \left((1\frac{1}{2}, (7, 10, 2)), \right. \\
& \left(0, 0, \text{'C-17-4'}, \right. \\
& \left. \left. \{(207, 45, 51, 1), (207, 52, 55, 1), (207\frac{1}{2}, 62, 61, \frac{1}{2}), (207, 64, 62, 3)\} \right) \right), \\
& \left((2, (7, 10, 2)), \right. \\
& \left(0, 0, \text{'C-17-4'}, \right. \\
& \left. \left. \{(256, 45, 51, 1), (256, 52, 55, 1), (256, 62, 61, 1), (255, 64, 62, 3)\} \right) \right), \\
& \left((3, (7, 9, 3)), \right. \\
& \left(0, 0, \text{'C-17-4'}, \right. \\
& \left. \left. \{(233, 45, 51, 1), (233, 52, 55, 1), (233, 61, 60, \frac{1}{2}), (231, 64, 62, 3)\} \right) \right), \\
& \dots \\
& \left((1, (4, 5, 7)), \right. \\
& \left(0, 0, \text{'C-50-2'}, \right. \\
& \left. \left. \{(107, 56, 58, 2), (107, 60, 60, 2), (107, 65, 63, 2)(108, 72, 67, \frac{3}{4})\} \right) \right),
\end{aligned} \tag{8.21}$$

giving $N = 35$ state-context pairs in total. I have tried to make the format clear by bracing the first pair $H'_0 = (i'_0, c'_0)$ in (8.21). The formats of i'_0 and c'_0 are analogous to (8.15) and (8.19) respectively. Various aspects of this generated output will be discussed, beginning with the realisation of H' as ontime-pitch-duration triples. Once H' has been realised it can be notated as a passage of music, as shown in Fig. 8.7. By definition, different pseudo-random numbers would have given rise to a different—perhaps more stylistically successful—passage of music, but the output in (8.21) and passage in Fig. 8.7 have been chosen as a representative example of RGMC for the model $(I^{(4)}, L^{(4)}, A^{(4)})$.

To convert the first element $H'_0 = (i'_0, c'_0)$ of the list H' into ontime-pitch-duration triples, an initial bass pitch is stipulated, say E4, having MIDI note number 64 and morphetic pitch number 62. The chord spacing $(7, 5, 4)$ determines the other MIDI note numbers (MNN), $64+7 = 71$, $64+7+5 = 76$, and $64+7+5+4 = 80$. The corresponding morphetic pitch numbers (MPN) are found by combining the initial bass MPN, 62, with the dataset from the musical context

$$D = \{(0, 48, 53, 1), (0, 55, 57, 1), (0, 60, 60, 1), (0, 64, 62, 1)\}. \quad (8.22)$$

In their original context, the MPNs were 53, 57, 60, and 62. As the initial bass MPN is stipulated as 62, there will need to be a transposition up of $9 = 62 - 53$ scale steps. The remaining MPNs are $57 + 9 = 66$, $60 + 9 = 69$, and $62 + 9 = 71$. Due to the bijection between pitch and MNN-MPN representations (discussed in Chapter 2, p. 11), the pitch material of the first element H'_0 of the list H' is determined. The MNN-MPN pairs $(64, 62)$,



Figure 8.7: Realised generated output of an RGMC for the model $(I^{(4)}, L^{(4)}, A^{(4)})$. This passage of music is derived from H' in (8.21). The numbers written above the stave give the opus/number and bar of the source. Only when a source changes is a new opus-number-bar written. The box in bars 5-6 is for the purpose of a later discussion.

(71, 66), (76, 69), and (80, 71) map bijectively to the pitches E4, B4, E5, and G♯5 (see the first chord in Fig. 8.7).

Calculating an ontime for the first element H'_0 of the list H' is more straightforward than determining the pitches. The beat of the bar in the state part i'_0 of H'_0 is 1, indicating that the mazurka chosen to provide the initial state for the generated output, op.24 no.2, begins on the first beat of the bar. Adhering to the convention that the first full bar of a piece of music begins with ontime zero, the ontime for each triple being realised from H'_0 will be zero.⁴ It can be seen in the dataset from the musical context (8.22) that each datapoint has the same duration, 1 crotchet beat. This duration becomes the duration of each of the realised triples for H'_0 . The realisation of durations is not always so straightforward, due to notes that, in their original context, belong to more than one minimal segment (cf. Def. 8.3 and see bars 4-5 of Fig. 8.3).

The second element

$$\begin{aligned} H'_1 = & \left((2, (7, 9, 8)), \right. \\ & (-5^*, -3^*, 'C-24-2', \\ & \left. \{(349, 43, 50, 1), (349, 50, 54, 1), (349, 59, 59, 1), (349, 67, 64, 1)\}) \right) \end{aligned} \quad (8.23)$$

of the list H' is converted into ontime-pitch-duration triples in much the same way as was the first element H'_0 . One difference is the use of contextual information; in particular the intervals between bass notes of consecutive minimal segments. For example, the interval in semitones between bass notes

⁴Had the chosen mazurka started with an anacrusis, say op.24 no.3, which begins with a crotchet upbeat, then the first ontime of the realised passage would have been -1.

of the asterisked chords shown in Fig. 8.6 is $\gamma_1 = -5$, and the interval in scale steps is $\gamma_2 = -3$. This is the origin of the two asterisked numbers in (8.23). The interval between bass notes is retained in the passage being realised, giving the MNN-MPN pairs (59, 59), (66, 63), (75, 68), and (83, 73), and thus the pitches B3, F♯4, D♯5, and B5 (see the second chord in Fig. 8.7).

The realisation of H' continues, mapping $H'_2, H'_3, \dots, H'_{34}$ to ontime-pitch-duration triples. Bar 2 of Fig. 8.7, corresponding to elements H'_3, H'_4, H'_5, H'_6 , is worthy of mention, as this is the first occurrence of a nonhomophonic texture. How such a texture arises from generated output will be explained by answering two questions:

1. Why does F♯5 in bar 2 of Fig. 8.7 last for 3 beats?
2. Why are there two notes with pitch B3 in bar 2 of Fig. 8.7, the first lasting for a minim and the second for a crotchet?

In answer to the first question, the context duration of F♯5 is 3 beats. This can be seen from the last datapoint in the dataset corresponding to H'_3 ,

$$D = \{(231, 45, 51, 1^*), (231, 52, 55, 1), (231, 62, 61, \frac{1}{2}), (231, 64, 62, 3^*)\}, \quad (8.24)$$

indicated by the asterisked 3 in (8.24).⁵ The state durations of H'_3, H'_4, H'_5, H'_6 are $\frac{1}{2}, \frac{1}{2}, 1, 1$ respectively. That is, in bar 2 of Fig. 8.7, the minimal segments (cf. Def. 8.3) last for $\frac{1}{2}$ beat, $\frac{1}{2}$ beat, 1 beat, and 1 beat. If F♯5 is present as a pitch in the next state, and its context duration is greater than the current state duration, then it will be held over into the next state when the generated output is realised. So F♯5 lasts for the entirety of bar 2. Its context

⁵The pitch information of the datapoint does not correspond to F♯5, due to the transposition process discussed above.

duration, 3, is greater than each of the state durations, $\frac{1}{2}$, $\frac{1}{2}$, and 1, but as it is not present as a pitch on the downbeat of bar 3, it ceases to sound at this time. In answer to the second question, the context duration of B3 on the downbeat of bar 2 is 1 crotchet, as indicated by the asterisked 1 in (8.24). Now this context duration, 1, is greater than the state durations $\frac{1}{2}$, $\frac{1}{2}$, but equal to the next state duration of 1. So even though B3 is present as a pitch in the next state H'_6 , beat 3 of bar 2, it will be realised as a minim followed by a crotchet, rather than being held for a full bar. These answers are intended to indicate how a nonhomophonic texture can arise from generated output.

The key signature, time signature, tempo, and initial bass note of the passage shown in Fig. 8.7 are borrowed from op.56 no.1, the opening of which was subject to analysis in Sec. 7.3.1. The corresponding information from the chosen initial state could have been used instead (H'_0 in (8.21) is from op.24 no.2), but the analysis from Sec. 7.3.1, and hence op.56 no.1, will be called upon in Sec. 9.3.

8.5 Periodic and absorbing states

The following definitions complete my formalisation of Markov chains in the context of stylistic composition. The first (periodic state) is important because it establishes how serendipitous repetitions can occur in realised output of a random generation Markov chain (RGMC). The second definition (absorbing state) motivates strategies for revising random choices, in the event that continuation of the RGMC is not possible. The management of absorbing states is a principal aspect of the models described in Chapter 9.

Definition 8.8. Periodic state. Let (I, L, A) be a Markov model, and $(X_n)_{n \geq 0}$ a Markov chain based on this model. The state $i \in I$ is said to have period d if visits to i by the chain can only occur in d time-step multiples. Formally

$$d = \gcd\{n \geq 0 : \mathbb{P}(X_n = i \mid X_0 = i) > 0\}, \quad (8.25)$$

where \gcd is the greatest common divisor. ■

Periodic states can give rise to serendipitous repetitions in realised output. For instance, if state $i_0 \in I$ has period $d = 3$, and $X_0 = i_0, X_1 = i_1, X_2 = i_2, X_3 = i_0, X_4 = i_1$, and $X_5 = i_2$, then the realised output will contain two occurrences of a pattern corresponding to i_0, i_1, i_2 .

Definition 8.9. Absorbing state. Let (I, L, A) be a Markov model, let $(X_n)_{n \geq 0}$ be a Markov chain based on this model, and let $i_k \in I$ be an arbitrary state. If the corresponding element L_k of the transition list L , as given in (8.18), is such that nothing proceeds from i_k , then i_k is said to be an *absorbing state*. If during a random generation Markov chain (RGMC, cf. Def. 8.6), the random variable X_k takes the value i_k , then the chain is said to be *absorbed*, as no random selection for X_{k+1} is possible.

In Sec. 3.3 and earlier in this chapter, Markov models were specified in terms of a transition matrix \mathbf{P} . The equivalent definition of an absorbing state $i_k \in I$ is that the corresponding row $(p_{k,i})_{i \in I}$ of \mathbf{P} contains only zeros.

For reasons that will be elaborated upon in the next chapter, it could be that state i_k has a number of possible continuations, but it is not possible to find an acceptable continuation subject to certain constraints. The term absorbing state is also used in such a scenario. ■

Suppose that the melody in Fig. 8.1 had ended with the pitch class G♯ instead of G♭. Then the transition matrix \mathbf{P} in (3.19) would contain a new row of zeros corresponding to G♯, as there are no transitions from this pitch class to any other in the melody. (This is equivalent to a transition list entry L_k from (8.18) with nothing proceeding from i_k , that is, $l_k = 0$.) A state such as G♯ is known as an absorbing state. During an RGMC process, it is possible for an absorbing state $i_k \in I$ to be generated, where $0 \leq k \leq N - 1$. If the RGMC process was intended to generate N states and $k < N - 1$, then the process has stopped prematurely. It is possible to restart the process from stage $k - 1$, using the next pseudo-random number to choose i'_k as an alternative continuation to i_{k-1} . The same outcome could arise, however, giving $i'_k = i_k$, either by chance or because i_{k-1} to i_k is the only observed transition. There are other ways to manage absorbing states, such as altering the zero rows so that continuation becomes possible.

In the interests of achieving generated output, one might be prepared to restart a prematurely absorbed process c_{absb} times at stage k , before reverting to stage $k - 1$ and revising the choice for the corresponding state i_{k-1} . If the intention is that the output of RGMC should consist of N states, then depending on the number of restarts, $M \geq N$ pseudo-random numbers may not be sufficient for achieving a generated output. After using M pseudo-random numbers, not achieving generated output $(i_n)_{0 \leq n \leq N-1}$ does not imply that no such sequence of states exists. When constraints are placed on RGMC however, as they will be below in terms of sources, range etc., then it is possible no sequence of states $(i_n)_{0 \leq n \leq N-1}$ exists that satisfies these constraints. This enhanced RGMC process—allowing c_{absb} restarts at each stage in the event

of premature absorption or constraints not being satisfied—can be thought of as a search (Mitchell, 1997). The objective of the search is to find a member sequence $(i_n)_{0 \leq n \leq N-1}$ of the set of all such sequences satisfying the constraints, but it is not known a priori whether or not this set is empty.

Application: Guiding the generation of stylistic compositions

This chapter begins by pointing out and proposing solutions to stylistic shortcomings evident in Fig. 8.7; shortcomings that are typical of music generated by random generation Markov chain (RGMC). In Sec. 9.2, previous definitions and ideas are brought together to define a model called Racchman-Oct2010 (acronym explained in due course). Section 9.3 addresses pattern inheritance. I describe and demonstrate a model (Racchmaninof-Oct2010) that ensures a generated passage contains repeated patterns, inherited on an abstract level from an existing template piece.

9.1 Stylistic shortcomings associated with random generation Markov chains

The stylistic shortcomings of music generated by RGMC have been pointed out before: while first-order Markov modelling ‘ensures beat-to-beat logic in new compositions, it does not guarantee the same logic at higher levels... phrases simply string together without direction or any large-scale structure’ (Cope, 2005, p. 91). The passage shown in Fig. 8.7 is arguably too short

to be criticised for lacking global structure, but other shortcomings are as follows.

9.1.1 Sources

Too many consecutive states come from the same original source. The numbers above each state in Fig. 8.7 are the opus, number within that opus, and the bar number in which the state occurs. For instance, in bars 1-3 of Fig. 8.7, five consecutive states herald from op.17 no.4, and then seven from op.17 no.2. Having criticised the output of EMI for bearing too much resemblance to original Chopin (in the discussion of Figs. 5.13 and 5.14), steps should be taken to avoid the current model being susceptible to the same criticism.

The use of hand-coded rules (or constraints) to guide the generation of a passage of music was mentioned in Sec. 5.2 (p. 92), in relation to Ebcioğlu (1994) and Anders and Miranda (2010). I questioned whether such systems alone are applicable beyond relatively restricted tasks in stylistic composition. Random generation Markov chains (RGMC), on the other hand, seem to be appropriate for modelling open-ended stylistic composition briefs. There is a role for constraints to play, however, in solving some of the stylistic shortcomings of RGMC outlined above. For instance, a rule that prohibits more than c_{src} consecutive states having the same source would go some way towards preventing a generated passage replicating an original Chopin mazurka. The rule does not remove entirely the possibility of replication, but only an exhaustive search of the database of thirty-nine mazurkas would do that. For example, the boxed material in Fig. 8.7 is the same (up to

transposition) as op.50 no.2, beginning bar 10. The boxed material is derived from states that change source (op.67 no.4, op.41 no.1, op.50 no.2), but coincidentally, these states result in replication.

9.1.2 Range

The passage in Fig. 8.7 begins in the top half of the piano’s range, due to stipulating the initial bass note as E4. But this is from op.56 no.1, so some mazurkas do begin this high. A four-note chord built on top of this bass note contributes to the sense of an unusually high opening. Therefore, a solution to this shortcoming ought to be sensitive to positions of both lowest- and highest-sounding notes in a chord. An awareness of the distribution of notes within that chord—spread evenly, or skewed towards the top, for instance—may also be useful. The monitoring of lowest-sounding notes would, presumably, prohibit the plummet to a chord with lowest note A0 in bar 7. This chord is preceded by a state consisting of a single note (C \sharp 2, while the right hand rests). In the current model, a relatively large number of such single-note states, with a large number of possible continuations, helps to guard against replication of source material. The downside is that among a large number of continuations, there will be some that result in stylistic problems.

Can the range problem be addressed in a manner analogous to the sources problem, that is by fixing parameters for the lowest- and highest-sounding notes? I would advise against such a proposal: if the constraint is too narrow, an attentive listener will notice that the music never leaves a certain range, compared to a Chopin mazurka; if relaxed to allow a wider range,

the constraint becomes redundant. The key phrase, arguably, is *compared to a Chopin mazurka*. The position of the lowest-sounding note in a Chopin mazurka can be tracked over time, and, whilst it is being generated, so can the lowest-sounding note of a passage. I propose a constraint that the absolute difference in semitone steps between these lowest notes remains below the parameter c_{\min} . If, at stage k , this is not the case, then an alternative continuation for stage k will be selected, or reversion to stage $k - 1$ may be required. A similar parameter c_{\max} tracks the difference between highest-sounding notes. The spread of notes within a minimal segment is tracked as well. The parameter \bar{c} is responsible for this, controlling the permissible absolute difference between the mean MIDI note number of a minimal segment from a Chopin mazurka and the mean MIDI note number at the same time point of a passage being generated.

9.1.3 Low-lielihood chords

Monitoring the introduction of new pitch material in a probabilistic fashion is one way of quantifying and thus controlling what may be perceived as a lack of tonal direction. For example, MIDI note numbers corresponding to the pitches B \sharp 4, E \sharp 4, and A \sharp 4 appear in bar 5 of Fig. 8.7 for the first time. Using a local empirical distribution, formed over the current ontime and a certain number of preceding beats, it is possible to calculate the likelihood of the chords that appear in bar 5, say (cf. Sec. 3.1). If the empirical likelihood of any chord is too low, then this could identify a stylistic shortcoming. Low likelihood alone may not be the problem, as a Chopin mazurka might contain several such chords: the temporal position of these chords within an excerpt

will be appropriate to altering or obscuring the tonal direction, however.

As with the range problem, the issue of low-likelihood chords appearing at inappropriate points may be avoided by using a comparative constraint. In order to arrive at a comparative constraint for low-likelihood chords, a likelihood profile (cf. Def. 3.5) is constructed for an excerpt of a Chopin mazurka, showing how the likelihood of minimal segments (cf. Def. 8.3) varies over time. The same can be done for a passage as it is being generated by RGMC. The two curves, or profiles, are compared, and if the absolute difference between these curves remains below the parameter c_{prob} , then the constraint pertaining to low-likelihood chords is satisfied.

Underneath the excerpt shown in Fig. 9.1A is a plot, Fig. 9.1B, of two likelihood profiles. A likelihood profile is a plot of the geometric mean likelihood $\bar{L}(S, t, c_{\text{beat}})$ of each minimal segment $S \subseteq D$ against ontime. Modelling of musical expectation is a complex affair, but local minima in a likelihood profile should indicate at least some unexpected or surprising moments in the corresponding excerpt of music (cf. Sec. 3.2). This excerpt was chosen because, for me, the F \sharp octave in bar 7 is a low-likelihood chord, even after repeated listening. The profile for $c_{\text{beat}} = 12$ (dashed line) has its global minimum at this point. The two profiles are coincident up to ontime 6, but diverge from this point, as slightly different empirical distributions are employed to calculate likelihood—one distribution looks back over approximately two bars of music ($c_{\text{beat}} = 6$), and the other over approximately four bars ($c_{\text{beat}} = 12$). The general downward trend at the beginning of the curve is due to the empirical distribution expanding to its specified purview (window size).

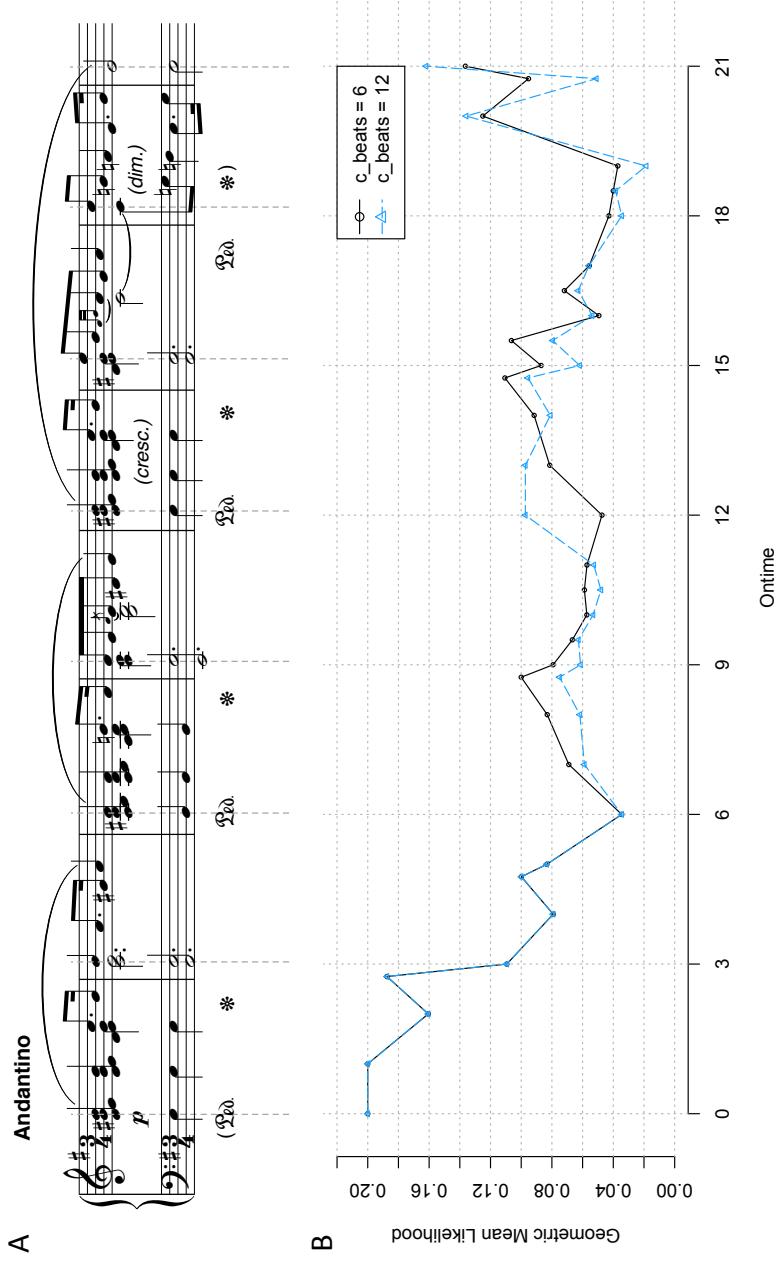


Figure 9.1: (A) Bars 1-8 of the Mazurka in E minor op.41 no.2 by Chopin; (B) Two likelihood profiles for the excerpt in Fig. 9.1, for different values of the parameter $c_{\text{beat}} = 6, 12$. A likelihood profile is a plot of the geometric mean likelihood of minimal segments against ontime.

When a passage is generated by random generation Markov chain (RGMC, cf. Def. 8.6) with constraints for *absorptions* (c_{absb}) and *consecutive sources* (c_{src}), and *comparative constraints* for *range* (c_{\min} , c_{\max} , and \bar{c}) and *low-likelihood chords* (c_{prob} , c_{beat}), it is necessary to use certain extra information, which can be taken from an existing excerpt of music.

Definition 9.1. Template. For an existing excerpt of music, a *template* consists of the following information:

- Tempo.
- Key signature.
- Time signature.
- Pitch of the first minimal segment's lowest-sounding note.
- Partition points (cf. Def. 8.3).
- Lowest- and highest-sounding, and mean MIDI note numbers at each partition point.
- Geometric mean likelihood at each partition point (likelihood profile).

■

For example, in Fig. 9.2B, the tempo, key signature, and time signature are retained from Fig. 9.2A, as is E4—the pitch of the first minimal segment's lowest-sounding note. Pseudo-plots of lowest- and highest sounding, and mean MIDI note numbers against ontine are indicated in Fig. 9.2B by the solid black lines passing through grey noteheads. The solid line in Fig. 9.2C is a likelihood profile for the excerpt from Fig. 9.2A. The use of a template

of some description as a basis for composition is discussed by N. Collins (2009, p. 108) and Hofstader (writing in Cope, 2001), who coins the verb to ‘templagiarise’ (p. 49). I would argue that the term *plagiarise* is too negative, except when the composer (or algorithm for music generation) uses: (1) a particularly well-known piece as a template and does little to mask the relationship; (2) too explicit a template (even if the piece is little known), the result being heavy quotation from the musical surface.

As an example of (1), the second movement of EMI’s *Sonata after Beethoven* is derived from that of Beethoven’s Sonata no.14 in C♯ minor, ‘Moonlight’, op.27 no.2. The discussion of EMI’s Mazurka no.4 in E minor (in relation to Figs. 5.13 and 5.14) serves as an instance of (2). This is not to say the use of a template is always problematic. If the meaning of ‘template’ is unambiguous (as in Def. 9.1) and the information contained in the template is employed abstractly (as with comparative constraints), then passages generated by this method can be stylistically successful without being accused of plagiarism. In the current model an extra precautionary step is taken of removing the piece selected for template construction from the database used to form the transition list. Only by coincidental replication, therefore, can a generated passage quote from the template piece.

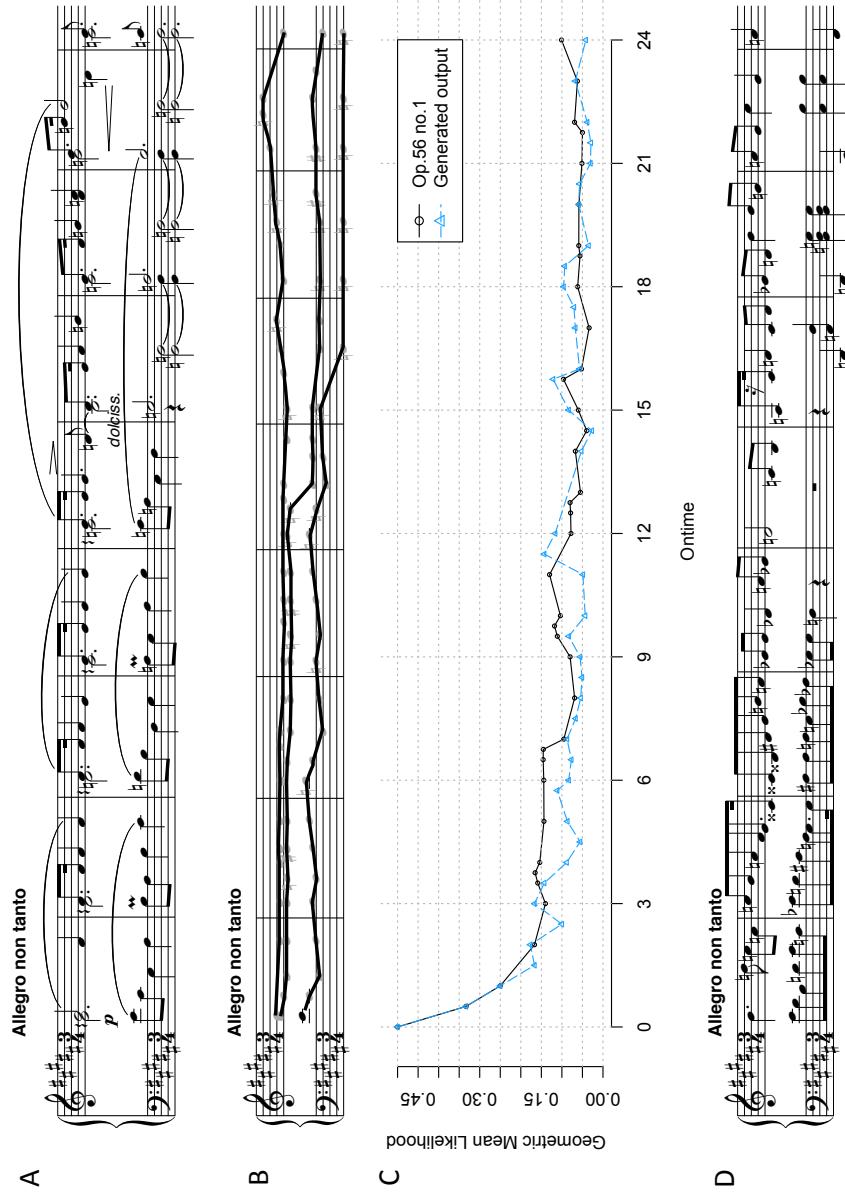


Figure 9.2: (A) Bars 1-9 of the Mazurka in B major op. 56 no.1 by Chopin; (B) Pseudo-plots of lowest- and highest sounding, and mean MNNs against ontine are indicated by black lines passing through grey noteheads; (C) Two likelihood profiles, one for the excerpt in Fig. 9.2A, and one for the passage in Fig. 9.2D; (D) Realised generated output of an RGMC for the model ($I^{(4)}$, $L^{(4)}$, $A^{(4)}$), with constraints applied to sources, range, and likelihood profile.

The generated passage shown in Fig. 9.2D satisfies the constraints without replicating the template piece (the excerpt from Chopin's op.56 no.1 shown in Fig. 9.2A).¹ One possible criticism of Fig. 9.2D is that bars 1-4 are too chromatic for the opening of a Chopin mazurka, so perhaps c_{prob} should be reduced, as this parameter controls the permissible difference between the template piece's likelihood profile and that of the generated output. These likelihood profiles are indicated by the solid and dashed lines respectively in Fig. 9.2C. The effect of the constraints is evident on comparing Fig. 9.2D to Fig. 8.7.

9.1.4 A sense of departure and arrival

The passage in Fig. 8.7 outlines a IV-I progression in bars 1-4, thus the first half of the passage conveys a sense of departure and arrival, albeit serendipitously. The move towards D minor in bars 7 and 8, on the other hand, does not convey the same sense of arrival. Stipulating a final bass note, in addition to stipulating the initial bass note of E4, would have increased the chance of the passage ending in a certain way. Students of chorale harmonisation are sometimes encouraged to compose the end of the current phrase first, and then to attempt a merging of the forwards and backwards processes, as indicated in Fig. 9.3. Cope (2005, p. 241) has also found the concept of composing backwards useful.

The remaining solution to be implemented in this section is the forwards-and backwards-generating process, which, it is proposed, will impart generated passages with a sense of departure and arrival. The idea of a backwards

¹The parameter values were $c_{\text{absb}} = 10$, $c_{\text{src}} = 3$, $c_{\text{min}} = c_{\text{max}} = 7$, $\bar{c} = 12$, $c_{\text{prob}} = .1$, and $c_{\text{beat}} = 12$.

The figure consists of three vertically aligned musical staves, labeled A, B, and C from top to bottom. Each staff has a treble clef, a key signature of one flat (B-flat), and a common time signature. Staff A shows the soprano and alto voices. Staff B shows the tenor and bass voices. Staff C shows both the soprano/alto and tenor/bass voices together. The music begins with a half note in the soprano, followed by quarter notes in the alto, tenor, and bass. The bass continues with eighth-note patterns. The soprano and alto have sixteenth-note patterns. The tenor and bass also have sixteenth-note patterns. The bass concludes with a sustained note.

Figure 9.3: Bars 1-2 of the chorale ‘Herzlich lieb hab ich dich, o Herr’, as harmonised (R107, BWV245.40) by J. S. Bach; (A) The three systems demonstrate how the excerpt might have been composed, starting with the cadence; (B) Working forwards from the beginning and backwards from the phrase’s end; (C) Merging the forwards and backwards processes.

Markov process was mentioned at the bottom of Def. 3.7—the practicalities are addressed below. Up to this point, a list of states $A^{(4)}$ from the beginning of each mazurka in the database has been used to generate an initial state. This list is referred to as the external initial states, now denoted $A_{\rightarrow}^{(4)}$. When generating backwards, a list $A_{\leftarrow}^{(4)}$ of external final states—that is, a list of states from the end of each mazurka in the database—may be appropriate. If, however, the brief were to generate a passage from bar one up to the downbeat of bar nine, then states from the very end of each mazurka are unlikely to provide stylistically suitable material for bar nine of a generated passage. Another list $A_{\leftarrow}^{(4)}$ of internal final states is required. This list contains three beat-spacing states (where these exist) from each mazurka in the database, taken from the time points at which the first three phrases are marked as ending in the score (Paderewski, 1953). For bar nine, say, of a generated passage, the internal final states will probably provide more stylistically suitable material than the external final states. The list $A_{\rightarrow}^{(4)}$ of internal initial states is defined similarly: it is a list consisting of three beat-spacing states (where these exist) from each database mazurka, taken from time points corresponding to the beginning of phrases two, three, and four. The internal initial states would be appropriate if the brief was to generate a passage from bar 9 onwards, say.

My four-step process for trying to ensure that a generated passage imparts a sense of departure and arrival is as follows. Let us assume the brief is to generate a passage from ontine x_1 to ontine x_2 , and let $x_{1|2} = \lceil (x_1 + x_2)/2 \rceil$.

- Use a forwards RGMC process with a template and constraints to generate c_{for} lots of output, realised as the datasets $D_1^{\rightarrow}, D_2^{\rightarrow}, \dots, D_{c_{\text{for}}}^{\rightarrow}$, all

of which are candidates for occupying the time interval $[x_1, x_{1|2}]$.

- Use a backwards RGMC process with the analogous template and constraints to generate c_{back} lots of output, realised as the datasets $D_1^{\leftarrow}, D_2^{\leftarrow}, \dots, D_{c_{\text{back}}}^{\leftarrow}$. These are candidates for occupying the time interval $[x_{1|2}, x_2]$.
- Consider all possible combinations of passages constructed by appending D_i^{\rightarrow} and D_j^{\leftarrow} , where $1 \leq i \leq c_{\text{for}}$ and $1 \leq j \leq c_{\text{back}}$, and then either (1) removing the datapoints of D_j^{\leftarrow} at ontime $x_{1|2}$, (2) removing the datapoints of D_i^{\rightarrow} at ontime $x_{1|2}$, or (3) superposing the datapoints of $D_i^{\rightarrow}, D_j^{\leftarrow}$. So, there will be $3 \times c_{\text{for}} \times c_{\text{back}}$ candidate passages in total.
- Of the $3c_{\text{for}}c_{\text{back}}$ candidate passages, select the passage whose states are all members of the transition list and whose likelihood profile is, on average, closest to that of the template piece.

9.2 RAndom Constrained CHain of MArkovian Nodes (Racchman)

This section brings together several previous definitions. The result is a model named Racchman-Oct2010, standing for RAndom Constrained CHain of MArkovian Nodes.² A date stamp is added in case it is superseded by future work. Racchman-Oct2010 is one of the models evaluated in Chapter 10, for the brief of composing the opening section of a Chopin mazurka (p. 94).

²The term *node* is a synonym of *vertex*, and is a reminder that the generation process can be thought of as walks in a digraph, such as Figs. 5.9 and 5.11B.

The definitions brought together are the random generation Markov chain (RGMC, Def. 8.6), the beat-spacing Markov model for Chopin mazurkas (Def. 8.7), and template (Def. 9.1). The discussion of absorbing states and restarting a RGMC (Sec. 8.5), and constraints for absorptions, sources, range, and low-likelihood chords (Secs. 9.1.1-9.1.3) are relevant to the following definition as well. Also, Racchman-Oct2010 uses the four-step process from Sec. 9.1.4 for trying to ensure that a generated passage imparts a sense of departure and arrival.

Definition 9.2. The RAndom Constrained CHain of MArkovian Nodes (Racchman-Oct2010) is an RGMC with the state space $I^{(4)}$ and transition list $L^{(4)}$ from Def. 8.7. It has four lists $A_{\rightarrow}^{(4)}$, $A_{\leftarrow}^{(4)}$, $A_{\downarrow}^{(4)}$, and $A_{\uparrow}^{(4)}$ for generating internal or external initial states as appropriate (cf. discussion in 9.1.4). At each stage $0 \leq n \leq N - 1$ of the RGMC, the generated output is realised and tested for the constraints pertaining to sources, range, and low-likelihood chords. If at an arbitrary stage $0 \leq k \leq N - 1$ any of these constraints are not satisfied, the RGMC is said to have reached an absorbing state, and an alternative continuation based on stage $k - 1$ is selected and retested, etc. If the constraints are not satisfied more than c_{absb} times at stage k , the state at stage $k - 1$ is removed and an alternative continuation based on stage $k - 2$ is selected and retested, etc. The RGMC continues until either: the generated output—when realised—consists of a specified number of beats, in which case the generated output is realised and stored as one of the candidate passages (see Sec. 9.1.4); or the constraints are not satisfied more than c_{absb} times at stage $k = 0$, in which case the RGMC is restarted. ■

Example output of Racchman-Oct2010 is given in Fig. 9.4.³ Figures 9.4A and 9.4B are realised output of a forwards RGMC process, and are candidates for occupying a time interval [0, 12]. Figures 9.4C and 9.4D are realised output of a backwards RGMC process, and are candidates for occupying a time interval [12, 24]. With the backwards process, the template contains the pitch of the last minimal segment's lowest-sounding note, as opposed to the first. Definitions of the transition list and likelihood profile are also reversed appropriately. As there are two candidates for each time interval ($c_{\text{for}} = c_{\text{back}} = 2$), and three ways of combining each pair of candidates (as described in the penultimate point above), there are $12 = 3c_{\text{for}}c_{\text{back}}$ candidate passages in total. Of these twelve passages, it is the passage shown in Fig. 9.4E whose states are all members of the transition list and whose likelihood profile is, on average, closest to that of the template.

The difference between Figs. 9.2D and 9.4E is a degree of control, in the latter case, over the sense of departure and arrival, due to the combination of forwards and backwards processes, and the extra pitch constraint for the last lowest-sounding note. The excerpt used as a template for both Figs. 9.2D and 9.4E is bars 1-9 of op.56 no.1, as shown in Fig. 9.2A. At the end of this excerpt, there is a pedal consisting of G2 and D2, which in the full piece persists for a further three bars, followed by chord V⁷ in bars 13-15 and chord I in bar 16. Arguably therefore, the template itself lacks a sense of arrival in bar 9, and this is reflected better in Fig. 9.4E, ending with chord ivb, than in Fig. 9.2D, which cadences on to chord v.

³The parameter values were $c_{\text{absb}} = 10$, $c_{\text{src}} = 4$, $c_{\text{min}} = c_{\text{max}} = 10$, $\bar{c} = 16$, $c_{\text{prob}} = .15$, and $c_{\text{beat}} = 12$.

A [Allegro non tanto]

B =

C = 5

D = 5

E =

Figure 9.4: (A) Passage generated by a forwards random generation Markov chain (RGMC); (B) Another passage from a forwards RGMC; (C) Passage generated by a backwards RGMC; (D) Another passage from a backwards RGMC; (E) There are three ways of merging each pair of forwards and backwards candidates. Of the twelve possible passages for this example, the passage shown has states that are all members of the transition list and a likelihood profile closest, on average, to that of the template.

9.3 Pattern inheritance

One of the criticisms of random generation Markov chains (RGMC) is that the resultant music lacks large-scale structure (Cope, 2005). As an example, ‘[i]n music, what happens in measure 5 may directly influence what happens in measure 55, without necessarily affecting any of the intervening material’ (ibid., p. 98). When developing the model Racchman-Oct2010, no attempt was made to address this criticism: any structure—local or global—that the listener hears in the generated passages of Figs. 8.7, 9.2D, and 9.4 has occurred serendipitously. The model Racchman-Oct2010 is not alone in ignoring matters of structure, for ‘[t]he formalization of music has not always covered so readily the form of music, particularly from a psychological angle that takes the listener into account’ (Collins, 2009, p. 103).

In this section, the matter of structure is addressed by a second model, Racchmaninof-Oct2010, standing for RAndom Constrained Chain of MArkovian Nodes with INheritance Of Form.⁴ As the name suggests, the only difference between this second model and the first model, Racchman-Oct2010, is the second model tries to ensure that discovered patterns from the template piece are inherited by the generated passage. Racchmaninof-Oct2010 comprises several runs of the simpler model Racchman-Oct2010, runs that generate material for ontime intervals, according to the rating of repeated patterns and until the ontime interval for an entire passage is covered. The pattern discovery algorithm SIACT is applied to a projection (ontime, MIDI note number, and morphetic pitch number) of the excerpt being used as a template and the results are filtered, as described in Sec. 7.3.1 (p. 176). The

⁴The term *pattern* would have been preferable to *form*, but Racchmaninop does not have the same ring.

resulting patterns are rated by the perceptually validated formula (6.4) and labelled in order of rank, so that pattern $P_{i,j}$ is rated higher than $P_{k,l}$ if and only if $i < k$. The second subscript denotes occurrences of the same pattern in lexicographic order. That is, pattern $P_{i,j}$ occurs before $P_{i,l}$ if and only if $j < l$. An example of the output of the discovery process was shown in Fig. 7.2. After filtering, three discovered patterns were left: pattern $P_{1,1}$ (indicated by the solid blue line) is rated higher than pattern $P_{2,1}$ (indicated by the dashed green line), which in turn is rated higher than $P_{3,1}$ (indicated by the dotted red line). The strengths and shortcomings of these results were discussed in Sec. 7.3.1—here I am more concerned with the application of the results to stylistic composition. Also discussed in Sec. 7.3.1 was the representation of the discovered patterns as a digraph, with an arc leading from the vertex for pattern $P_{i,j}$ to the vertex for $P_{k,l}$ if and only if $P_{i,j} \subset P_{k,l}$. The corresponding graph for the discovered patterns shown in Fig. 7.2 was given in Fig. 7.3. The position of each vertex is immaterial, but it is helpful to place each vertex horizontally at the ontine where the corresponding pattern begins, and vertically by pattern ranking. The total number of vertices emanating from a pattern’s vertex was defined as that pattern’s *subset score*, denoted \S . For instance, pattern $P_{3,3}$ has a subset score of $\S(P_{3,3}) = 2$, whereas pattern $P_{3,1}$ has a subset score of $\S(P_{3,1}) = 0$.

In the second model for generating stylistic compositions, Racchmaninoff-Oct2010, an attempt is made to ensure that the same type of patterns discovered in the template excerpt occur in a generated passage. In intuitive terms, the location but not the content of each discovered pattern is retained, as indicated in Fig. 9.5. For a generated passage, it should be possible to anno-

tate the score correctly with these same boxes, meaning that the discovered patterns have been inherited.

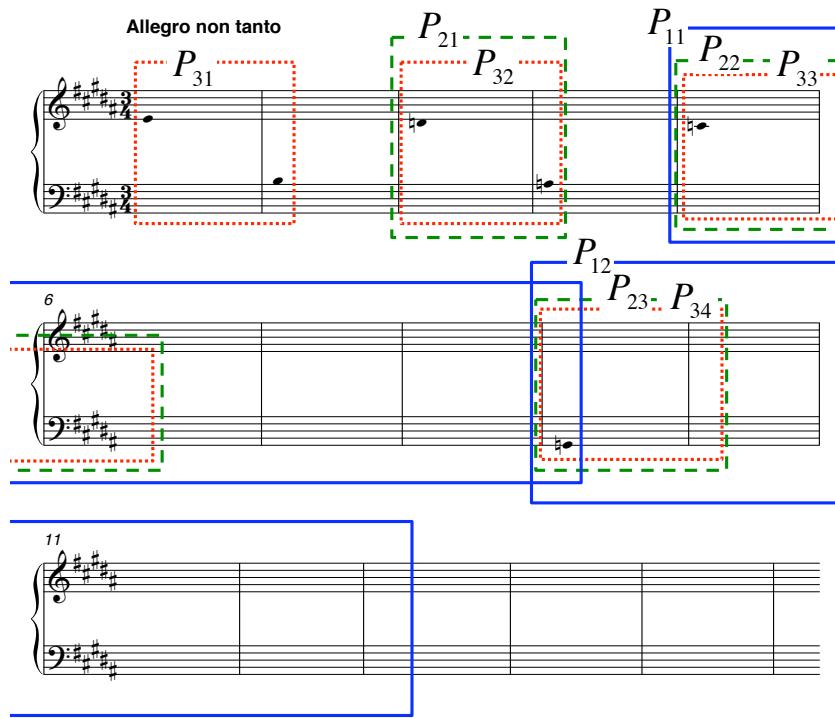


Figure 9.5: A representation of the supplementary information retained in a *template with patterns*. For comparison, an ordinary template (cf. Def. 9.1) is represented in Fig. 9.2. Most of the content of the excerpt from op.56 no.1 has been removed, but the location of the discovered patterns remains.

Definition 9.3. Template with patterns. The term *template* was the subject of Def. 9.1. The phrase *template with patterns* is used to mean that the following supplementary information is retained when patterns $P_{1,1}, P_{2,1}, \dots, P_{M,1}$ have been discovered (algorithmically) in an excerpt. For each discovered pattern $P_{i,1}$, retain:

- The ontimes of the first and last datapoints. For the sake of simplicity, these are rounded down and up respectively to the nearest integer.
- Its translators $\mathbf{v}_{i,2}, \mathbf{v}_{i,3}, \dots, \mathbf{v}_{i,m_i}$ in D , which bring $P_{i,1}$ to the subsequent occurrences $P_{i,2}, P_{i,3}, \dots, P_{i,m_i}$.
- The lowest-sounding pitch of the first and last minimal segments of the region in which the discovered pattern $P_{i,j}$ occurs ($j \neq 1$ if the algorithm discovered an occurrence other than the first).
- The subset score, denoted $\S(P_{i,1})$, which is the number of other discovered patterns of which $P_{i,1}$ is a subset. The scores $\S(P_{i,2}), \S(P_{i,3}), \dots, \S(P_{i,m_i})$ are retained also. ■

With reference to Fig. 9.6, it is demonstrated how this supplementary information is employed in the generation of a passage. The passage to be generated can be thought of as an open interval of ontimes $[a, b] = [0, 45]$, the same length as the excerpt chosen for the template (op.56 no.1). When the set of intervals U for which material has been generated *covers* the interval $[a, b]$, the process is complete. At the moment this set is empty, $U = \emptyset$.

1. Generation begins with the pattern $P_{i,j}$ that has maximum subset score $\S(P_{i,j})$. Tied scores between $\S(P_{i,j})$ and $\S(P_{k,l})$ are broken by highest rating ($\min\{i, k\}$) and then by lexicographic order ($\min\{j, l\}$). It is evident from the graph in Fig. 7.3 that $P_{3,3}$ has the maximum subset score. The ontimes of the first and last datapoints have been retained in the template with patterns, so it is known that material for the ontine interval $[a_1, b_1] = [12, 15]$ must be generated. This is done using the first model Rachman-Oct2010, with internal initial and final states, and the

lowest-sounding pitches retained in the template with patterns. The resulting music is contained in box 1 in Fig. 9.6. The set of intervals for which material has been generated becomes $U = \{[12, 15]\}$.

2. Having retained the nonzero translators of $P_{i,1} = P_{3,1}$ in D in the template with patterns, translated copies of the material generated in step 1 are placed appropriately, giving the music contained in boxes labelled 2 in Fig. 9.6. Now $U = \{[0, 3], [6, 9], [12, 15], [24, 27]\}$. It is said that patterns $P_{3,1}, P_{3,2}, P_{3,3}, P_{3,4}$ have been *addressed*.
3. Among the unaddressed patterns, generation continues with the pattern that has the highest subset score, in this example $P_{2,2}$. This pattern has corresponding ontime interval $[a_2, b_2] = [12, 15]$. As this interval is contained in U , no material is generated. (Had $[a_2, b_2] = [12, 17]$, say, then material would have been generated for $[15, 17]$ and connected to that already generated for $[12, 15]$. Had $[a_2, b_2] = [9, 17]$, say, then material for $[9, 12]$ and $[15, 17]$ would have been generated and connected either side of that for $[15, 17]$.) As ontime intervals for patterns $P_{2,1}$ and $P_{2,3}$ have also been covered, it is said that patterns $P_{2,1}, P_{2,2}, P_{2,3}$ have been addressed. Generation continues with the pattern $P_{1,1}$, as this is the remaining unaddressed pattern with the highest subset score. (In the example, occurrences of $P_{3,1}$ and $P_{2,1}$ have now been addressed, so $P_{1,1}$ and $P_{1,2}$ are the only choices.) Pattern $P_{1,1}$ has an ontime interval of $[a_3, b_3] = [12, 24]$. Now $[12, 15] \in U$, meaning that material must be generated for the remainder of this interval, $[15, 24]$. Again, the model Racchman-Oct2010 is used, and the resulting music is contained in box 3 in Fig. 9.6. As $[12, 15], [24, 27] \in U$, initial and

final states for the material to fill [15, 24] have been generated already.

This is illustrated by the overlapping of box 3 by surrounding boxes in Fig. 9.6. Now $U = \{[0, 3], [6, 9], [12, 15], [15, 24], [24, 27]\}$.

4. Having retained the nonzero translator of $P_{1,1}$ in D in the template with patterns, a translated copy of the material generated in step 3 is placed appropriately, giving the music contained in box 4 in Fig. 9.6. Now $U = \{[0, 3], [6, 9], [12, 15], [15, 24], [24, 27], [27, 36]\}$.
5. All patterns contained in the template have been addressed, but still U does not cover the whole passage's ontime interval $[a, b] = [0, 45]$. Material for the remaining intervals, $[a_4, b_4] = [3, 6]$, $[a_5, b_5] = [9, 12]$, and $[a_6, b_6] = [36, 45]$, is generated in this final step. The model Racchman-Oct2010 is used three times (once for each interval), and the resulting music appears in boxes labelled 5 in Fig. 9.6. The intervals [3, 6], [9, 12], and [36, 45] are included in U , and the process is complete, as U now covers $[a, b] = [0, 45]$.

The above list outlines an example run of the model I call Racchmaninof-Oct2010 (RAndom Constrained Chain of MArkovian Nodes with INheritance Of Form).⁵ The result is a Markov model where, in Cope's (2001) terms, it *is* possible for what happens at bar 5 to influence bar 55. For instance, comparing Figs. 7.2 and 9.6, the listener gets the impression that the locations—but not the content—of the discovered patterns have been inherited by the generated passage. The example run provides an impression of Racchmaninof's workings, but for completeness a formal definition follows.

⁵The parameter values were $c_{\text{absb}} = 10$, $c_{\text{src}} = 4$, $c_{\min} = c_{\max} = 12$, $\bar{c} = 19$, $c_{\text{prob}} = .2$, and $c_{\text{beat}} = 12$.



Figure 9.6: Passage generated by the model Racchmaninof-Oct2010, standing for RAndom Constrained Chain of MArkovian Nodes with INheritance Of Form. The numbered boxes indicate the order in which different parts of the passage are generated, and correspond to the numbered list after Def. 9.3. This passage is used in the evaluation in Chapter 10, as stimulus 29.

Definition 9.4. RAndom Constrained CHain of MArkovian Nodes with INheritance Of Form (Racchmaninof-Oct2010). Take an existing excerpt of music and apply SIACT (cf. Def. 7.2) to the projection of on-time, MIDI note number, and morphetic pitch number. Rate the discovered patterns according to the importance formula (6.4), and filter as described in Sec. 7.3.1. Retain information from the existing excerpt in a *template with patterns* (cf. Def. 9.3). Let $[a, b]$ be an open interval of ontimes, and U be the set of intervals (initially empty) for which material is generated.

1. If U covers $[a, b]$ then the process is complete.
2. Else, let $P_{i,j}$ be the unaddressed discovered pattern with maximal subset score, and let a_i and b_i be the ontimes of its first and last datapoints respectively. *Unaddressed* means that this pattern has not been considered on a previous iteration. If all discovered patterns have been addressed, put $a_i = a$ and $b_i = b$, go to step 3, after which U will cover $[a, b]$ and the process is complete.
3. Use Racchman-Oct2010 (cf. Def. 9.2) to generate material for each ontime interval in $[a_i, b_i]$ that is not already covered by U . Depending on previous iterations, initial and final states may be specified by surrounding generated material. If not, use the external/internal initial/final states as appropriate (cf. discussion in Sec. 9.1.4).
4. Insert copies of the generated material in locations specified by the nonzero translators of $P_{i,j}$, which are retained in the *template with patterns*. It is said that patterns $P_{i,1}, P_{i,2}, \dots, P_{i,m}$ have been addressed.
Go to step 1. ■

One may question why the generation ought to proceed according to subset scores. In short, doing so ensures the inheritance of nested patterns. Nested patterns were discussed earlier in relation to Fig. 7.4, where a discovered pattern $P_{1,1}$ contained two occurrences, $P_{2,1}$ and $P_{2,2}$, of another pattern. By definition, $\S(P_{2,1}) > \S(P_{1,1})$, and so, if proceeding by maximum subset score, material will be generated first for the interval corresponding to $P_{2,1}$. This material will be translated appropriately to address $P_{2,2}$, as well as any subsequent occurrences. Material will be generated second for the interval corresponding to $P_{1,1}$, if any of this interval remains unaddressed. In this way, it is ensured that in the generated passage, there is a pattern in the same location as $P_{1,1}$ that itself contains two occurrences of a pattern in the same locations as $P_{2,1}, P_{2,2}$. The alternative approach would be to address the interval corresponding to $P_{1,1}$ first, not second. When it comes to the second step of addressing the interval for $P_{2,1}$, this interval has been covered in the first step and no new material is generated. Therefore, the alternative approach does not guarantee that $P_{1,1}$ itself contains any patterns.

Some of the shortcomings of Racchmaninof-Oct2010 are mentioned now, as a prelude to more thorough evaluation in the next chapter. First, Racchmaninof-Oct2010 has no mechanism for ensuring that overlapping patterns are inherited (cf. pattern A in Figs. D.1 and D.2). Overlapping occurrences were removed by one of the filters applied to SIACT in Sec. 7.3.1. A mechanism for inheriting overlapping patterns could be introduced in a more intricate version. Second, Racchmaninof-Oct2010 cannot handle subtle variations on patterns (unlike the Bol Processor of Bel and Kippen, 1994). For example, there is a subtle difference between patterns $P_{3,2}$ and $P_{2,1}$ as shown in

Fig. 7.2, the latter containing two extra notes that create a dotted rhythm. A music analyst might call bar 3 a transposed variation of bar 1, presaged by the dotted rhythm in bar 2. In the generated passage, the location of pattern $P_{3,3}$ is addressed first, which defines material for the location of pattern $P_{3,2}$. No new material is generated for the location of pattern $P_{2,1}$, as the corresponding interval has been covered already. Hence, the subtle variation relationship between $P_{3,2}$ and $P_{2,1}$ is not inherited by the generated passage. Third, Racchmaninof-Oct2010 connects two previously unconnected intervals of a generated passage with mixed success. For example, bars 2 and 3 of Fig. 9.6 dovetail elegantly enough to mask the transposed repetition of bar 1 in bar 3. Less successful perhaps is the link from bar 3 to bar 4, where a held F4 creates a dissonance with E4 on the downbeat of bar 4. Most likely, the dissonance would be heard as enthusiastic legato on the performer's behalf (similarly at bars 12 to 13), but such chords do not appear in the database, so this is a slight problem with the model. Further comments on the models Racchman-Oct2010 and Racchmaninof-Oct2010 appear in the following chapter as part of the evaluation, where several passages generated by each model—including that shown in Fig. 9.6—feature among the stimuli.

10

Comparative evaluation of models of stylistic composition

10.1 Evaluation questions

This chapter consists of an evaluation of the models developed in Chapters 8 and 9 (Racchman-Oct2010 and Racchmaninof-Oct2010). The stylistic composition brief chosen as the subject of the evaluation is:

Chopin mazurka. Compose the opening section (approximately sixteen bars) of a mazurka in the style of Chopin.

This brief was introduced and discussed in Chapter 5 (p. 94). The purpose of the evaluation is to address the following questions:

1. How do mazurkas generated by the models described in Chapters 8 and 9 (Racchman-Oct2010 and Racchmaninof-Oct2010) compare in terms of *stylistic success* to:
 - Original Chopin mazurkas;
 - *Mazurkas, after Chopin* (Cope, 1997) attributed to EMI;
 - Mazurkas by other human composers?

2. Are judges able to distinguish between the different categories of music stimulus (e.g., human-composed or computer-based)? In particular, does a given judge do better than by chance at distinguishing between human-composed stimuli and those based on computer programs that learn from Chopin's mazurkas.
3. In terms of the stylistic success ratings for each stimulus, is there a significant level of interjudge reliability? What about interjudge reliability for other parts of the task, such as *aesthetic pleasure*?
4. For a given judge, is there significant correlation between any pair of the following: ratings of stylistic success; ratings of aesthetic pleasure; the categorisation of a stimulus as computer-based?
5. Are there particular aspects of a stimulus that lead to its stylistic success rating (high or low)? Are certain musical attributes useful predictors of stylistic success?

10.2 Methods for answering evaluation questions

The general framework for the evaluation is Amabile's (1996) Consensual Assessment Technique (CAT), as discussed in Sec. 5.7 (p. 122). In my particular version of the CAT, a judge's task involves giving ratings of stylistic success (Pearce and Wiggins, 2007) and aesthetic pleasure, and distinguishing between different categories of music stimulus (Pearce and Wiggins, 2001). Each of the evaluation questions can be cast as quantitative, testable hy-

potheses, apart from the first part of question 5, which was assessed using judges' open-ended textual responses. Subject to an appropriate level of interjudge reliability, question 1 will be answered using analysis of variance (ANOVA, cf. Example A.48, p. 323). The different systems for generating mazurkas will be represented by binary variables x_1, x_2, \dots, x_p , and for the mean stylistic success rating of a stimulus y , inferences will be made of the form

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p, \quad (10.1)$$

where $\alpha, \beta_1, \beta_2, \dots, \beta_p$ are coefficients to be estimated from the data. Testing the null hypothesis of no linear relationship,

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_p = 0, \quad (10.2)$$

will indicate the significance of the model in (10.1). Furthermore, for coefficients β_i and β_j , representing the relative stylistic success of mazurka-generating systems i and j , a test of the *contrast*

$$H_0 : \beta_j - \beta_i = 0 \quad (10.3)$$

will indicate whether the mazurkas generated by system j are rated as significantly different in terms of stylistic success to those of system i .

Question 2 will be answered by imagining that a judge guesses the category to which each stimulus belongs. Under these circumstances, it is possible to calculate a score s such that the probability of the 'guessing judge' achieving a score of s or higher is .05. Any judge that scores equal to or higher than s is said to be able to distinguish between different categories of music

stimulus.

Kendall's (1948) coefficient of concordance W can be used to assess agreement within a group of judges (question 3). Taking the judges' ratings of stylistic success, for example, the coefficient reflects overall interjudge reliability. Amabile (1996) uses the Spearman-Brown prediction formula (Nunnally, 1967), which seems to be more appropriate for considering the relationship between reliability and test length. Since Kendall's coefficient of concordance assesses reliability for a group of judges, it is also worth knowing whether particular pairs of judges' ratings are significantly positively correlated. The significance of such pairings will be investigated by simulation using Pearson's product moment correlation coefficient (cf. Def. A.25).

Simulation of Pearson's product moment correlation coefficient can also be used to answer part of question 4: for a given judge, is there significant positive correlation between ratings of stylistic success and aesthetic pleasure? Further, I am curious to know whether judges appear to be biased against what they perceive as computer-based stimuli. Moffat and Kelly (2006), for example, found evidence of this bias. One way to investigate this matter is to focus on judges' stylistic success ratings for those Chopin mazurkas that are misclassified by the judge as computer-based. Although the judge does not think that such a stimulus is a Chopin mazurka, the stimulus will bear many—if not all—of the stylistic traits of a mazurka, so should still receive at least a middling rating for stylistic success.

In addressing question 5, I will be suggesting how aspects of the models developed in Chapters 8 and 9 (Racchman-Oct2010 and Racchmaninoff-Oct2010) can be improved. To answer the first part of question 5—are there

particular aspects of a stimulus that lead to its stylistic success rating (high or low)?—I will undertake a textual analysis of the judges' open-ended responses. The responses will be grouped into six categories (pitch range, melody, harmony, phrasing, rhythm, and other), as per Pearce and Wiggins (2007). The other category will be reserved for comments that do not fit in one of the first five categories, or that are too vague. Each comment will also be categorised as positive, negative, or neutral, to identify the general aspects of the models from Chapters 8 and 9 that need most attention. To answer the second part of question 5—are certain musical attributes useful predictors of stylistic success?—quantifiable attributes will be determined for each stimulus, such as *chromaticism*, the number of non-key notes in a stimulus. There will be at least one attribute for each of the categories pitch range, melody, harmony, phrasing, and rhythm. Following the approach of Pearce and Wiggins (2007), a model useful for relating judges' ratings of stylistic success to the attributes will be determined using variable selection. Should any attributes emerge with significant negative coefficients, these attributes could be the basis for specific future improvements. It should be noted that a single incongruent chord, rhythm, or melodic leap can be responsible for reducing the rated stylistic success of a whole passage. If quantifying a musical attribute involves averaging over a passage, then it is important that single incongruities are not diluted.

10.3 Judges and instructions

The first group of judges (8 males and 8 females), referred to hereafter as *concertgoers*, were recruited at a concert containing music by Camille Saint-

Saëns (1835-1921) and Marcel Tournier (1879-1951) for violin and harp, which took place in St Michael's Church, The Open University, on 29 September, 2010. The second group of judges (7 males and 9 females), referred to hereafter as *experts*, were recruited from various email lists (JISCMail Music Training, and music postgraduate lists at the University of Cambridge; King's College, London; and the University of York). The expert judges were pursuing or had obtained a Master's Degree or PhD, and either played/sang nineteenth-century music or considered it to be one of their research interests.

Both the concertgoers (mean age = 59.65 years, $SD = 5.51$) and experts (mean age = 31.25, $SD = 11.89$) were paid £10 for an hour for their time, during which they were asked to listen to excerpts of music and answer corresponding multiple-choice and open-ended questions.¹ Judges participated one at a time, they were seated at a computer, and the instructions and subsequent tasks were presented using a graphical user interface. The instructions, which were the same for concertgoer and expert judges, began by introducing Chopin's mazurkas. Judges listened to two examples of Chopin mazurkas (op.24 no.1 and op.41 no.4, approximately the first sixteen bars) and were asked to comment verbally on musical characteristics that the excerpts had in common. I listened and responded to these comments to set judges at their ease, and to make sure that they were comfortable navigating the interface and playing/pausing the embedded sound files. Judges received these instructions for the main task:

'In the following task, you will be asked to listen to and answer
questions about short excerpts of music.'

¹The accompanying CD (or <http://www.tomcollinsresearch.net>) includes a copy of the instructions for judges, as well as the music stimuli used in the study.

- Some of these excerpts will be from *Chopin mazurkas*.
- Some excerpts will be from the work of *human composers*, but *not* Chopin mazurkas. For example, a fantasia by Mozart would fit into this category, as would an impromptu by Chopin, as would a mazurka by Grieg.
- Some excerpts will be based on *computer programs* that learn from Chopin's mazurkas.'

The last category includes the models described in Chapters 8 and 9, as well as *Mazurkas, after Chopin* by 'David Cope, with Experiments in Musical Intelligence' (Cope, 1997). The category is referred to hereafter as *computer-based* stimuli. Judges were warned that when distinguishing between categories, some of the computer-based stimuli were more obvious than others.

The instructions go on to point out that part of the task requires judges to distinguish between the three different categories above, another part of the task requires judges to rate a stimulus' stylistic success, and a further part requires judges to rate the aesthetic pleasure conferred by a stimulus. Working definitions of stylistic success and aesthetic pleasure were provided:

Stylistic success. An excerpt of music is stylistically successful as a Chopin mazurka if, in your judgement, its musical characteristics are in keeping with those of Chopin's mazurkas. Use the examples from the Introduction as a means of comparison, and/or any prior knowledge about this genre of music.

Suppose I know an excerpt is not a Chopin mazurka. Can it still be stylistically successful? Yes, if in your judgement its musical characteristics are in keeping with those of Chopin's mazurkas.

Suppose I know an excerpt is a Chopin mazurka. Can I give it anything other than the highest stylistic rating? Yes, if for any reason you judge it to be an unusual example of a Chopin mazurka.

Aesthetic pleasure. Would you be likely to add a recording of this piece to *your* music collection? It is fine to give low ratings for aesthetic pleasure, but please remember that you are listening to a synthesized piano sound, and try to imagine how much you might enjoy the excerpt if it was played expressively.

Ratings of stylistic success and aesthetic pleasure were elicited using a seven-point scale, with 1 for low stylistic success (or aesthetic pleasure) and 7 for high. For each stimulus, the three questions (distinguish, style, and aesthetic) were framed above by the embedded sound file and a question that checked whether the judge had heard an excerpt before, and below by a textbox for any other comments. There was also a textbox for comments specific to the rating of stylistic success. These questions will be referred to collectively as the *question set*.

The main task consisted of thirty-two stimuli. Judges were asked to calibrate their rating scales by listening to at least part of all the stimuli, presented on a single page (Amabile, 1996). By clicking *next*, judges met the embedded sound file for the first stimulus and the corresponding question set. Clicking *next* again, they moved on to the second stimulus and the second question set, etc. It was possible to listen to the same stimulus several times, to alter answers, and to revisit previous stimuli and the instructions.

The order of presentation of stimuli was randomised for each judge, and three different question orders were used (distinguish, style, aesthetic; style,

distinguish, aesthetic; aesthetic, distinguish, style) to mitigate ordering effects. Immediately prior to the main task, each judge completed the same short warm up task, responding to the question set for three excerpts. A judge's answers to the warm up task were reviewed before the main task, and it was disclosed that one of the warm up excerpts was a Chopin mazurka (op.6 no.2). The three Chopin mazurkas (two from the introductory instructions and one from the warm up task) were embedded in a menu to one side of the interface, so that at any point, a judge could remind themselves of the example mazurkas. The warm up task was intended to help judges to familiarise themselves with the format of the user interface, the question set, and the rating scale. It also gave them an opportunity to ask questions. The whole procedure (warm up and main tasks) had been tested in a pilot study and adjusted accordingly for ease of understanding and use.

10.4 Selection and presentation of stimuli

Stimuli were prepared as MIDI files with a synthesised piano sound. Each stimulus was the first forty-five beats from the selected piece, which is approximately fifteen bars in triple time. To avoid abrupt endings, a gradual fade was applied to the last nine beats. Depending on a judge's preference and the listening environment (I travelled to expert judges, rather than vice versa), stimuli were presented via external speakers (RT Works 2.0) or headphones (Performance Plus II ADD-800, noise cancelling). Several options were considered when preparing the sound files:

1. Exact MIDI, e.g. metronomically exact and dynamically uniform.

2. Expressive MIDI, e.g. with expressive timing and dynamic variation.
3. Audio recorded by a professional pianist.

Spiro, Gold, and Rink (2008) demonstrate that there is considerable variety in professional recordings of Chopin's mazurkas, especially with respect to rubato. In terms of option 3, I was concerned that using recordings by a pianist would introduce an uncontrollable source of variation, and that there may be some bias on the pianist's part—conscious or otherwise—against excerpts perceived as being computer-based. The second option is also problematic, as the computer-based excerpts do not bear expressive markings. Instead of using expressive markings, one could employ an algorithm for converting exact MIDI into expressive MIDI (Widmer and Goebl, 2004), but this would also introduce a source of variation (albeit controlled). For these reasons, option 1 was chosen for preparing the sound files. The tempo of each excerpt was retained, and where a piece did not have a tempo marking, the tempo of the framework or template piece was used. For instance, it was demonstrated in Sec. 5.6.2 (pp. 116-117) that Chopin's Mazurka in F minor op.68 no.4 is the most likely framework for the Mazurka no.4 in E minor of EMI. None of the above options 1-3 is ideal, so it would be worth making *format of sound file* a variable in the future. The shortcoming of option 1 is that metronomically exact and dynamically uniform MIDI sounds bland and mechanical, and as such, some of the *meaning* of the music is removed. To partly compensate for this, for the two mazurkas used in the introductory instructions, both audio and exact-MIDI versions were included. Judges were asked to consider the expressive differences between audio and exact MIDI, they were reminded that judgements should not involve the quality of the

sound recording, and they were encouraged to imagine how a stimulus would sound if performed expressively by a professional pianist.

The stimuli were prepared from the following pieces:

Chopin mazurka. Mazurkas by Chopin in:

1. B♭ minor op.24 no.4.
2. G major op.67 no.1.
3. A♭ major op.7 no.4.
4. F♯ minor op.59 no.3.
5. C♯ minor op.63 no.3.
6. B minor op.33 no.4.

Human other.

7. Mazurka in G minor from *Soirées musicales* op.6 no.3 by Clara Schumann.
8. Prelude in B major from Twenty-four Preludes op.28 no.11 by Chopin.
9. Romance in F major from Six Piano Pieces op.118 no.5 by Johannes Brahms.
10. Rondeau, ‘Les baricades mystérieuses’, from *Ordre* no.6 by François Couperin (1668-1733).
11. No.5 (Etwas rasch) from Six Little Piano Pieces op.19 by Arnold Schoenberg (1874-1951).

-
12. Mazurka in F major, ‘Michaela’s mazurka’, by David A. King.²

Computer-based.

EMI. *Mazurkas, after Chopin* (Cope, 1997) attributed to EMI. Mazurkas in:

- 13. A minor no.1.
- 14. C major no.2.
- 15. B \flat major no.3.
- 16. E minor no.4.
- 17. B \flat major no.5.
- 18. D major no.6.

System A. Passages generated by the model Racchman-Oct2010 as described in Chapter 9, with parameter values for number of absorptions permitted at each stage ($c_{\text{absb}} = 10$), for number of consecutive states heralding from the same source ($c_{\text{src}} = 4$), for constraining range ($c_{\text{min}} = c_{\text{max}} = \bar{c} = 19$), for constraining low-likelihood chords ($c_{\text{prob}} = .2$, and $c_{\text{beat}} = 12$), and for ensuring a sense of departure/arrival ($c_{\text{for}} = c_{\text{back}} = 3$). The Chopin mazurka used as a template is given in brackets. Mazurkas in:

- 19. C major (op.56 no.2).
- 20. E \flat minor (op.6 no.4).
- 21. E minor (op.41 no.2).
- 22. C minor (op.56 no.3).

²Retrieved 12 October, 2010, from <http://www.sibeliusmusic.com>. This is a site where amateur composers can publish music scores.

23. A minor (op.17 no.4).

24. F minor (op.63 no.2).

System B. Passages generated by the model Racchmaninof-Oct2010 as described in Chapter 9, with parameter values less than or equal to $c_{absb} = 10$, $c_{src} = 4$, $c_{min} = c_{max} = \bar{c} = 31$, $c_{prob} = 1$, and $c_{beat} = 12$, and $c_{for} = c_{back} = 3$. The Chopin mazurka used as a template is given in brackets. Mazurkas in:

25. C \sharp minor (op.50 no.3).

26. C major (op.67 no.3).

27. B major (op.41 no.3).

System B*. Passages generated by the model Racchmaninof-Oct2010 as described in Chapter 9, with parameter values less than or equal to $c_{absb} = 10$, $c_{src} = 4$, $c_{min} = c_{max} = \bar{c} = 24$, $c_{prob} = .2$, and $c_{beat} = 12$, and $c_{for} = c_{back} = 3$. Again, the Chopin mazurka used as a template is given in brackets. Mazurkas in:

28. C major (op.68 no.1).

29. B major (op.56 no.1).

30. F major (op.68 no.3).

31. A minor (op.7 no.2).

32. Ab major (op.24 no.3).

The main difference between Systems A, B, and B* is that Systems B and B* use pattern inheritance. The difference between Systems B and B* is that the parameter values of the latter are tighter, meaning that one would expect the judged stylistic success of System B* stimuli to be greater on average

than stimuli from System B. It should be noted as a result that Systems A and B* have comparable parameter values, whereas Systems A and B do not. Different numbers of stimuli per category for Systems B and B* are permissible for the chosen analytical methods: if after 2-3 hours, the algorithm implementing model Racchmaninof-Oct2010 had produced no output, the process was stopped, constraint parameter values were relaxed (increased), and the process was restarted. For stimuli from System B, parameters were relaxed to such an extent that the stimuli are not directly comparable with those of System A. One might speculate that the templates used for System B were atypical mazurkas, as it was relatively difficult to generate a passage that satisfied the comparative constraints. A mazurka section generated by System B* appeared in Fig. 9.6, and was used as stimulus 29. Further stimuli from Systems A, B, and B* are given in Appendix E.

The Chopin mazurkas (stimuli 1-6) were selected as being representative of the corpus. The database used by Systems A, B, and B* to generate passages did not contain any of these mazurkas. Otherwise, substantial between-stimuli references could have occurred. The template pieces were selected at random from the remaining mazurkas. For Systems A, B, and B*, the database used to generate a passage for stimuli n did not contain the template mazurka selected for stimuli n , to reduce the probability of replicating existing music.

The category *human other* is something of a catch-all. It contains two mazurkas by composers other than Chopin and a piece by Chopin that is not a mazurka. Non-mazurka music from a range of musical periods is also represented, from Baroque (Couperin) through Romantic (Brahms) to Twentieth

Century (Schoenberg).

10.5 Results

10.5.1 Answer to evaluation question 3

The analysis begins by answering question 3, as this determines how question 1 is approached. For the time being the concertgoers (judges 1-16) and experts (judges 17-32) will be kept separate. Considering ratings of stylistic success, Kendall's coefficient of concordance is significant for both the concertgoers ($W = .520$, $\chi^2(31) = 258$, $p = 1.26 \times 10^{-37}$) and the experts ($W = .607$, $\chi^2(31) = 301$, $p = 5.01 \times 10^{-46}$). Turning to pairwise correlations for judges' ratings of stylistic success, 102 of the 120 ($= 16 \times 15/2$) inter-judge correlations were significant at the .05 level for concertgoers.³

For the experts, 116 of the 120 inter-judge correlations were significant at the .05 level. One expert judge appeared in all four of the nonsignificant correlations. A higher proportion of expert judges' ratings are significantly correlated, compared to the proportion for the concertgoer judges, suggesting that it is appropriate to continue considering the two groups separately. The few judges that did not produce significantly correlated stylistic success ratings tended not to have made use of the full range of the rating scale. This does not seem a strong enough justification for removing any data.

³The p -values were calculated by simulation, as it is not possible to assume that a judge's ratings of stylistic success are normally distributed. For any pair of judges' ratings x_1, x_2, \dots, x_{32} and y_1, y_2, \dots, y_{32} , the following steps were repeated 1000 times: (i) randomly permute y_1, y_2, \dots, y_{32} to give $y_{i(1)}, y_{i(2)}, \dots, y_{i(32)}$; (ii) calculate Pearson's product moment correlation coefficient for the pairs $(x_1, y_{i(1)}), (x_2, y_{i(2)}), \dots, (x_{32}, y_{i(32)})$. The p -value is the proportion of correlation coefficients in step (ii) that are greater than the correlation of the original pairs $(x_1, y_1), (x_2, y_2), \dots, (x_{32}, y_{32})$.

10.5.2 Answer to evaluation question 1

As the ratings of stylistic success are mainly significantly positively correlated, it is reasonable to take the mean rating of stylistic success for each excerpt. These are shown in Table 10.1, along with the percentage of judges that classified each excerpt correctly (more of which in answer to question 2), and the percentage of judges that classified each excerpt as a Chopin mazurka. The first column of Table 10.1 gives the stimulus number. The details of each stimulus were given in Sec. 10.4; in brief, stimuli 1-6 are Chopin mazurkas, 7-12 are from the category human other, the rest are from the computer-based category, with stimuli 13-18 from EMI, 19-24 from System A, 25-27 from System B, and 28-32 from System B*. System A is an implementation of the Racchman-Oct2010 model. Systems B and B* are implementations—differing in their parameter values—of the Racchmaninoff-Oct2010 model (with pattern inheritance). The details of these models are described in Chapters 8 and 9.

It is possible to make general observations about the stylistic success ratings in Table 10.1. For instance,

- Clara Schumann’s mazurka (stimulus 7) is rated by the expert judges as more stylistically successful than any of Chopin’s mazurkas, and than any of those from EMI.
- All but one of the excerpts from System B* (stimuli 28-32) are rated by the expert judges as more stylistically successful than the amateur mazurka (stimulus 12).
- Both Chopin’s mazurkas and those from EMI appear to be rated as

Table 10.1: Mean stylistic success ratings, percentage of judges distinguishing correctly, and percentage of judges classing a stimulus as a Chopin mazurka. The stimulus number corresponds to the list given in Sec. 10.4, and the boxed numbers are for the purposes of discussion.

Stimulus	Mean style success		Distinguish correct (%)		Classed Ch. mazurka (%)	
	C'goers	Experts	C'goers	Experts	C'goers	Experts
Chopin mazurkas						
1	4.56	5.38	31.3	68.8	31.3	68.8
2	5.63	5.56	68.8	81.3	68.8	81.3
3	4.13	4.38	37.5	43.8	37.5	43.8
4	5.75	5.75	81.3	81.3	81.3	81.3
5	5.13	5.82	56.3	81.3	56.3	81.3
6	4.19	4.88	43.8	62.5	43.8	62.5
Human other (7 Clara, 8 Ch. Prel., 9 Brahms, 10 Coupn, 11 Schnbg, 12 King)						
7	5.63	6.13	0.0	0.0	81.3	93.8
8	3.94	3.25	62.5	68.8	18.8	25.0
9	3.06	2.00	81.3	87.5	12.5	6.3
10	2.56	1.56	81.3	81.3	6.3	6.3
11	1.19	1.38	68.8	81.3	0.0	0.0
12	3.06	2.69	31.3	68.8	12.5	0.0
Computer-based: EMI						
13	4.75	5.88	25.0	6.3	43.8	87.5
14	5.38	5.13	12.5	25.0	62.5	56.3
15	5.19	4.88	18.8	18.8	50.0	50.0
16	5.25	5.50	12.5	25.0	56.3	75.0
17	5.75	6.00	6.3	6.3	87.5	87.5
18	5.25	5.63	12.5	6.3	75.0	68.8
Computer-based: System A						
19	3.25	3.31	62.5	56.3	18.0	6.3
20	4.75	4.38	25.0	62.5	56.3	25.0
21	2.81	2.69	75.0	81.3	6.3	0.0
22	2.75	2.38	50.0	68.8	18.8	0.0
23	2.75	2.63	62.5	87.5	6.3	0.0
24	3.13	3.19	68.8	93.8	18.8	0.0
Computer-based: System B						
25	2.00	1.81	75.0	81.3	0.0	0.0
26	2.94	2.69	75.0	68.8	0.0	0.0
27	2.25	2.75	68.8	50.0	12.5	12.5
Computer-based: System B*						
28	3.25	2.88	43.8	81.3	25.0	0.0
29	2.94	3.06	87.5	75.0	6.3	6.3
30	2.69	2.63	56.3	68.8	12.5	6.3
31	2.75	2.89	50.0	87.5	0.0	0.0
32	2.50	2.75	81.3	93.8	12.5	0.0

more stylistically successful than those of Systems A, B, and B*.

To investigate differences in stylistic success properly, however, one ought to conduct an ANOVA using indicator variables for mazurka-generating systems. One ANOVA was conducted for concertgoer judges, and another for experts. The contrasts for the ANOVAs are given in Table 10.2, and should be interpreted as follows. If the number in the i th row, j th column of the table is positive (negative), then the j th source produces excerpts that are rated as more (less) stylistically successful than those of the i th source. The magnitude of the number indicates the significance of this difference in stylistic success. For instance, the concertgoers judged mazurkas from EMI as more stylistically successful than those of System B* (as $4.875 > 0$). And the asterisks next to 4.875 indicate that a test of the null hypothesis ‘no difference in stylistic success ratings between System B* and EMI’ versus the two-sided alternative ‘EMI rated significantly higher or lower than System B* in terms of stylistic success’ results in rejection of the null hypothesis at the .001 level.

Table 10.2 shows that, in terms of stylistic success, the Chopin mazurkas are rated significantly higher than those of Systems A, B, and B*. The mazurkas from EMI rate significantly higher for stylistic success than Systems A, B, and B* as well. The excerpts from EMI are not rated significantly differently to the Chopin mazurkas. It would have been encouraging to see the contrasts between System B* and System A, and between System B* and System B emerge as statistically significant, but they did not. Significance of the contrast between System B* and System A would constitute evidence that the introduction of pattern inheritance leads to a significant increase in stylistic success. Significance of the latter contrast between System B* and

Table 10.2: Contrasts for two ANOVAs, one conducted using concertgoer ratings of stylistic success as the response variable, the other using expert ratings. The regression formula is given in (10.1). If the number in the i th row, j th column of the table is positive (negative), then the j th source produces excerpts that are rated as more (less) stylistically successful than those of the i th source. The magnitude of the number indicates the significance of this difference in stylistic success. One, two, and three asterisks indicate significance at the .05, .01, and .001 levels respectively, testing a two-sided hypothesis using a $t(26)$ distribution. Overall significance of the regression is reported in the bottom row of each table, with s being the error standard deviation.

Concertgoers						
Source	System B*	Human other	System A	Chopin mazurka	EMI	
System B	0.429	0.844	0.844	2.500***	2.865***	
System B*	.	0.830	0.830	4.145***	4.875***	
Human other	.	.	0.000	3.477***	4.242***	
System A	.	.	.	3.477***	4.242***	
Chopin mazurka	0.765	

$$F(5, 26) = 10.12, \quad p = 1.827 \times 10^{-5}, \quad s = 0.825$$

Experts						
Source	System B*	Human other	System A	Chopin mazurka	EMI	
System B	0.421	0.417	0.677	2.875***	3.083***	
System B*	.	-0.008	0.468	4.485***	4.865***	
Human other	.	.	0.499	4.712***	5.111***	
System A	.	.	.	4.212***	4.612***	
Chopin mazurka	0.399	

$$F(5, 26) = 12.16, \quad p = 3.874 \times 10^{-6}, \quad s = 0.904$$

System B would constitute evidence that a tightening of parameters leads to a significant increase in stylistic success. There is an increase (of 0.429 for the concertgoers and 0.421 for the experts) but it is not significant at the .05 level.

Concerned about the potential for ordering effects, I investigated the proportion of times p_1 that a stimulus from EMI was misclassified (as Chopin mazurka or human other) when it followed a stimulus from Systems A, B, or B*, compared to the proportion of times p_2 that a stimulus from EMI was misclassified when it followed a Chopin mazurka. A significant difference between p_1 and p_2 would indicate that judges were lulled into a false sense of security by the more obvious computer-based stimuli. The calculated proportions are $p_1 = .885 \approx 77/87$ and $p_2 = .839 \approx 26/31$. These proportions are not significantly different at the .05 level ($z = -0.665, p = 0.51$). It would appear that ordering effects have not inflated the results in favour of stimuli from EMI.

10.5.3 Answer to evaluation question 2

If a judge is guessing answers to the question about distinguishing between the categories Chopin mazurka, human other, and computer-based, the probability of the judge distinguishing 16 or more of the 32 excerpts correctly is less than .05. So a score of 16 or more is used as a threshold to indicate that judges scored better than by chance. Of the 16 concertgoer judges, 8 scored better than by chance. Of the 16 expert judges, 15 scored better than by chance.⁴ Low percentages in columns four and five of Table 10.1 indi-

⁴Using normal approximations to the binomial distribution, the power of this test is .946 at the .05 level, assuming an alternative mean score of 19.5, which is the observed

cate that judges had trouble distinguishing an excerpt correctly as Chopin mazurka, human other, or computer-based. It can be seen that the excerpts from EMI do particularly well, with none of excerpts 13-18 being classified as computer-based by judges more than 25% of the time.

10.5.4 Answer to evaluation question 4

For the majority of judges, a judge's ratings of stylistic success and aesthetic pleasure are significantly positively correlated. This does not necessarily imply that judges failed to understand the nuanced difference between stylistic success and aesthetic pleasure. More likely, this correlation is due to there being only a couple of excerpts (Couperin Rondeau, Brahms Romance) that one would expect to receive low stylistic success ratings but that are eminently aesthetically pleasing. In fact, if the analysis is limited to stylistic success and aesthetic pleasure ratings for the Couperin Rondeau and the Brahms Romance, the correlation between stylistic success and aesthetic pleasure is not significant at the .05 level.

To investigate whether judges appear to be biased against what they perceive as computer-based stimuli (Moffat and Kelly, 2006), but what are in fact genuine Chopin mazurkas, a two-sample *t*-test was conducted. The data consist of the judges' ratings of stylistic success, restricted to stimuli 1-6 (Chopin mazurkas). The first sample contains ratings where judges misclassified stimuli as computer-based. The data associated with one participant—participant 4—were removed from this analysis, as they revealed a strong bias against all stimuli perceived as computer-based. Even with this data

mean score of the expert judges. For the concertgoers, with mean observed score ≈ 16.1 , the power of the corresponding test is .648.

removed, the result of a two-sided *t*-test suggests judges do appear to be biased against genuine Chopin mazurkas that they perceive as computer-based stimuli ($t(184) = -3.11, p = 2.14 \times 10^{-3}$).

10.5.5 Answer to evaluation question 5

For stimuli from Systems A, B, and B*, the experts made a total of 65 negative comments after rating stylistic success, which were categorised as follows: 12% for pitch range, 11% for melody, 43% for harmony, 3% for phrasing, 6% for rhythm, leaving 25% categorised as other. Among the *other* category were several comments on texture and repetition, but not enough to warrant categories in their own right. From a total of 27 positive comments, the most highly represented of the musical categories was rhythm. The concert-goer comments on stylistic success ratings exhibited similar profiles for both positive and negative categories. It would appear from these results that harmony is the general aspect of the models from Chapters 8 and 9 requiring most attention.

Are certain musical attributes of an excerpt useful predictors of its stylistic success? A model for relating judges' ratings of stylistic success to musical attributes was next determined using stepwise selection.⁵ The explanatory variables consisted of the ten variables from Chapter 6 (and defined in Appendix C) that can be applied to a whole excerpt: *pitch centre, signed pitch*

⁵Stepwise selection adds and/or eliminates variables from a model, beginning with the most significant explanatory variable, which is added if it is significant at the .05 level. Then the least significant variable in the model is eliminated, unless it is significant. The process is repeated until no further additions/eliminations can be made according to these rules. Stepwise selection is used here in preference to forward selection and backward elimination because backward elimination would result in overfitting, as there are eighteen explanatory variables for thirty datapoints.

range, unsigned pitch range, small intervals, intervallic leaps, chromatic, credential, rhythmic density, rhythmic variability, and metric syncopation. As well as these explanatory variables, eight new attributes were proposed, based on existing work: a chord labelling algorithm, called HarmAn, which was discussed on p. 23 (Pardo and Birmingham, 2002); keyscapes, which display the output of a key finding algorithm and were discussed on p. 26 (Sapp, 2005); and general metric weights, which were defined in Def. 4.7 (originally by Volk, 2008). Rather than address each of the eight new variables in turn, I will describe some differences between excerpts of music—differences that I hope one or more of the variables will capture. Full mathematical definitions of the eight new variables appear in Appendix C (p. 379 onwards).⁶

Failure to establish key. Among the expert judges' comments that were categorised as negative, 43% pertained to harmony. Harmony is multifaceted, but it would appear that passages generated by the models from Chapter 9 do not establish as strong a sense of key as the Chopin mazurkas. For instance, computer-generated stimulus 20 (shown in Fig. E.2) has a key signature of E♭ minor, inherited from the template, but there is an immediate passing modulation to G minor, followed by several more passing modulations.

Irregular harmonic rhythm. *Harmonic rhythm* refers to *where* and *how regularly* chord changes occur. So compared to establishment of key, harmonic rhythm is a different facet of harmony, as an excerpt may

⁶Where a database of Chopin mazurkas is involved in the calculation of a variable, this database consists of thirty-one mazurkas: op.6 nos.1 & 3, op.7 nos.1-3, op.17 nos.1-3, op.24 nos.2 & 3, op.30 nos.1-4, op.33 nos.1-3, op.41 no.1, op.50 nos.1 & 2, op.56 no.1, op.50 nos.1 & 2, op.63 no.1, op.67 nos.2 & 4, op.68 nos.1-4.

contain very regular chord changes without ever establishing a key. Chopin mazurkas do tend to have regular harmonic rhythm, whereas some of the passages generated by the models from Chapter 9 do not. For example, the first four bars of computer-generated stimulus 19 (shown in Fig. E.1) contain only the C major triad, and then in bar 5 there are three different chords: B \flat minor triad, F major triad, G dominant 7th.

Too complex or too simple? The weak or transient sense of key and irregular harmonic rhythm of excerpts generated by the models from Chapter 9 can sound complex (or random) compared to the corresponding facets of Chopin mazurkas. Other stimuli, however, such as the mazurka by the amateur composer (stimulus 12), stick resolutely to one key and have a pulse-like harmonic rhythm, so they sound simple compared to Chopin mazurkas. It would be elegant if a single variable could take large values for both oversimple and overcomplex excerpts, rather than having two variables—one variable taking large positive values for complexity (and large negative values for simplicity), and the other variable vice versa. That is, if Chopin's mazurkas are clustered around some point μ on a simple-complex continuum for key establishment (or harmonic rhythm), and some other excerpt (e.g., one of the stimuli) is at point x on the continuum, then the single variable should capture the absolute distance $|x - \mu|$.

The model that resulted using *stepwise selection* for the eighteen explana-

tory variables (ten from Chapter 6, plus eight new) was

$$\begin{aligned}
 \text{rating} = & 0.56 - 6.92 \cdot \text{rel_metric_weight_entropy} \\
 & - 0.05 \cdot \text{unsigned_pitch_range} - 0.88 \cdot \text{metric_syncopation} \\
 & + 4.97 \cdot \text{max_metric_weight_entropy} - 1.05 \cdot \text{keyscape_entropy} \\
 & - 0.11 \cdot \text{pitch_centre} - 1.50 \cdot \text{mean_metric_weight_entropy}, \\
 \end{aligned} \tag{10.4}$$

with test statistic $F(7, 24) = 17.34$, $p = 5.7 \times 10^{-8}$, and $s = 0.70$ as the error standard deviation. The stepwise model has a value of $r^2 = 0.83$, meaning that it explains 83% of the variation in ratings of stylistic success. This model was built in order to suggest specific variables for new constraints in future work, so it is discussed again in the next section. It is worth saying here that the stepwise model probably contains too many (four) variables to do with metre, especially as the coefficient for *maximum metric weight entropy* is positive.

10.6 Conclusions and future work

The participant study described in this chapter was intended to evaluate two models of musical style (Racchman-Oct2010 and Racchmaninof-Oct2010, see Chapter 9 for details), for the brief of generating the opening section of a mazurka in the style of Chopin. Using an adapted version of the Consensual Assessment Technique (Amabile, 1996; Pearce and Wiggins, 2007), judges listened to short excerpts of music and, among other questions, were asked to rate each excerpt in terms of stylistic success as a Chopin mazurka. In

addition to the computer-generated stimuli from my models, genuine Chopin mazurkas were among the stimuli, as well as other human-composed music. Mazurkas from another computer model called EMI (Cope, 1997) offered a further source of comparison.

The work presented in Chapters 5, 8, and 9 constitutes a thorough review, development, and evaluation of computational models of musical style. The detailed description of two models in Chapters 8 and 9—Racchman-Oct2010 and Racchmaninof-Oct2010—achieves a full level of disclosure, and I have published the source code for my models.⁷ The evaluation has produced some encouraging results. First, as shown in Table 10.1, all but one of the excerpts from System B* (stimuli 28-32) are rated by the expert judges as more stylistically successful than the amateur mazurka (stimulus 12). Second, stimulus 20 (System A) was miscategorised as a Chopin mazurka by 56% of concertgoer judges and 25% of expert judges, and stimulus 28 (System B*) was miscategorised similarly by 25% of concertgoer judges (boxed numbers in Table 10.1). Taken together, these results suggest that some aspects of musical style are being modelled effectively by Racchman-Oct2010 and Racchmaninof-Oct2010, and that at least some of the generated passages can be considered on a par with human-composed music.

That said, the results also indicate potential for future improvements. Chopin mazurkas are rated significantly higher in terms of stylistic success than those of Systems A (Racchman-Oct2010), B, and B* (both Racchmaninof-Oct2010). The mazurkas from EMI rate significantly higher for stylistic success than Systems A, B, and B* as well. The excerpts from EMI are not rated significantly differently to the Chopin mazurkas.

⁷Please see the accompanying CD (or <http://www.tomcollinsresearch.net>)

The results showed no statistically significant difference between stylistic success ratings for patterned computer-generated stimuli (from Systems B and B*) versus nonpatterned (System A). This does not mean that repeated patterns are unimportant for computational modelling of musical style, however. Some judges were sensitive to repetition: ‘It sounds like a human composer in that it is unified’ (expert judge 3 on stimulus 28; ‘First half appears to be repeated’ (concertgoer judge 16 on stimulus 12). Perhaps other aspects of style, such as harmony or melody, need to better-modelled in the first place, before judges begin to use the presence or absence of repeated patterns as a basis for rating stylistic success. There is also the argument that perception of repeated patterns requires deep engagement with a piece. Judges had an hour to rate thirty-two excerpts of music. Arguably, a listener is unlikely to gain much of an appreciation of the repeated patterns within an excerpt, when only a couple of minutes will be spent listening to and thinking about it. Another possible reason why there was no significant difference due to pattern inheritance is that System B* involves generating music over several time intervals, trying to stitch an imperceptible *seam* between forwards- and backwards processes for each interval. Each excerpt from System A had only one seam. It would be worth examining whether seams are perceived by judges as stylistic shortcomings, because if so ratings for System B* could suffer more than ratings for System A.

Judges’ comments about stimuli were used to build a model (10.4), in order to suggest specific variables for new constraints in future work. A variable that uses keyscapes (discussed in relation to Fig. 2.6) called *keyscape entropy* (defined on p. 382) emerged as a candidate for a new constraint that

monitors the establishment of key. As constraints for pitch range and mean pitch already exist in Systems A, B, and B*, the presence of the variables *unsigned pitch range* and *pitch centre* in (10.4) suggests that parameters for these constraints were too relaxed (low). Further work is required in order to investigate whether such constraints can be tightened (and new constraints added), and still have the models produce output within a couple of hours.

Do the judges' comments shed any light on listening strategies for distinguishing between human-composed and computer-generated music?⁸ It can be difficult to articulate the reasoning behind distinguishing one way or another, and perhaps this is reflected by similar comments from judges leading to different decisions: concertgoer judge 5 categorised stimulus 1 as human other, having observed that 'the intro seemed not in character'; whereas expert judge 6 categorised it correctly, observing that stimulus 1 is 'harmonically...complex but also goes where one hopes it will. [S]lightly unusual opening (solo right hand), but seems to get going after this'. There was evidence of both *instantaneous* and *holistic* listening strategies being employed to distinguish between human-composed and computer-generated music: 'all the gestures in themselves work, but the way they are put together certainly does not' (expert judge 7 on stimulus 27); 'I thought is was Chopin at first, but there is a rhythm that leads me to believe it isn't. Between bars 7-8' (expert judge 3, again on stimulus 27).

Some comments revealed misunderstanding of the mazurka style. For in-

⁸A reminder of the different stimulus categories may be helpful before embarking on this discussion: stimuli 1-6 are Chopin mazurkas, and stimuli 7-12 are in the category human other. Within the computer-based category (stimuli 13-32), 13-18 are mazurkas attributed to EMI (Cope, 1997), 19-24 are from System A (Racchman-Oct2010), 25-27 are from System B, and 28-32 are from System B* (both Systems B and B* use Racchmaninoff-Oct2010).

stance, parallel fifths are more common in Chopin mazurkas (see Fig. 8.6) than in J.S. Bach's chorale harmonisations, say. But expert judge 4 observes 'dissonant downbeats, parallel fifths—eek!' in stimulus 32. As another example, the third beat of the bar in a Chopin mazurka might not contain any new notes, as some mazurkas emphasise the second beat. Concertgoer judge 16, however, categorises stimulus 28 as human other, perhaps because 'the missing third beat in bars one and three sound[s] untypical'. Judges were sensitive to random-sounding aspects of excerpts, but vacillated over whether or not *randomness* was an indicator of the computer-based category. Both in relation to stimulus 19, for instance, concertgoer judge 14 observed 'it sounds too random to be computer generated', whereas for concertgoer judge 16, the 'rhythm [was] mostly OK but the random melodic line seems computerish'. Finally, although this may not have had a bearing on the distinguishing question, expert judges appeared to be more receptive than concertgoer judges to the atonal excerpt by Schoenberg (stimulus 11): 'love it—it sounds almost 12-tone' (expert judge 4); '[c]ould well be by a modern composer, not my cup of tea, a computer program would do better than this' (concertgoer judge 8).

11

Conclusions and future work

11.1 Conclusions

This thesis has considered algorithms for the discovery of patterns in music, as well as the application of these algorithms in the context of automated stylistic composition. My contribution to methods for pattern discovery has been twofold. First, I have investigated which musical attributes of a discovered pattern are useful predictors of the pattern's perceived importance, and, using variable selection, found that a weighted combination of three attributes (*compactness*, *expected occurrences*, and *compression ratio*) explains data collected from students of music analysis. Second, I have defined the Structure Induction Algorithm with Compactness Trawling (SIACT), which improves upon the recall and precision values of other pattern discovery algorithms, evaluated using a benchmark of independently analysed Baroque keyboard works. SIACT is the newest addition to the SIA family of algorithms (Meredith et al., 2003; Forth and Wiggins, 2009), and is an attempt to solve the problem of isolated membership, which, as demonstrated, affects the rest of the family. Combining and applying these two contributions, I ran SIACT on the opening section of the Mazurka in B major op.56 no.1 by Chopin, then used the formula for rating pattern importance to present the

ten top-rated patterns (Appendix D). The output patterns seem promising in this instance, and, in later chapters, the discovery-rating process is applied to automated stylistic composition. There are arguments (which point to further work and evaluation) for filtering out certain types of patterns and for being able to discover extra inexact occurrences of the top-rated patterns (e.g., to avoid the user browsing through near-duplicates of discovered patterns). It would not surprise me if within a few years, an algorithm with SIACT at its core were used to prepare an analysis essay, just as Huron (2001b) used a pattern matching tool to assist the analysis of Brahms' op.51 no.1. Computer-assisted pattern discovery could be a defining feature of scholarship for the next generation of music analysts.

My contribution to computational modelling of musical style has been to develop and evaluate two algorithms: one called Racchman-Oct2010; the other called Racchmaninof-Oct2010 (RAndom Constrained CHain of MArkovian Nodes with INheritance Of Form). The evaluation focused on generating the opening section of a mazurka in the style of Chopin. Analysis of the judges' responses suggests that some aspects of musical style are being modelled effectively by Racchman-Oct2010 and Racchmaninof-Oct2010, and that sometimes passages generated by these models were difficult to distinguish from original Chopin mazurkas. Regarding stylistic success ratings, however, there is certainly potential to improve upon this set of results in the future. The evaluation aside, my review and development of employing random generation Markov chains (RGMC) to model musical style achieve a full level of disclosure, and I have published the source code for my models.¹ I hope my description of and source code for the models Racchman-Oct2010

¹Please see the accompanying CD (or <http://www.tomcollinsresearch.net>)

and Racchmaninof-Oct2010 will act as a catalyst for future work and contributions from other researchers. This may help cure what Pearce et al. (2002) call the malaise affecting research on computational models of musical style.

How has this thesis shed light on musical style? Arguably, pattern inheritance (in which the temporal and registral positions of discovered repeated patterns from an existing template piece are used to guide the generation of a new passage of music) is one of the most interesting aspects of the current work. The difference between the models Racchman-Oct2010 and Racchmaninof-Oct2010 is that the latter includes pattern inheritance, and so I have demonstrated that it is possible to define a music-generating algorithm where, say, ‘what happens in measure 5 may directly influence what happens in measure 55, without necessarily affecting any of the intervening material’ (Cope, 2005, p. 98). This is an important step towards more sophisticated computational models of stylistic composition, although the prize remains unclaimed for demonstrating experimentally that pattern inheritance alone can lead to improved ratings of stylistic success.

11.1.1 Revisiting hypotheses from the introduction

This section restates five hypotheses from the introduction (Chapter 1), and discusses, from a technical point of view, the extent to which each hypothesis has been substantiated.

Hypothesis 1. Music analysts’ ratings of discovered patterns as relatively noticeable and/or important can be modelled by a linear combination of quantifiable pattern attributes, such as the number of notes a pattern contains, the number of times it occurs, its *compactness*

etc. (cf. Sec. 6.1.3). Furthermore, this combination of attributes will offer a better explanation of the analysts' ratings than any previously proposed formula does individually.

Hypothesis 2. The recall and precision values (4.11) of certain algorithms for pattern discovery in music (SIA, SIATEC, and COSIATEC, described in Sec. 4.2 and by Meredith et al., 2003) are adversely affected by the problem of isolated membership, as exemplified in Sec. 7.1 (p. 166).

Hypothesis 3. The problem of isolated membership can be addressed by a method that I call compactness trawling. By implementing this method as a compactness trawler (CT) and appending it to the algorithm SIA, the result will be an algorithm (SIACT, cf. Def. 7.2) with higher recall and precision values than the existing members of the SIA family, as evaluated on a particular benchmark.

Hypothesis 4. A random generation Markov chain (RGMC) with appropriate state space and constraints is capable of generating passages of music that are judged as successful, relative to an intended style, within the framework of the Consensual Assessment Technique (Amaibile, 1996; Pearce and Wiggins, 2007). Two models described in Chapter 9 (Racchman-Oct2010 and Racchmaninof-Oct2010) enable specific instances of this hypothesis to be tested for the Chopin mazurka brief (p. 94).

Hypothesis 5. The difference between the models Racchman-Oct2010 and Racchmaninof-Oct2010 is that the latter includes pattern inheritance.

I hypothesise that altering an RGMC to include pattern inheritance from a designated template piece will lead to higher judgements of stylistic success, again within the framework of the Consensual Assessment Technique.

Evidence in support of hypothesis 1 was presented in Chapter 6. Specifically, the so-called forward model (6.2) emerged as the strongest predictor for music analysts' ratings of discovered patterns, accounting for just over 70% of the variability in the responses. Table 6.1 (p. 141) shows that, individually, the best predictor of the analysts' ratings was the *compactness* variable, which accounted for 63% of the variability in the responses ($r^2 = .63$). Thus, it is clear that the combination of attributes present in the forward model (6.2) offers a better explanation of the analysts' ratings than any of the proposed attributes do individually.

The main piece of evidence in support of hypothesis 2 is a music example, Fig. 4.11, which is discussed once in Chapter 4 (p. 72), and again in Chapter 7 (p. 165). The first discussion suggests that for a small and conveniently chosen excerpt of music, the maximal translatable pattern (MTP, Meredith et al., 2002) named P in (4.2) corresponds exactly to a perceptually salient pattern. In the second discussion, the excerpt (and dataset representation) is enlarged by one bar, and the MTP, renamed P^+ in (7.2), gains some temporally isolated members. As a result, the salient pattern is lost inside the MTP. A single example does not constitute strong evidence, but intuitively, the larger the dataset, the more likely it is that this problem of isolated membership will occur. As each existing algorithm in the SIA family uses MTPs, each of their recall and precision values are adversely affected.

Evidence in support of hypothesis 3 (and in further support of hypothesis 2) was presented in Sec. 7.2. A music analyst analysed the Sonata in C major L1 and the Sonata in C minor L10 by D. Scarlatti, and the Prelude in C \sharp minor BWV849 and the Prelude in E major BWV854 by J.S. Bach. A benchmark of translational patterns was formed for each piece, according to the intra-opus translational pattern discovery task (cf. Def. 4.2). Three algorithms—SIA (Meredith et al., 2002), COSIATEC (Meredith et al., 2003) and my own, SIACT—were run on datasets that represented L1, L10, BWV849, and BWV854. Often COSIATEC did not discover any target patterns, so for these pieces it has zero recall and precision, as shown in Table 7.1. Of the two remaining contenders, SIA and SIACT, SIACT (cf. Def. 7.2) out-performs SIA in terms of both recall and precision. Having examined cases in which SIA and COSIATEC fail to discover targets, I ascribe the relative success of SIACT to its being intended to solve the problem of isolated membership.

With regards hypothesis 4, Chapter 10 describes an experiment in which sixteen concert-going judges and sixteen expert judges listened to excerpts of music, and were told that some of the excerpts were from Chopin mazurkas, some were from the work of human composers but were not Chopin mazurkas, and some were based on computer programs that learn from Chopin’s mazurkas. The last category included fourteen excerpts that were generated by the models described in Chapter 9. System A is a shorthand for excerpts generated by Racchman-Oct2010, and System B for excerpts generated by Racchmaninof-Oct2010. System B* is also a shorthand for excerpts generated by Racchmaninof-Oct2010, but with a set of parameters directly comparable

to those of System A. The evaluation produced some encouraging results: all but one of the excerpts from System B* were rated by the expert judges as more stylistically successful than the mazurka by an amateur composer; an excerpt from System A was miscategorised as a Chopin mazurka by 56% of concertgoer judges and 25% of expert judges, and an excerpt from System B* was miscategorised similarly by 25% of concertgoer judges.

The results also indicate potential for future improvements. Table 10.2 shows that, in terms of stylistic success, the Chopin mazurkas are rated significantly higher than those of Systems A, B, and B*. There is little evidence, in this specific instance, of a random generation Markov chain (RGMC) being capable of generating passages of music that are judged as successful, relative to the intended style. This does not contradict the hypothesis in general, however, that some such RGMC exists. It could be that some aspect of the state space and/or the constraints is inappropriate. In the analysis (Sec. 10.5.5), I was able to determine how several quantifiable attributes of an excerpt detract from its stylistic success rating, thus highlighting specific areas for future improvements.

In terms of hypothesis 5, the contrast of interest in Table 10.2 is between System B* and System A. Significance of this contrast in favour of System B* would constitute evidence that the introduction of pattern inheritance leads to a significant increase in stylistic success. This significance was not observed, however. There is little evidence, in this specific instance, that altering an RGMC to include pattern inheritance leads to higher judgements of stylistic success. Again, this does not contradict the general hypothesis that it is possible to alter some RGMC to include pattern inheritance, and

observe such an effect. Possible reasons for the nonsignificant pattern inheritance result were discussed in Sec. 10.6. In short, one reason is to do with the time judges were given to complete the task. A second reason is that an excerpt from System B* contains more so-called seams between forwards- and backwards-generating processes, compared with an excerpt from System A. It would be worth examining whether seams are perceived by judges as stylistic shortcomings.

11.1.2 The precision and runtime of SIA

Having improved upon the recall of existing structural induction algorithms, it seems appropriate to conclude with some remarks on precision and runtime. The computational complexity of SIA is $O(kn^2 \log n)$, where k is the dimension of the dataset D on which SIA is run, and n is the number of datapoints in D .² Meredith et al. (2002) state that for a dataset representation of a piece containing approximately 3500 points, SIA takes approximately 2 minutes to run. This runtime seems acceptable, given the dataset representation of one of the longest Chopin mazurkas (op.56 no.1) contains approximately 2200 points. If, however, one runs SIA on many different projections of the same dataset, on the dataset for an entire multi-movement work, or on the dataset for a piece with thick textures (e.g., a symphony), then the runtime of SIA will become an issue. Is there anything that can be done? SIA calculates the upper triangle of the similarity array \mathbf{A} in (4.9), and performs a sort. Mention was made of limiting the calculation to only the first r superdiagonals of \mathbf{A} (cf. p. 168 and Def. A.3). The issue is: if \mathbf{v} is the generating vector of an

²Hashing can reduce the computational complexity to $O(kn^2)$, but relies on prior assumptions about the dataset (Meredith, 2006b).

MTP that corresponds to a noticeable and/or important pattern P , so that $P = \text{MTP}(\mathbf{v}, D)$, and \mathbf{v} lies beyond the first r superdiagonals, then P will not be discovered via $\text{MTP}(\mathbf{v}, D)$. For instance, the generating vector used as an example in Chapter 7, $\mathbf{v} = (3, 3)$, appears in the eighth superdiagonal.

Assumption 11.1. Assumption of compactness. It is possible to exploit the findings of Chapters 6 and 7; that for a noticeable and/or important pattern $P = \{\mathbf{d}_{i_1}, \mathbf{d}_{i_2}, \dots, \mathbf{d}_{i_l}\}$, the datapoints $\mathbf{d}_{i_1}, \mathbf{d}_{i_2}, \dots, \mathbf{d}_{i_l}$ tend to be relatively compact in the dataset D . Therefore, it can be assumed that one or more of the difference vectors $\mathbf{d}_{i_2} - \mathbf{d}_{i_1}, \mathbf{d}_{i_3} - \mathbf{d}_{i_2}, \dots, \mathbf{d}_{i_l} - \mathbf{d}_{i_{l-1}}$ lies on or within the r th superdiagonal of the similarity array \mathbf{A} , where r is small. ■

Perhaps the above difference vectors, which can be calculated much quicker than the whole upper triangle of \mathbf{A} , can be used to discover P , rather than relying on the generating vector \mathbf{v} . To develop this idea into an algorithm requires the idea of conjugacy.

Definition 11.2. Conjugacy array, conjugate pattern, and conjugate TEC. Let P be a pattern in a dataset D , with translational equivalence class $\text{TEC}(P, D) = \{P_1, P_2, \dots, P_m\}$. For an occurrence $P_i \in \text{TEC}(P, D)$, let $P_i = \{\mathbf{p}_{i,1}, \mathbf{p}_{i,2}, \dots, \mathbf{p}_{i,l}\}$. The *conjugacy array* $\mathbf{J}_{P,D}$ for the pattern P in the dataset D is defined by

$$\mathbf{J}_{P,D} = \begin{pmatrix} \mathbf{p}_{1,1} & \mathbf{p}_{1,2} & \cdots & \mathbf{p}_{1,l} \\ \mathbf{p}_{2,1} & \mathbf{p}_{2,2} & \cdots & \mathbf{p}_{2,l} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{p}_{m,1} & \mathbf{p}_{m,2} & \cdots & \mathbf{p}_{m,l} \end{pmatrix}. \quad (11.1)$$

Each row of $\mathbf{J}_{P,D}$ constitutes an element of $\text{TEC}(P, D)$, but what about the

columns of $\mathbf{J}_{P,D}$? Letting Q be the set of datapoints from the first column, $Q = \{\mathbf{p}_{1,1}, \mathbf{p}_{2,1}, \dots, \mathbf{p}_{m,1}\}$, each column of $\mathbf{J}_{P,D}$ constitutes an element of $\text{TEC}(Q, D)$. It is said that P and Q are *conjugate patterns*, and that $\text{TEC}(P, D)$ and $\text{TEC}(Q, D)$ are *conjugate TECs*. ■

Example 11.3. An excerpt by D. Scarlatti is represented as a dataset in Fig. 11.1A. The translational equivalence class $\{P_1, P_2, P_3\}$ is indicated by dotted lines. Two members, Q_1 and Q_3 , of the conjugate translational equivalence class are indicated by solid lines.

The excerpt represented in Fig. 11.1A was discussed first in relation to Fig. 4.11, and then again in Chapter 7. To summarise, it was labelled D^+ , and P_1 can be discovered by running SIA on D^+ , followed by applying a compactness trawler to one of the output patterns, $P^+ = \text{MTP}(\mathbf{v}, D^+)$, where $\mathbf{v} = (3, 3)$. As mentioned above, \mathbf{v} appears in the eighth superdiagonal of the similarity array.

Is it possible to discover \mathbf{v} (and hence P^+ and P_1) without calculating the whole upper triangle of the similarity array, which is what SIA does? Figure 11.1B shows three vectors (indicated by arrows) that lie on the first superdiagonal of the similarity array, in other words difference vectors for adjacent members of the lexicographically-ordered dataset. When all vectors from the first superdiagonal are sorted, these three vectors appear next to one another, as they are equal. Using \mathbf{u} to label these three vectors, pattern Q_3 can be discovered by retaining the indices of datapoints that give rise to \mathbf{u} . Running SIA on Q_3 , which is relatively quick because Q_3 has far fewer datapoints than D^+ , the vector $\mathbf{v} = (3, 3)$ will be among the difference vectors indicated by arrows in Fig. 11.1C (with \mathbf{v} being the solid arrow). Calculating P^+ , the

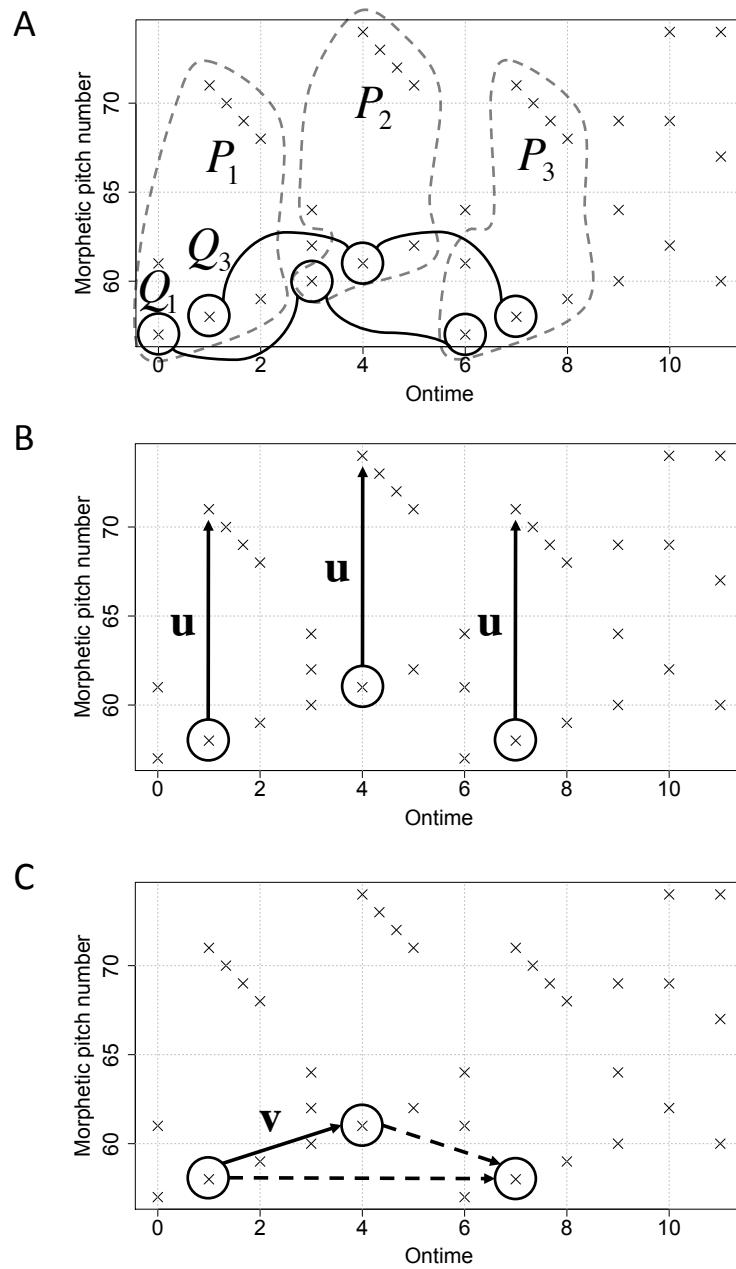


Figure 11.1: (A) A dataset representation for bars 13-16 of the Sonata in C major L3 by D. Scarlatti. The translational equivalence class $\{P_1, P_2, P_3\}$ is indicated by dotted lines. Two members, Q_1 and Q_3 , of the conjugate translational equivalence class are indicated by solid lines; (B) Three difference vectors for adjacent members of the lexicographically-ordered dataset, all labelled u , as they are equal; (C) Among the difference vectors for members of Q_3 is $v = (3, 3)$, indicated by the solid arrow.

MTP of \mathbf{v} , can be thought of as *switching between conjugate representations*, as overall pattern P_1 is discovered via discovery of Q_3 . ■

Motivated by the previous example, I will now define an algorithm SIAR, standing for Structure Induction Algorithm for r superdiagonals. As the name suggests, it discovers patterns based on calculating only the first r superdiagonals of the similarity array \mathbf{A} from (4.9). SIAR combines the assumption of compactness (Assumption 11.1) with the concept of conjugacy (Def. 11.2).

Definition 11.4. Structure induction algorithm for r superdiagonals (SIAR). Let $D = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n\}$ be a dataset in lexicographic order, and \mathbf{A} be the similarity array for D , as defined in (4.9).

1. Calculate *only* the first r superdiagonals of the similarity array \mathbf{A} .
2. List the difference vectors from step 1 in lexicographic order, retaining the index of the datapoint that gave rise to each difference vector. That is, if $\mathbf{u} = \mathbf{d}_j - \mathbf{d}_i$ is a difference vector, then i should be retained alongside \mathbf{u} in the sorted list of difference vectors. I will label a set of datapoints giving rise to the same difference vector \mathbf{u} as

$$E_{\mathbf{u}} = \{\mathbf{d}_{i_1}, \mathbf{d}_{i_2}, \dots, \mathbf{d}_{i_m}\}. \quad (11.2)$$

3. SIA is applied to each dataset $E_{\mathbf{u}}$ if $m > 1$ (and is relatively quick because $m \ll n$). The difference vectors that result from applying SIA to each of the lists $E_{\mathbf{u}}$ are stored in one ordered list, labelled L .
4. Now I switch to the conjugate representation. For each distinct element

- \mathbf{w} of the list L from step 3, calculate the maximal translatable pattern, $\text{MTP}(\mathbf{w}, D)$.
5. The vector-MTP pairs $(\mathbf{w}, \text{MTP}(\mathbf{w}, D))$, where $\mathbf{w} \in L$, are the output of SIAR. ■

Further research is required to investigate whether the recall and precision of SIAR are consistently higher than the recall and precision of SIA, and whether the runtime of SIAR is consistently less than the runtime of SIA, but the results of initial investigations are encouraging.

11.2 Future work

Some suggestions for future work have already been made in Secs. 6.3.2, 7.3.2, and 10.6. Common concerns amongst these suggestions are:

- The need for evaluation on different benchmarks (Secs. 6.3.2 and 7.3.2) and/or databases (e.g., Chordia, Sastry, Malikarjuna, and Albin, 2010).
- The need for evaluation with different parameter settings, state spaces, and constraints (Sec. 10.6). In particular, it would be worth comparing the size of different state spaces and sparseness of different transition matrices.
- The need to relate computational research topics and research findings back to concepts in music analysis and composition.

In the last case, for instance, a music analyst might criticise the proto-analytical class of repetition types (cf. Def. 4.1) for being oversimple; for not including patterns that involve *thematic metamorphosis*, say, which is

the ‘process of modifying a theme so that in a new context it is different but yet manifestly made of the same elements’ (Macdonald, 2001, p. 694). There is no systematic mechanism within the SIA family for discovering instances of thematic metamorphosis, so this concept in music analysis could act as a springboard for a new computational research topic.

11.2.1 The adaptation of SIACT for audio summarisation

The remainder of this chapter will address the adaptation of SIACT for audio summarisation. Audio representations are discussed in Chapter 2 and Appendix B, and it is interesting to consider the challenges posed when applying SIACT to transcribed audio.

Taking the audio signal for a piece of music as input, the output of an audio summarisation algorithm is a time interval $[a, b]$, suggesting the portion of the audio signal that provides the most representative $(b - a)$ seconds of the piece. Example applications of audio summarisation include browsing music databases, where users require representative portions of audio, and chart countdowns, such as:

...Up eleven places in the chart this week, at number three, its Nicole Sherzinger with ‘Right there’. [Plays 5 seconds of song.] At two, its Pitbull, ‘Give me everything’. [Plays 5 seconds of song.] Which means this week’s number one is brand new: it’s Example and ‘Changed the way you kiss me’. [Plays whole song.]

At the end of Chapter 2 (p. 26) I mentioned the merging of audio and symbolic representations that can be achieved using an automated transcrip-

tion algorithm, and gave Melodyne as an example program (cf. Fig. 2.7). An adapted version of SIACT could be applied to the output of an automatic transcription algorithm (pairs of ontimes and MIDI note numbers, most likely), the discovered patterns could be rated for musical importance, and the top-rated pattern used to output a time interval $[a, b]$ that constitutes a representative summary of the input audio signal. That is, some version of SIACT might be a candidate component for an audio summarisation algorithm. It is debatable whether any member of the SIA family (including SIACT) is suited to the task of audio summarisation: these algorithms work for representations of polyphonic pieces and are capable of discovering nested and overlapping patterns; perhaps a simpler algorithm for segmentation of melodies would be just as effective for audio summarisation, and faster.

At present, when SIACT is applied to the output of an automatic transcription algorithm, among the output are many instances of patterns A , B , B' , and C , where B is a translation of A , C is a translation of B' , and the patterns B and B' are almost but not quite equal. As an example, I return to the automatic transcription of the portion of ‘To the end’ by Blur (1994), shown in Fig. 2.7, now represented as a dataset in Fig. 11.2. It is evident from Fig. 11.2 that B is a translation of A , C is a translation of B' , and the patterns B and B' are not quite equal. Consequently most, if not all, discovered patterns have *two* occurrences, which reduces the potency of the rating formula (6.4). A solution to the above problem would be to coerce the patterns A, B, B' , and C into a *fuzzy* (approximate) version of a TEC, $\{A, B, B'C\}$. Counting the number of approximate occurrences of A in the dataset, the potency of the rating formula (6.4) would be reestablished.

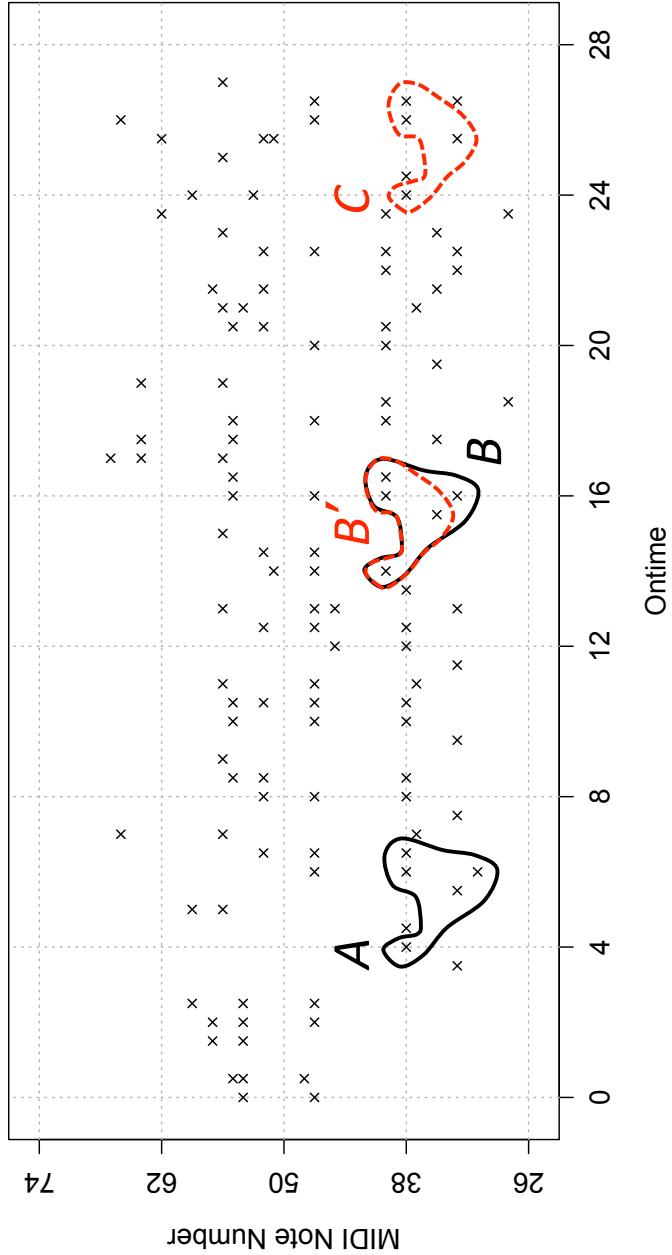


Figure 11.2: A dataset representation for bars 1-8 of 'To the end' by Blur (1994), from the Melodyne automatic transcription algorithm. Four patterns, A , B , B' , and C , are annotated. Pattern B is a translation of A , pattern C is a translation of B' , and the patterns B and B' are almost but not quite equal.

For a member of the SIA family to be applied in audio summarisation, the general version of the problem outlined above could be broken down as follows:

Problem 1. For a dataset D of dimension k and two patterns $P, Q \in D$, determine the *extent* $\varepsilon \in [0, 1]$ to which Q is a translation of P .

Problem 2. For all discovered patterns and their translations P_1, P_2, \dots, P_M in a dataset D , let the *partition*

$$U_1 = \{P_1, P_2, \dots, P_{i_1}\}, \quad (11.3)$$

$$U_2 = \{P_{i_1+1}, P_{i_1+2}, \dots, P_{i_2}\}, \quad (11.4)$$

$$\vdots$$

$$U_N = \{P_{i_{N-1}+1}, P_{i_{N-1}+2}, \dots, P_{i_N=M}\} \quad (11.5)$$

be such that for two arbitrary patterns $P, Q \in U_j$, where $1 \leq j \leq N$, and for an arbitrary pattern $R \in \{P_1, P_2, \dots, P_M\} \setminus U_j$, the *extent* to which Q is a translation of P is greater than the *extent* to which R is a translation of P . Or, if this rule has to be broken, the partition in (11.3)-(11.5) is the partition that breaks the rule a minimal number of times.

Problem 1 might be addressed by finding a vector $\mathbf{v} \in \mathbb{R}^k$ such that according to some distance function δ , the distance $\delta(\tau(P, \mathbf{v}), Q)$ is a local minimum, and then setting the *extent* as $\varepsilon = 1 - \delta$. Similar problems have been addressed by Romming and Selfridge-Field (2007), Clifford et al. (2006), and Ukkonen et al. (2003). Romming and Selfridge-Field's (2007) approach

involves calculating the similarity array

$$\text{sim}(P, Q) = \begin{pmatrix} \mathbf{q}_1 - \mathbf{p}_1 & \mathbf{q}_2 - \mathbf{p}_1 & \cdots & \mathbf{q}_m - \mathbf{p}_1 \\ \mathbf{q}_1 - \mathbf{p}_2 & \mathbf{q}_2 - \mathbf{p}_2 & \cdots & \mathbf{q}_m - \mathbf{p}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{q}_1 - \mathbf{p}_l & \mathbf{q}_2 - \mathbf{p}_l & \cdots & \mathbf{q}_m - \mathbf{p}_l \end{pmatrix}, \quad (11.6)$$

which is a generalisation of the similarity array from (4.9).

A solution to problem 1 can act as a springboard for tackling problem 2, with the *extent* to which Q is a translation of P being used to determine whether Q is included in a partition U_j that already contains P . Returning to the example shown in Fig. 11.2, the extent to which B' is a translation of A ought to be close to one (maximal), making B' and C strong candidates for inclusion in a partition that already contains patterns A and B .

The work presented over the course of this thesis has demonstrated improvements to pattern discovery algorithms (in terms of recall and ability to rate output), as well as an application in automated stylistic composition, but this work is only a beginning. The topics of precision, runtime, and summarisation merit further investigation, and point to high-level considerations, such as whether the SIA family can be applied to very large databases of music (audio or symbolic), and perhaps even to databases beyond the discipline of music.

Appendices

A

Mathematical definitions

This appendix contains all of the definitions necessary for understanding the mathematics in the thesis, with the exception of some methods for statistical analysis. Cameron (1998) is a suitable companion for most of Defs. A.1-A.28. Ross (2006), the main source for Defs. A.29-A.45, contains many supplementary examples and problems. Readers looking for more details on methods for statistical analysis may find Lunn's (2007a; 2007b) lecture notes a good starting point, with Daly et al. (1995) and Davison (2003) for further reading.

Definition A.1. Vector. A vector is a collection of numbers, separated by commas and enclosed by parentheses ‘(’ and ‘)’. A vector may contain the same number more than once. It is standard to use lowercase bold letters to denote vectors. ■

Example A.2. Here are some examples of vectors:

$$\mathbf{a} = (1, 2, 3), \quad \mathbf{b} = (2, 1, 3), \quad \mathbf{c} = (c_1, c_2, \dots, c_n). \quad (\text{A.1})$$

The vector \mathbf{c} demonstrates the general notation for a vector, that is one to which numerical values have not been assigned. The ellipsis ‘...’ is useful for saving time and space. The vectors \mathbf{a} and \mathbf{b} from (A.1) are *not* considered to

be equal: they contain the same numbers, but in different orders. In general, two vectors $\mathbf{x} = (x_1, x_2, \dots, x_m)$ and $\mathbf{y} = (y_1, y_2, \dots, y_n)$ are said to be equal if $m = n$ and $x_i = y_i$, where $i = 1, 2, \dots, m$. ■

Definition A.3. Matrix, matrix operations, and array. Whereas a vector is a list of numbers, a *matrix* is a table of numbers, consisting of m rows and n columns. The entry in the i th row, j th column of a matrix \mathbf{A} is denoted $(\mathbf{A})_{i,j}$ or $a_{i,j}$. So

$$\mathbf{A} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix}. \quad (\text{A.2})$$

The sum of two $m \times n$ matrices \mathbf{A} and \mathbf{B} is defined by $(\mathbf{A} + \mathbf{B})_{i,j} = (\mathbf{A})_{i,j} + (\mathbf{B})_{i,j}$. Similarly for subtraction. For a constant $\lambda \in \mathbb{R}$, $\lambda\mathbf{A}$ is defined by $(\lambda\mathbf{A})_{i,j} = \lambda(\mathbf{A})_{i,j}$. The *diagonal* of an $m \times n$ matrix \mathbf{A} is a list consisting of the elements $a_{i,i}$, where $1 \leq i \leq \min\{m, n\}$. The *upper triangle* of \mathbf{A} is a list consisting of the elements $a_{i,j}$, where $1 \leq i \leq \min\{m, n\}$ and $i < j$. The r th *superdiagonal* of a \mathbf{A} is a list consisting of the elements $a_{i,i+r}$, where $1 \leq i \leq n - r$.

The product of \mathbf{A} , an $m \times n$ matrix, and \mathbf{B} , an $n \times p$ matrix, is written \mathbf{AB} , an $m \times p$ matrix, and its i th row, j th column is given by

$$(\mathbf{AB})_{i,j} = \sum_{k=1}^n a_{i,k} b_{k,j}. \quad (\text{A.3})$$

Other matrix operations include *transposition* and *inversion*. For an $m \times n$

matrix \mathbf{A} , the *transpose* is written \mathbf{A}^T , and its i th row, j th column is given by

$$(\mathbf{A}^T)_{i,j} = (\mathbf{A})_{j,i}. \quad (\text{A.4})$$

The identity matrix \mathbf{I} is an $m \times m$ matrix such that $(\mathbf{I})_{i,j} = 1$ for $i = j$, and $(\mathbf{I})_{i,j} = 0$ otherwise. For \mathbf{A} , an $m \times n$ matrix, under certain conditions (not specified here) there exists \mathbf{B} , an $n \times m$ matrix, such that $\mathbf{AB} = \mathbf{I}$. In which case, we say that \mathbf{B} is the *matrix inverse* of \mathbf{A} , and use the notation $\mathbf{A}^{-1} = \mathbf{B}$.

A one-dimensional *array* is a vector; a two-dimensional *array* is a matrix. It is possible to extend the concept of an array to d dimensions, although such arrays are not easily displayed on paper, and the index notation becomes unwieldy. Let us consider the case $d = 3$. We can define $\mathbf{A}^{(k)}$ to be an $m \times n$ matrix with i th row, j th column denoted $a_{i,j,k}$, and imagine stacking p matrices $\mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \dots, \mathbf{A}^{(p)}$ *back to back* to form an $m \times n \times p$ *block* of numbers. If we denote the stacked matrices by \mathbf{A} , then \mathbf{A} is a three-dimensional array.

In Chapters 4, 7, and 11 of the thesis, I use the notation

$$\mathbf{A} = \begin{pmatrix} \mathbf{a}_{1,1} & \mathbf{a}_{1,2} & \cdots & \mathbf{a}_{1,n} \\ \mathbf{a}_{2,1} & \mathbf{a}_{2,2} & \cdots & \mathbf{a}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{a}_{m,1} & \mathbf{a}_{m,2} & \cdots & \mathbf{a}_{m,n} \end{pmatrix} \quad (\text{A.5})$$

for a three-dimensional array. That is, the element $a_{i,j,k}$ of the array \mathbf{A} can be thought of as the k th element of the vector $\mathbf{a}_{i,j}$. ■

Definition A.4. String. A string is a collection of alphabetic characters

enclosed by quotation marks ‘ ’ and ‘ ’. For musical purposes, other admissible characters in a string are the accidental symbols ‘ ♭ ’, ‘ ♯ ’, ‘ ♫ ’, ‘ ♪ ’, and ‘ ♩ ’, as well as the space symbol ‘ ’. Similar to vectors, a string may contain the same character more than once and it is standard to use lowercase bold letters to denote strings. ■

Example A.5. Here are some examples of strings:

$$\mathbf{s} = \text{‘Piano’}, \quad \mathbf{t} = \text{‘Violin I’}, \quad \mathbf{u} = \text{‘ATGCAACT’}, \quad \mathbf{v} = \text{‘G♯’}. \quad (\text{A.6})$$

The comments in Example A.2 about general notation, the use of ellipses, and equality apply also to strings. ■

Definition A.6. Concatenation. For two strings $\mathbf{s} = \text{‘}s_1s_2\cdots s_m\text{’}$ and $\mathbf{t} = \text{‘}t_1t_2\cdots t_n\text{’}$, the notation $\text{conc}(\mathbf{s}, \mathbf{t})$ is used to mean the concatenation of the two strings, that is $\text{conc}(\mathbf{s}, \mathbf{t}) = \text{‘}s_1s_2\cdots s_mt_1t_2\cdots t_n\text{’}$. ■

Definition A.7. List and set. A list is a collection of elements. Admissible elements of a list are numbers, vectors, strings, sets (see below), and lists themselves. Like vectors, the elements of a list are separated by commas and enclosed by parentheses ‘(’ and ‘)’. For a list, the order of elements matters as far as equality is concerned. A list may contain the same element more than once. It is standard to use uppercase italic letters to denote lists, and lowercase italic letters to denote their elements.

A set is a collection of elements. Admissible elements of a set are numbers, vectors, strings, lists, and sets themselves. The elements of a set are separated by commas and enclosed by curly brackets ‘{’ and ‘}’. Unlike vectors, strings, and lists, a set is unordered as far as equality is concerned, and must not

contain repeated elements. As with lists, it is standard to use uppercase italic letters to denote sets, and lowercase italic letters to denote their elements. The notation $a \in A$ is used to mean a is an element of the set A . A set A is said to be a subset of a set B if for each $a \in A$, $a \in B$. Two sets A and B are said to be equal if A is a subset of B , and B is a subset of A . The notation $A \subset B$ is used to mean that A is a subset of B but not equal to it, and $A \subseteq B$ to mean A is a subset of B or equal to it. ■

Example A.8. Here is an example of a list:

$$L = (3, 4, \mathbf{a}, 5, 3, (2, \mathbf{b}), \text{'Viola'}) , \quad (\text{A.7})$$

and here are several examples of sets:

$$A = \{2, 1, 3\}, \quad B = \{4, 3, 2\}, \quad C = \{1, 3, 2\}, \quad D = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n\}. \quad (\text{A.8})$$

So $A = C$. Eventually I will run out of letters to represent numbers and sets, in which case the Greek alphabet may also be employed, as well as some kind of indexing system, as with D in (A.8). Unless stated otherwise, definitions are refreshed with each new numbered equation. That is, \mathbf{a} and \mathbf{b} from (A.7) do not bear any relation to \mathbf{a} and \mathbf{b} from (A.1). In fact, each could be a vector or a string. ■

Definition A.9. Union, intersection, set difference, and Cartesian product. The union of two sets A and B , written $A \cup B$, is the set of all elements x such that $x \in A$ or $x \in B$. The previous sentence can be

expressed as set notation:

$$A \cup B = \{x : x \in A \text{ or } x \in B\}. \quad (\text{A.9})$$

The ‘or’ is inclusive, meaning it is acceptable for x to be in both A and B .

The intersection of two sets A and B , written $A \cap B$, is the set of all elements x such that $x \in A$ and $x \in B$. That is,

$$A \cap B = \{x : x \in A \text{ and } x \in B\}. \quad (\text{A.10})$$

The set difference of two sets A and B , written $A \setminus B$, is the set of all elements x such that $x \in A$ and $x \notin B$, where ‘ \notin ’ means ‘not in’. That is,

$$A \setminus B = \{x : x \in A \text{ and } x \notin B\}. \quad (\text{A.11})$$

The Cartesian product of two sets A and B , written $A \times B$, is the set of all lists (a, b) such that $a \in A$ and $b \in B$. That is,

$$A \times B = \{(a, b) : a \in A, b \in B\}. \quad (\text{A.12})$$

Each of these definitions (union, intersection, and Cartesian product) extend naturally to n sets A_1, A_2, \dots, A_n . For instance,

$$A_1 \times A_2 \times \cdots \times A_n = \{(a_1, a_2, \dots, a_n) : a_1 \in A_1, a_2 \in A_2, \dots, a_n \in A_n\}. \quad (\text{A.13})$$

Sometimes, Cartesian products over the same set are abbreviated. For instance, $A \times A \times A = A^3$. ■

Example A.10. Taking the definitions of A and B from (A.8),

$$A \cup B = \{1, 2, 3, 4\}, \quad A \cap B = \{2, 3\}, \quad A \setminus B = \{1\}. \quad (\text{A.14})$$

Again taking the definition of A from (A.8), and letting $B = \{\text{'Fl}', \text{'Hn'}\}$,

$$A \times B = \{(1, \text{'Fl'}), (1, \text{'Hn'}), (2, \text{'Fl'}), (2, \text{'Hn'}), (3, \text{'Fl'}), (3, \text{'Hn'})\}. \quad (\text{A.15})$$

■

Definition A.11. Function. A function, represented by an italic letter such as f or a non-italic short word such as \max or \cos , is a collection of rules that describe how elements of one set A called the *domain* are mapped to elements of another set B . A mathematical shorthand for the previous sentence is $f : A \rightarrow B$. The set denoted $f(A)$ and defined by $f(A) = \{f(a) : a \in A\}$ is called the *image* of the function. ■

Example A.12. With A and B defined as in (A.8), an example of a function is

$$f(a) = \begin{cases} 2, & \text{if } a = 1, \\ 3, & \text{if } a = 2, \\ 4, & \text{if } a = 3. \end{cases} \quad (\text{A.16})$$

The mathematics ' $f(a)$ ' is read ' f of a '. Convention stipulates that the *argument*, an element a of the domain A , is placed within parentheses or square brackets to the right of the function name, in this case f . The function states that $1 \in A$ maps to $2 \in B$, $2 \in A$ maps to $3 \in B$, and $3 \in A$ maps to $4 \in B$. Alternatively, one could write $f(1) = 2$, $f(2) = 3$, and $f(3) = 4$. It

would be more concise (and therefore preferable) to define $f : A \rightarrow B$ by

$$f(a) = a + 1, \quad a \in A. \quad (\text{A.17})$$

Such concise definitions of a function are not always possible. For instance, with A and B defined as in (A.8), let $g : A \rightarrow B$ be given by

$$g(a) = \begin{cases} 2, & \text{if } a = 1, \\ 3, & \text{if } a = 1, \\ 2, & \text{if } a = 2. \end{cases} \quad (\text{A.18})$$

This function defies attempts at concision. ■

Definition A.13. Well defined, onto, one-to-one, bijective, and invertible. A function $f : A \rightarrow B$ is said to be well defined if the mapping of each element $a \in A$ to $b \in B$ is unambiguous. (For example, f in A.16 is well defined, whereas g in A.18 is not well defined, as it is unclear whether $1 \in A$ should map to $2 \in B$ or $3 \in B$.) If for each element $b \in B$ of a function $f : A \rightarrow B$, there exists (at least) one element $a \in A$ such that $f(a) = b$, then f is said to be *onto*. Another property that a function $f : A \rightarrow B$ might exhibit is *one-to-oneness*. If for each element $a_1 \in A$, there is *no other* element $a_2 \in A$ such that $f(a_1) = f(a_2)$, then f is said to be *one-to-one*.

A function $f : A \rightarrow B$ that is both one-to-one and onto is called *bijective*. A function $f : A \rightarrow B$ is said to be *invertible* if there exists a function $f^{-1} : B \rightarrow A$ such that $f(a) = b$ if and only if $f^{-1}(b) = a$. It can be shown (but will not be shown here) that a function f is invertible if and only if it is bijective. ■

Example A.14. Here are some more examples of functions, exhibiting various combinations of one-to-one and onto properties.

$$f_1 : \mathbb{R} \rightarrow \mathbb{R}, \text{ by } f_1(x) = x^2, \quad (\text{A.19})$$

$$f_2 : \mathbb{Z} \rightarrow \mathbb{Z}, \text{ by } f_2(m) = m^3, \quad (\text{A.20})$$

$$f_3 : \mathbb{R}^2 \rightarrow \mathbb{R}, \text{ by } f_3[(x, y)] = x + y, \quad (\text{A.21})$$

$$f_4 : \mathbb{R} \rightarrow \mathbb{R}, \text{ by } f_4(x) = x^3, \quad (\text{A.22})$$

$$f_5 : \mathbb{R}^n \rightarrow \mathbb{R}, \text{ by } f_5[(x_1, x_2, \dots, x_n)] = \frac{1}{n}(x_1 + x_2 + \dots + x_n), \quad (\text{A.23})$$

$$f_6 : \mathbb{R}^n \rightarrow \mathbb{R}, \text{ by } f_6[(x_1, x_2, \dots, x_n)] = (x_1 \cdot x_2 \cdots x_n)^{(1/n)}, \quad (\text{A.24})$$

$$f_7 : \mathbb{R} \rightarrow [-1, 1], \text{ by } f_7(t) = t - t^3/3! + t^5/5! - t^7/7! + \dots, \quad (\text{A.25})$$

$$f_8 : \mathbb{R} \rightarrow [-1, 1], \text{ by } f_8(t) = 1 - t^2/2! + t^4/4! - t^6/6! + \dots. \quad (\text{A.26})$$

The function f_1 is neither one-to-one nor onto. Both 1^2 and $(-1)^2$ equal 1, so f_1 is not one-to-one. There is no real number x such that $x^2 = -1$, so f_1 is not onto. Without further explanation, f_2 is one-to-one but not onto, f_3 is not one-to-one but is onto, and f_4 is both one-to-one and onto. None of the functions f_5, f_6, f_7, f_8 are one-to-one, but they are all onto. The function f_5 is the arithmetic mean, and f_6 is the geometric mean, where ‘.’ is a more accepted sign than ‘ \times ’ for multiplying numbers. Writing out the functions (A.23)-(A.26) in full each time can be cumbersome, so a shorthand called *sigma notation* is used. For example, (A.23) can be re-written as

$$f_5(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n x_i, \quad (\text{A.27})$$

which reads ‘ f_5 of the vector \mathbf{x} equals 1 divided by n times the sum from i equals 1 to i equals n of x_i ’. The arithmetic mean of a vector \mathbf{x} is sometimes denoted $\bar{\mathbf{x}}$. Similarly,

$$f_6(\mathbf{x}) = \left(\prod_{i=1}^n x_i \right)^{(1/n)}, \quad (\text{A.28})$$

It is harder to cajole the functions f_7 and f_8 into sigma notation, but here they are:

$$\sin(t) = f_7(t) = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{(2i-1)!} t^{2i-1}, \quad (\text{A.29})$$

$$\cos(t) = f_8(t) = \sum_{i=0}^{\infty} \frac{(-1)^i}{(2i)!} t^{2i}. \quad (\text{A.30})$$

These functions are shown with their special names, sin (short for sine) and cos (short for cosine) respectively. Plots of these functions are shown in Fig. A.1. ■

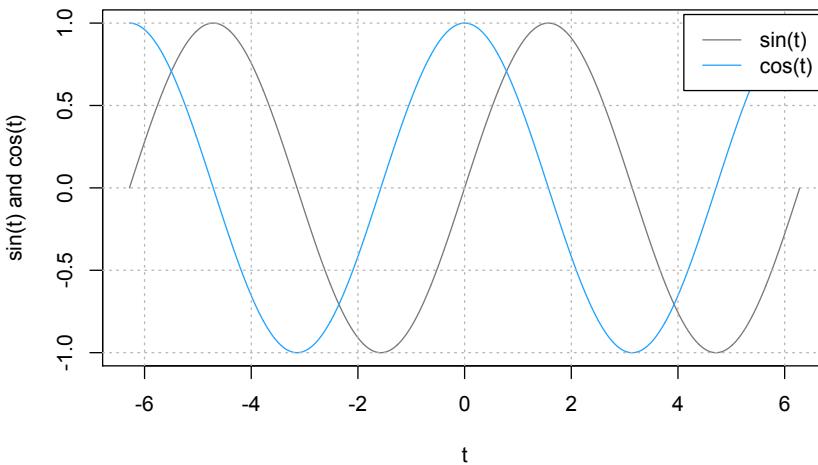


Figure A.1: Plots of the sinusoidal functions sine and cosine, given in expanded form in (A.25) and (A.26) respectively.

Definition A.15. Combination of functions. For two functions $f : A \rightarrow B$ and $g : B \rightarrow C$, the *combination* $f \circ g : A \rightarrow C$ is defined by $g(f(a))$, where $a \in A$. For n functions $f_1 : A_0 \rightarrow A_1, f_2 : A_1 \rightarrow A_2, \dots, f_n : A_{n-1} \rightarrow A_n$, the *combination* $f_1 \circ f_2 \circ \dots \circ f_n : A_0 \rightarrow A_n$ is defined by $f_n(f_{n-1}(\dots(f_2(f_1(a_0)))\dots))$, where $a_0 \in A_0$. Often this is called composition of functions, but the term *combination* will be used here, to avoid confusion with musical composition. ■

Example A.16. Let $f_1 : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ be defined by $f_1(a_0) = 2\pi 440 a_0$, let $f_2 : \mathbb{R}_+ \rightarrow [-1, 1]$ be defined by $f_2(a_1) = \sin(a_1)$, and let $f_3 : [-1, 1] \rightarrow [-0.7, 0.7]$ be defined by $f_3(a_2) = 0.7a_2$. Then

$$f_1 \circ f_2 \circ f_3(a_0) = f_3(f_2(f_1(a_0))) \quad (\text{A.31})$$

$$= 0.7 \sin(2\pi 440 t) \quad (\text{A.32})$$

$$= 0.7 \left(\sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{(2i-1)!} (2\pi 440 a_0)^{2i-1} \right). \quad (\text{A.33})$$

■

Definition A.17. Binary operator. A binary operator is a function $f : A^2 \rightarrow A$. It is common to see elements of the argument for a binary operator written either side of the function symbol, rather than to the right. That is, $x + y$ is equivalent to and more common than $f_3[(x, y)]$, where f_3 was defined in (A.21). The general symbol for a binary operator is ‘ \circ ’, so one might see $x \circ y$. This should not be confused with the same symbol used for *combinations of functions* (Def. A.15). Sometimes the symbol is dropped altogether, so $xy = x \circ y$. Apart from addition over the real numbers, other examples of binary operators include subtraction and multiplication. ■

Definition A.18. Modulo arithmetic. It can be shown (but will not be shown here) that for $a \in \mathbb{N}$, an arbitrary integer $n \in \mathbb{Z}$ can be expressed uniquely as $n = am + b$, where $b, m \in \mathbb{Z}$, and $0 \leq b < a$. For example, fixing $a = 12$, we have $61 = 12 \cdot 5 + 1$, and $-7 = 12 \cdot (-1) + 5$. This fact is used to define a function $f : (\mathbb{Z} \times \mathbb{N}) \rightarrow \mathbb{Z}_a$ by $f[(n, a)] = b$, where $n = am + b$ for integers b, m , and $0 \leq b < a$. In words, it is said that ‘ n equals b modulo a ’.

For two elements $x, y \in \mathbb{Z}_a$, the binary operator of addition modulo a , written ‘ $+_a$ ’, is defined by

$$x +_a y = \begin{cases} x + y, & \text{if } x + y < a, \\ x + y - a, & \text{otherwise.} \end{cases} \quad (\text{A.34})$$

■

Definition A.19. Group. A *group* (G, \circ) consists of a set G and a binary operation \circ , such that:

1. *Closure.* For all $x, y \in G$, $x \circ y \in G$.
2. *Associativity.* For all $x, y, z \in G$, $(x \circ y) \circ z = x \circ (y \circ z)$.
3. *Identity.* There exists $e \in G$ such that $e \circ x = x \circ e = x$, for all $x \in G$.
4. *Inverses.* For each $x \in G$, there exists an element written x^{-1} such that $x^{-1} \circ x = x \circ x^{-1} = e$.

■

Example A.20. It can be verified that each of $(\mathbb{R}, +)$, (\mathbb{R}^*, \times) , (\mathbb{R}_+^*, \times) , $(\mathbb{Q}, +)$, (\mathbb{Q}^*, \times) , $(\mathbb{Z}, +)$, and $(\mathbb{Z}_a, +_a)$ satisfy the conditions for closure, associativity, identity, and inverses given above, and so are groups.

Let x be defined as the clockwise rotation of a triangle about a point by 120° , let y be the same but by 240° , let e be the identity rotation (by 0°), and

let the binary operator \circ be defined as combinations of rotations, so that, for example, $x \circ x = x^2 = y$. Then letting $G = \{e, x, y\}$, it can be verified that (G, \circ) is a group.

Another group (G, \circ) consists of rotations of the cube that map vertices to vertices. Again, the binary operator is defined as combinations of rotations. The set G consists of twenty-four elements, one of which z is illustrated in Fig. A.2. The left-hand side of Fig. A.2 shows a cube with vertices labelled $\omega_1, \omega_2, \dots, \omega_8$. In the middle of Fig. A.2, an axis is drawn through vertices ω_1 and ω_7 . If the cube is rotated by 120° about this axis as indicated by the arrow, then the vertices assume new positions, shown on the right-hand side of Fig. A.2. The next definition is motivated by the way in which the vertices of the cube are affected by such rotations. ■

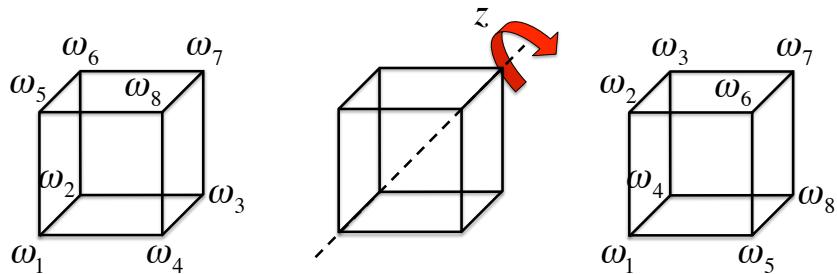


Figure A.2: The cube to the left has vertices labelled $\omega_1, \omega_2, \dots, \omega_8$. The cube in the middle is subject to a rotation by 120° about the axis through ω_1 and ω_7 . The cube to the right shows the vertices in their post-rotation positions.

Definition A.21. Action of a group on a set. Let (G, \circ) be a group and Ω be a set. We say that G acts on Ω if the function $f : G \times \Omega \rightarrow \Omega$ satisfies the following conditions for each $\omega \in \Omega$:

1. For the identity element $e \in G$, $f(e, \omega) = \omega$.

2. For all $x, y \in G$, $f(x, f(y, \omega)) = f(x \circ y, \omega)$. ■

Example A.22. If, as above, the group (G, \circ) consists of rotations of the cube that map vertices to vertices, and $\Omega = \{\omega_1, \omega_2, \dots, \omega_8\}$ is the set of cube vertices, then G acts on Ω . With $z \in G$ defined as the rotation by 120° as illustrated in Fig. A.2, we have $f(z, \omega_1) = \omega_1$, and $f(z, \omega_2) = \omega_5$, and so on.

If there is a bijection $b : \Omega \rightarrow G$ for a set Ω and a group (G, \circ) , then G acts on Ω via the function $f : G \times \Omega \rightarrow \Omega$, defined by $f(x, \omega) = b^{-1}(x \circ b(\omega))$. ■

Definition A.23. Equivalence relation. A *relation* on a set S is a subset R of $S \times S$, indicating the ordered pairs of elements of S that are *related*. For $(s, t) \in R$, we write $s \sim t$, meaning s and t are related.

A relation is said to be:

- *Reflexive* if $s \sim s$ for all $s \in S$.
- *Symmetric* if $s \sim t$ implies $t \sim s$ for all $s, t \in S$.
- *Transitive* if $(s \sim t \text{ and } t \sim u)$ implies $s \sim u$ for all $s, t, u \in S$.

An *equivalence relation* on a set S is a relation that is reflexive, symmetric, and transitive. For an equivalence relation R on a set S , two elements $s, t \in S$ such that $s \sim t$ are said to be in the same *equivalence class*. ■

Example A.24. Let $S = \mathbb{Z}$, $s, t \in S$, and R be a relation on S such that $s \sim t$ if $s \leq t$. It can be checked that the relation *is* reflexive, *is not* symmetric, and *is* transitive.

Now let $S = \mathbb{R}^2$ be the set of points in the plane, $(s_x, s_y), (t_x, t_y) \in S$, and R be a relation on S such that $(s_x, s_y) \sim (t_x, t_y)$ if $\sqrt{s_x^2 + s_y^2} = \sqrt{t_x^2 + t_y^2}$.

In words, the point (s_x, s_y) is related to the point (t_x, t_y) if they are the same distance from the origin. It can be checked that this *is* an equivalence relation, and each *equivalence class* is a circle with centre the origin. ■

Definition A.25. Sample correlation coefficient. The sample correlation coefficient (also known as the Pearson product-moment correlation coefficient) of two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ is a function $f : (\mathbb{R}^n \times \mathbb{R}^n) \rightarrow [-1, 1]$ given by

$$f[(\mathbf{x}, \mathbf{y})] = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}. \quad (\text{A.35})$$

■

Example A.26. For the vectors $\mathbf{x} = (-8, 7, 2, 0)$ and $\mathbf{y} = (-5, 4, 0, 1)$, the sample correlation coefficient is $f[(\mathbf{x}, \mathbf{y})] = 0.971$. Keeping \mathbf{x} the same and letting $\mathbf{z} = (9, -6, 4, 4)$, the sample correlation coefficient is $f[(\mathbf{x}, \mathbf{z})] = -0.923$. Keeping \mathbf{x} the same and letting $\mathbf{w} = (1, 0, 8, -4)$, the sample correlation coefficient is $f[(\mathbf{x}, \mathbf{w})] = 0.072$.

So the sample correlation coefficient measures the strength of the linear relationship between two vectors, returning values close to 1 for a positive linear relationship, values close to -1 for a negative linear relationship, and values close to 0 for no linear relationship. ■

Definition A.27. Countable and cardinality. A set A is said to be *countable* (or *countably infinite*) if there exists a one-to-one function $f : A \rightarrow \mathbb{N}$. Otherwise it is *uncountable*. A set $A = \{a_1, a_2, \dots, a_n\}$ with a finite number of elements is said to have *cardinality* $n = |A|$. ■

Example A.28. The sets $A = \{2, 1, 3\}$ and $B = \{b_1, b_2, \dots, b_n\}$ are countable. The sets \mathbb{Z} and \mathbb{Q} are countable. In the latter case, there is an elegant

proof that consists of constructing the matrix

$$\mathbf{A} = \left(\begin{array}{cccccc} \frac{1}{1} & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \dots \\ \frac{2}{1} & \frac{2}{2} & \frac{2}{3} & \frac{2}{4} & \frac{2}{5} & \dots \\ \frac{3}{1} & \frac{3}{2} & \frac{3}{3} & \frac{3}{4} & \frac{3}{5} & \dots \\ \frac{4}{1} & \frac{4}{2} & \frac{4}{3} & \frac{4}{4} & \frac{4}{5} & \dots \\ \frac{5}{1} & \frac{5}{2} & \frac{5}{3} & \frac{5}{4} & \frac{5}{5} & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array} \right). \quad (\text{A.36})$$

The list $(\frac{1}{1}, \frac{2}{1}, \frac{1}{2}, \frac{1}{3}, \frac{2}{2}, \frac{3}{1}, \frac{4}{1}, \frac{3}{2}, \dots)$ is formed by tracing a line over successive diagonals of \mathbf{A} . If each element $\frac{a}{b}$ in the list is proceeded by $-\frac{a}{b}$, and zero placed at the very beginning, then the rational numbers \mathbb{Q} have been put in a one-to-one correspondence with the natural numbers \mathbb{N} .

The irrational numbers \mathbb{I} and real numbers \mathbb{R} are uncountable. Intervals, such as (a, b) and $[a, b]$, are uncountable. ■

Definition A.29. Sample space and event (Ross, 2006). The sample space of an experiment, denoted S , is the set of all possible outcomes. An event E is a subset of the sample space. The event E is said to have occurred if the experiment's outcome is contained in E . ■

Example A.30. Ross (2006). In an experiment that consists of rolling two dice, the sample space consists of thirty-six vectors

$$S = \{(i, j) : i, j \in \{1, 2, \dots, 6\}\}, \quad (\text{A.37})$$

where the outcome (i, j) occurs if i appears on the leftmost die and j on the

rightmost die.

In an experiment that consists of measuring the lifetime of a transistor in hours, the sample space consists of all nonnegative real numbers

$$S = \{x \in \mathbb{R} : x \geq 0\}. \quad (\text{A.38})$$

■

Definition A.31. Union, intersection, complement, and mutual exclusivity (Ross, 2006). The union and intersection of two sets were defined in Def. B.1. These definitions apply also to events, and can be extended from two to a countable number of events using a form of sigma notation (cf. p. 299) as follows. If E_1, E_2, \dots are events, the union of these events, denoted by $\bigcup_{i=1}^{\infty} E_i$, is defined to be that event consisting of all outcomes that are in E_i for at least one value of i , where $i = 1, 2, \dots$. Similarly, the intersection of these events, denoted by $\bigcap_{i=1}^{\infty} E_i$, is defined to be the event consisting of those outcomes that are in all of the events E_i , where $i = 1, 2, \dots$

For an event E , the event E^C , called the *complement* of E , contains all events in the sample space S that are not in E . Recalling the definition of *set difference* (Def. B.1), $E^C = S \setminus E$.

For two events E and F , if $E \cap F = \emptyset$, where \emptyset is the empty set, then E and F are said to be *mutually exclusive*. ■

Axioms A.32. Axioms of probability (Ross, 2006).

1. For an experiment with sample space S , and an arbitrary event $E \subseteq S$, there exists a well defined function $\mathbb{P} : E \rightarrow [0, 1]$.
2. $\mathbb{P}(S) = 1$.

3. For arbitrary, mutually exclusive events E_1, E_2, \dots , that is $E_i \cap E_j = \emptyset$

when $i \neq j$,

$$\mathbb{P}\left(\bigcup_{i=1}^{\infty} E_i\right) = \sum_{i=1}^{\infty} \mathbb{P}(E_i). \quad (\text{A.39})$$

The notation $\mathbb{P}(E)$ is referred to as the probability of the event E . Results such as $\mathbb{P}(E^c) = 1 - \mathbb{P}(E)$ can be derived from the axioms. ■

Example A.33. If two fair dice are rolled, what is the probability that the sum of the upturned faces equals 8?

Solution. The state space for rolling two dice was given in Example A.30, and the events of interest are $E_1 = (2, 6)$, $E_2 = (3, 5)$, $E_3 = (4, 4)$, $E_4 = (5, 3)$, and $E_5 = (6, 2)$. The five events are mutually exclusive and equiprobable, each with probability $\frac{1}{36}$. So the desired probability is $\frac{5}{36}$. ■

Definition A.34. Conditional probability and Bayes' formula (Ross, 2006). For two events E and F , if $\mathbb{P}(F) > 0$, then

$$\mathbb{P}(E | F) = \frac{\mathbb{P}(E \cap F)}{\mathbb{P}(F)}. \quad (\text{A.40})$$

The left-hand side of this equation reads ‘the probability that the event E occurs given (or conditional on) the event F having occurred’.

Now suppose that F_1, F_2, \dots, F_n are mutually exclusive events such that $\bigcup_{i=1}^n F_i = S$, where S is the sample space. Then for some event E , it can be shown (but will not be shown here) that

$$\mathbb{P}(E) = \sum_{i=1}^n \mathbb{P}(E | F_i) \mathbb{P}(F_i). \quad (\text{A.41})$$

Equation (A.41) is sometimes called the *law of total probability*.

Bayes' formula. With E and F_1, F_2, \dots, F_n defined as above,

$$\mathbb{P}(F_j | E) = \frac{\mathbb{P}(E \cap F_j)}{\mathbb{P}(E)} \quad (\text{A.42})$$

$$= \frac{\mathbb{P}(E | F_j)\mathbb{P}(F_j)}{\sum_{i=1}^n \mathbb{P}(E | F_i)\mathbb{P}(F_i)}, \quad (\text{A.43})$$

where $j \in \{1, 2, \dots, n\}$ is arbitrary. Equation (A.43), Bayes' formula, follows from (A.40) and (A.41). ■

Example A.35. Adapted from Ross (2006). A note is played and a listener is asked to declare the pitch of the note. The listener is able to try out (play) *one* pitch before answering, and is told of three equally likely possibilities. We assume the listener is competent to the extent that if they try out an incorrect pitch, they will not declare that pitch. Let $1 - \beta_i$ denote the probability that the listener tries out and declares the i th pitch to be that of the note, when in fact this is correct, $i = 1, 2, 3$. The quantities $\beta_1, \beta_2, \beta_3$ are sometimes referred to as *overlook probabilities*. What is the conditional probability that the i th pitch is that of the note, given the listener tries out but does not declare the first pitch to be that of the note?

Solution. Let F_i , $i = 1, 2, 3$, be the event that the i th pitch is that of the note, and let E be the event that the listener tries out but does not declare

the first pitch to be that of the note. From Bayes' formula (A.43),

$$\mathbb{P}(F_1 | E) = \frac{\mathbb{P}(E \cap F_1)}{\mathbb{P}(E)} \quad (\text{A.44})$$

$$= \frac{\mathbb{P}(E | F_1)\mathbb{P}(F_1)}{\sum_{i=1}^3 \mathbb{P}(E | F_i)\mathbb{P}(F_i)} \quad (\text{A.45})$$

$$= \frac{\beta_1 \cdot \frac{1}{3}}{\beta_1 \cdot \frac{1}{3} + 1 \cdot \frac{1}{3} + 1 \cdot \frac{1}{3}} \quad (\text{A.46})$$

$$= \frac{\beta_1}{\beta_1 + 2}. \quad (\text{A.47})$$

For $j = 2, 3$,

$$\mathbb{P}(F_j | E) = \frac{\mathbb{P}(E \cap F_j)}{\mathbb{P}(E)} \quad (\text{A.48})$$

$$= \frac{1 \cdot \frac{1}{3}}{\beta_1 \cdot \frac{1}{3} + 1 \cdot \frac{1}{3} + 1 \cdot \frac{1}{3}} \quad (\text{A.49})$$

$$= \frac{1}{\beta_1 + 2}. \quad (\text{A.50})$$

It is worth pointing out that the amount in (A.47) is less than one third, and the amount in (A.50) is more than one third. This makes intuitive sense: if the listener tries out but does not declare the first pitch, then the initial probabilities of the i th pitch being correct ($= \frac{1}{3}$, $i = 1, 2, 3$) are updated in favour of the second and third pitches. Also, the closer the overlook probability β_1 is to one, the closer the amounts in (A.47) and (A.50) are to one third. ■

Definition A.36. Independent events (Ross, 2006). Two events E and F are said to be *independent* if $\mathbb{P}(E \cap F) = \mathbb{P}(E) \cdot \mathbb{P}(F)$. Two events E and F that are not independent are said to be *dependent*.

The events E_1, E_2, \dots, E_n are said to be independent if for any subset $\{E_{i_1}, E_{i_2}, \dots, E_{i_m}\}$ of them,

$$\mathbb{P}\left(\bigcap_{j=1}^m E_{i_j}\right) = \mathbb{P}(E_{i_1}) \cdot \mathbb{P}(E_{i_2}) \cdots \mathbb{P}(E_{i_m}). \quad (\text{A.51})$$

■

Example A.37. Ross (2006). Suppose, as in Example A.30, that two fair dice are rolled. Let E be the event that the sum of the dice is 8, and F be the event that the leftmost dice shows 3. Then

$$\mathbb{P}(E \cap F) = \mathbb{P}(\{(3, 5)\}) = \frac{1}{36}. \quad (\text{A.52})$$

Having determined $\mathbb{P}(E) = \frac{5}{36}$ in Example A.33,

$$\mathbb{P}(E) \cdot \mathbb{P}(F) = \frac{5}{36} \cdot \frac{1}{6} = \frac{5}{216}. \quad (\text{A.53})$$

Therefore, E and F are dependent. Suppose we let E be the event that the sum of the dice is 7. Now

$$\mathbb{P}(E \cap F) = \mathbb{P}(\{(3, 4)\}) = \frac{1}{36}, \quad (\text{A.54})$$

and

$$\mathbb{P}(E) \cdot \mathbb{P}(F) = \frac{1}{6} \cdot \frac{1}{6} = \frac{1}{36}. \quad (\text{A.55})$$

Therefore, E and F are independent. ■

Definition A.38. Discrete random variable and probability mass function (Ross, 2006). Let S be a sample space and E_1, E_2, \dots be mutually

exclusive events such that $\bigcup_{i=1}^{\infty} E_i = S$. A *discrete random variable* is a function $X : E_i \rightarrow \mathbb{R}$, well-defined for each value of $i = 1, 2, \dots$. An arbitrary element of the image of X is denoted by a lowercase letter, such as x , or x_1, x_2, \dots if there are many. When an event E_i from the sample space is observed as the outcome, it is said that the random variable X takes or assumes a value x . The probability of the event E_i , denoted $\mathbb{P}(E_i)$, is equal to the probability that X takes or assumes the value x , written $\mathbb{P}(X = x)$.

The *probability mass function* of a discrete random variable X is defined by

$$p(x) = \mathbb{P}(X = x). \quad (\text{A.56})$$

The domain of the function p is the countable set of values $\{x_1, x_2, \dots\}$ that X can take. A probability mass function inherits properties from the Axioms of Probability (cf. Def. A.32). First, $p(x_i) \geq 0$, where $i = 1, 2, \dots$. Second, $\sum_{i=1}^{\infty} p(x_i) = 1$. ■

Example A.39. Ross (2006). Often we are interested in some function of the outcome of an experiment, rather than the actual outcome itself. For instance, when rolling two dice, we might be interested in the sum of the two dice, and not really concerned about the separate values of each die. Random variables enable focus on a function of the experiment's outcome. Letting X be a random variable for the sum of two rolled dice, we have

$$\mathbb{P}(X = 2) = \mathbb{P}(\{(1, 1)\}) = \frac{1}{36}, \quad (\text{A.57})$$

$$\mathbb{P}(X = 3) = \mathbb{P}(\{(1, 2), (2, 1)\}) = \frac{1}{18}, \quad (\text{A.58})$$

$$\mathbb{P}(X = 4) = \mathbb{P}(\{(1, 3), (2, 2), (3, 1)\}) = \frac{1}{12}, \quad (\text{A.59})$$

and so on. ■

Definition A.40. Bernoulli and binomial random variables (Ross, 2006). Suppose that the outcome of an experiment is either a success, in which case the discrete random variable X takes the value 1, or a failure, in which case X takes the value 0. Then the probability mass function of X is

$$p(x) = \mathbb{P}(X = x) = \begin{cases} 1 - \theta, & \text{if } x = 0, \\ \theta, & \text{if } x = 1, \end{cases} \quad (\text{A.60})$$

where $0 \leq \theta \leq 1$ is the probability that the outcome of the experiment is a success. The random variable X is called a Bernoulli random variable.

Now suppose that in n independent experiments, each experiment has a successful outcome with probability θ , and failed outcome with probability $1 - \theta$. A discrete random variable Y that represents the number of successes that occur in n experiments is called a binomial random variable with parameters n, θ . We write $Y \sim B(n, \theta)$ as a shorthand to mean that Y is binomially distributed with parameters n, θ . The probability mass function is

$$p(i) = \binom{n}{i} \theta^i (1 - \theta)^{n-i}, \quad i = 0, 1, \dots, n. \quad (\text{A.61})$$
■

Definition A.41. Expectation, variance, and entropy of a discrete random variable. Let X be a discrete random variable taking the values x_1, x_2, \dots , and let X have the probability mass function p . Then the

expectation (also called the mean) of X , denoted $\mathbb{E}(X)$, is

$$\mu = \mathbb{E}(X) = \sum_{i=1}^{\infty} x_i p(x_i). \quad (\text{A.62})$$

The expectation is a weighted average of the values assumed by X . The *variance* of X , denoted $\mathbb{V}(X)$, is

$$\mathbb{V}(X) = \mathbb{E}[(X - \mu)^2] = \mathbb{E}(X^2) - \mu^2. \quad (\text{A.63})$$

The variance quantifies the average square distance between X and its mean.

Sometimes, we talk about a probability mass function as a *probability vector*. That is, the vector \mathbf{p} with i th element $p_i = p(x_i)$, where $i = 1, 2, \dots$. For the discrete random variable X with probability vector $\mathbf{p} = (p_1, p_2, \dots, p_n)$, the *entropy* of X , denoted $H(X)$, is

$$H(X) = - \sum_{i=1}^n p_i \log_2 p_i, \quad (\text{A.64})$$

where $-\log_2 p_i$ is known as the *information content* (Shannon, 1948). The entropy of a random variable quantifies the *uncertainty* associated with its outcome, with small positive values for low uncertainty, and large positive values for high uncertainty. ■

Example A.42. Suppose that X is a discrete random variable representing the sum of two rolled, *fair* dice. In Example A.39 we began calculating the probability mass function of X , which will now be given as a probability vector,

$$\mathbf{p} = \left(\frac{1}{36}, \frac{1}{18}, \frac{1}{12}, \frac{1}{9}, \frac{5}{36}, \frac{1}{6}, \frac{5}{36}, \frac{1}{9}, \frac{1}{12}, \frac{1}{18}, \frac{1}{36} \right). \quad (\text{A.65})$$

After some calculations, we find $\mathbb{E}(X) = 7$, $\mathbb{V}(X) \approx 5.83$, and $H(X) \approx 3.27$.

Now suppose that Y is a discrete random variable representing the sum of two rolled dice, where one die is fair and the other biased towards higher scores, so that it shows $1, 2, \dots, 6$ with respective probabilities $\frac{1}{32}, \frac{1}{32}, \frac{1}{16}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}$.

The probability vector \mathbf{q} for Y is

$$\mathbf{q} = \left(\frac{1}{192}, \frac{1}{96}, \frac{1}{48}, \frac{1}{24}, \frac{1}{12}, \frac{1}{6}, \frac{31}{192}, \frac{5}{32}, \frac{7}{48}, \frac{1}{8}, \frac{1}{12} \right). \quad (\text{A.66})$$

After some calculations, we find $\mathbb{E}(Y) \approx 8.53$, $\mathbb{V}(Y) \approx 4.57$, and $H(Y) \approx 3.07$. The biased die causes the expected value of Y to increase slightly compared with that of X . At the same time, the variance and entropy of Y are smaller respectively than the variance and entropy of X .

In general, it is possible to redistribute the mass of a probability vector \mathbf{p} , giving \mathbf{q} , such that for the corresponding random variables X and Y , $\mathbb{V}(X) < \mathbb{V}(Y)$, and $H(X) > H(Y)$. ■

Example A.30 contains an experiment where the lifetime of a transistor is measured in hours. The sample space was all nonnegative real numbers. The nonnegative real numbers, \mathbb{R}_+ , are *uncountable* (cf. Def. A.27), just like the real numbers, \mathbb{R} . Discrete random variables cannot be used to model the exact outcome of such experiments, as their image must be a *countable* set. Another type of random variable is required, called a *continuous random variable*.

Definition A.43. Continuous random variable and probability density function (Ross, 2006). We say that X is a *continuous random variable* if there exists a nonnegative function f , defined for all $x \in \mathbb{R}$, such that for

each set $A \subseteq \mathbb{R}$,

$$\mathbb{P}(X \in A) = \int_A f(x) dx, \quad (\text{A.67})$$

that is, the area between the curve $f(x)$ and the x -axis over which A is defined. The function f is known as the *probability density function* of X . It has the property that

$$1 = \mathbb{P}(X \in \mathbb{R}) = \int_{-\infty}^{\infty} f(x) dx, \quad (\text{A.68})$$

as X must belong to some interval.

Probability statements concerning X are answered in terms of f . For example, the probability that X takes a value in the interval $[a, b]$ is given by

$$\mathbb{P}(a \leq X \leq b) = \int_a^b f(x) dx. \quad (\text{A.69})$$

Definitions for the expectation, variance, and entropy of a continuous random variable are analogous to the discrete definitions, replacing sums with integrals. ■

Example A.44. The lifetime of a transistor measured in hours can be modelled by a continuous random variable X with probability density function

$$f(x) = \begin{cases} \frac{1}{\mu} e^{-x/\mu}, & \text{if } x \geq 0, \\ 0, & \text{if } x < 0, \end{cases} \quad (\text{A.70})$$

where $\mu > 0$ is an arbitrary constant. Supposing a value of $\mu = 100$, what is the probability that a transistor will work between 80 and 140 hours before breaking?

Solution.

$$\mathbb{P}(80 \leq X \leq 140) = \int_{80}^{140} \frac{1}{100} e^{-x/100} dx \quad (\text{A.71})$$

$$= -e^{-x/100} \Big|_{80}^{140} \quad (\text{A.72})$$

$$= e^{-0.8} - e^{-1.4} \quad (\text{A.73})$$

$$\approx .203. \quad (\text{A.74})$$

■

Definition A.45. Normal random variable (Ross, 2006). We say that X is a normal random variable, or that X is normally distributed, with parameters μ and σ^2 if the density of X is given by

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/2\sigma^2}, \quad (\text{A.75})$$

where $x \in \mathbb{R}$. The notational shorthand $X \sim N(\mu, \sigma^2)$ means that X is normally distributed with parameters μ and σ^2 . ■

Questions of a statistical nature are often answered in terms of the normal distribution, or in terms of related distributions, such as the t -distribution or F -distribution. There is an established method called *hypothesis testing* for stating and addressing questions concerning statistical significance. The following example gives a flavour for hypothesis tests, with more details being available elsewhere (Lunn, 2007a; Daly et al., 1995).

Example A.46. A previously unknown collection of Baroque 'cello concertos claimed to be by Antonio Vivaldi (1678-1741) is bequeathed to a library. The

Table A.1: The rhythmic density of various opening movements from known and supposed Vivaldi 'cello concertos. Data fabricated for the purpose of the example.

Bequeathed concertos		Library concertos		
6.44	1.86	5.14	3.15	6.15
3.66	3.67	4.19	4.29	5.29
4.58	4.04	5.37	4.46	5.81
5.06	3.32	4.75	5.33	4.42
3.09		4.69	4.82	
5.14		6.76	5.10	

librarian wishes to test this claim, so for each bequeathed concerto and for each Vivaldi 'cello concerto already held by the library, they calculate the rhythmic density of the opening movement. We can assume that rhythmic density is an appropriate aspect of the music to quantify, and fabricate some data for the purpose of the example (see Table A.1). The librarian wants to know whether these two sets of measurements constitute evidence of a different composer. The so-called *null hypothesis* (sometimes denoted H_0) is that the two samples have underlying distributions with the same mean. The *alternative hypothesis* (or H_1) is that the two samples have underlying distributions with different means.

The difference between the means of each sample is $-0.897 = 4.086 - 4.983$. Is this difference significant, taking into consideration the size of and variation within each sample? We will not go into the details, but the difference in means is weighted by 0.416, and the ratio $-0.897/0.416 \approx -2.155$ is supposed to be an observation from a t -distribution (denoted T and similar to the normal distribution in Def. A.45) with twenty-four so-called degrees

of freedom. It can be checked that $p = \mathbb{P}(|T| > 2.155) = .041$, so there is only a probability of .041 that the two samples have underlying distributions with the same mean. A typical cutoff point for rejecting the null hypothesis in favour of the alternative is $\alpha = .05$. As we have observed a p -value less than α , the null hypothesis is rejected. Giving a musical interpretation of this statistical result, there is evidence that the two sets of concertos may be by different composers. ■

Example A.47. Multiple linear regression. In Chapters 6 and 10, I consider *linear models* of the form

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p, \quad (\text{A.76})$$

where y is the rating given to an aspect of a music excerpt, x_1, x_2, \dots, x_p are variables for the excerpt under consideration, and $\alpha, \beta_1, \beta_2, \dots, \beta_p$ are regression coefficients. Suppose that listeners are presented with already-discovered repeated patterns from one or more pieces of music, and asked to rate each pattern's musical importance on a scale from 1 (not at all important) to 10 (highly important). The rating given by a listener, which is more generally known as the *response*, is represented by y in (A.76). The variable x_1 could represent cardinality—the number of notes contained in one occurrence of a pattern. The next variable x_2 could represent the number of occurrences of a pattern in a particular excerpt, etc. Linear means linear in the coefficients, so linear models are a very broad family of functions. For instance, both of

$$\text{rating} = \alpha + \beta_1 \text{cardinality} \cdot \text{occurrences}, \quad (\text{A.77})$$

$$\text{rating} = \alpha + \beta_1 \text{cardinality} + \beta_2 \text{occurrences}^2 \quad (\text{A.78})$$

are linear models.

In (A.76), the rating y is known, as are the values of the variables x_1, x_2, \dots, x_p , so the aim is to estimate the coefficients $\theta = (\alpha, \beta_1, \beta_2, \dots, \beta_p)$.

This is done by considering a *linear regression model*

$$Y_i = \underbrace{\alpha + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \cdots + \beta_p x_{i,p}}_{(\dagger)} + \varepsilon_i, \quad i = 1, 2, \dots, n. \quad (\text{A.79})$$

Capital letters for the n ratings Y_1, Y_2, \dots, Y_n indicate that these are random variables. On the right hand side there is an expression (\dagger) similar to that in (A.76). The notation has been altered so that $x_{i,j}$ is the value of the j th variable taken by the i th observation, where $i = 1, 2, \dots, n$, and $j = 1, 2, \dots, p$. These are often referred to as the *explanatory variables* or *predictors*, as they ‘explain’ the response (ratings). The terms $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n$ are non-observable, assumed to be independent and normally distributed random variables (cf. Def. A.45) with zero mean and constant variance. Sometimes they are referred to as *departures*, as their inclusion in (A.79) adjusts for (\dagger) ‘departing’ (being different) from Y_i . More commonly though, they are called *residual errors*. The coefficients $\theta = (\alpha, \beta_1, \beta_2, \dots, \beta_p)$ are estimated so as to minimise the sum of squares of the departures, $\sum_{i=1}^n \varepsilon_i^2$.

Continuing with the example of rating already-discovered patterns, suppose that a listener rates the musical importance of five patterns as 9, 2, 8, 4, and 1, and that these patterns have respective cardinalities 15, 3, 4, 7, 3, and respective occurrences 3, 2, 5, 3, 2. This information can be expressed

as

$$9 = \alpha + 15\beta_1 + 3\beta_2 + \varepsilon_1, \quad (\text{A.80})$$

$$2 = \alpha + 3\beta_1 + 2\beta_2 + \varepsilon_2, \quad (\text{A.81})$$

$$8 = \alpha + 4\beta_1 + 5\beta_2 + \varepsilon_3 \quad (\text{A.82})$$

$$4 = \alpha + 7\beta_1 + 3\beta_2 + \varepsilon_4 \quad (\text{A.83})$$

$$1 = \alpha + 3\beta_1 + 2\beta_2 + \varepsilon_5. \quad (\text{A.84})$$

The ratings y_1, y_2, \dots, y_5 are the observed values of the responses Y_1, Y_2, \dots, Y_5 . The above simultaneous equations can be expressed more concisely as

$$\mathbf{y} = \mathbf{X}\theta + \varepsilon, \quad (\text{A.85})$$

where

$$\mathbf{y} = \begin{pmatrix} 9 \\ 2 \\ 8 \\ 4 \\ 1 \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} 1 & 15 & 3 \\ 1 & 3 & 2 \\ 1 & 4 & 5 \\ 1 & 7 & 3 \\ 1 & 3 & 2 \end{pmatrix}, \quad \theta = \begin{pmatrix} \alpha \\ \beta_1 \\ \beta_2 \end{pmatrix}, \quad \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{pmatrix}. \quad (\text{A.86})$$

The matrix \mathbf{X} is often referred to as the *design matrix*. As mentioned above, the coefficients θ are estimated so as to minimise the sum of squares of the departures, $\sum_{i=1}^n \varepsilon_i^2$. The estimated coefficients are denoted with ‘hats’, $\hat{\theta} = (\hat{\alpha}, \hat{\beta}_1, \hat{\beta}_2, \dots, \hat{\beta}_p)$. For this example (and more generally) it can be shown

(but will not be shown here) that

$$\hat{\theta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (\text{A.87})$$

minimises $\sum_{i=1}^n \varepsilon_i^2$. Matrix transpose, multiplication, and inverses are defined in Def. A.3, and will be required to understand the following regression calculations:

$$\mathbf{X}^T \mathbf{X} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 15 & 3 & 4 & 7 & 3 \\ 3 & 2 & 5 & 3 & 2 \end{pmatrix} \begin{pmatrix} 1 & 15 & 3 \\ 1 & 3 & 2 \\ 1 & 4 & 5 \\ 1 & 7 & 3 \\ 1 & 3 & 2 \end{pmatrix} = \begin{pmatrix} 5 & 32 & 15 \\ 32 & 308 & 98 \\ 15 & 98 & 51 \end{pmatrix}, \quad (\text{A.88})$$

$$(\mathbf{X}^T \mathbf{X})^{-1} \approx \begin{pmatrix} 1.984 & -0.053 & -0.482 \\ -0.053 & 0.010 & 0.003 \\ -0.482 & -0.003 & 0.168 \end{pmatrix}, \quad (\text{A.89})$$

$$\mathbf{X}^T \mathbf{y} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 15 & 3 & 4 & 7 & 3 \\ 3 & 2 & 5 & 3 & 2 \end{pmatrix} \begin{pmatrix} 9 \\ 2 \\ 8 \\ 4 \\ 1 \end{pmatrix} = \begin{pmatrix} 24 \\ 204 \\ 85 \end{pmatrix}, \quad (\text{A.90})$$

$$\begin{aligned} \hat{\theta} &= \begin{pmatrix} \hat{\alpha} \\ \hat{\beta}_1 \\ \hat{\beta}_2 \end{pmatrix} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \\ &\approx \begin{pmatrix} 1.984 & -0.053 & -0.482 \\ -0.053 & 0.010 & 0.003 \\ -0.482 & -0.003 & 0.168 \end{pmatrix} \begin{pmatrix} 24 \\ 204 \\ 85 \end{pmatrix} \approx \begin{pmatrix} -4.126 \\ 0.449 \\ 2.017 \end{pmatrix}. \end{aligned} \quad (\text{A.91})$$

Therefore, in this example, the empirically derived formula for rating pattern importance is

$$\text{rating} = -4.126 + 0.449 \cdot \text{cardinality} + 2.017 \cdot \text{occurrences.} \quad (\text{A.92})$$

The model on which this formula is based is flawed: too few data ($n = 5$); only two explanatory variables considered (cardinality and occurrences); no assumptions about $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_5$ were checked. It gives a flavour, however, for fitting a statistical model and deriving a formula empirically.

Example A.48. Analysis of variance (ANOVA). ANOVA is a special case of multiple linear regression (see above), where the predictors are binary variables that represent different *blocks* and/or *treatments* of observations. For instance, suppose that twelve people participate in a study on aural music skills, which is intended to investigate the efficacy of a new aural skills training method. Using a pre-training aural test, the participants are divided (by the median test score) into two blocks of six skilled and six unskilled participants. Within each block, the first two randomly selected participants undertake no aural skills training, the next two receive training according to an existing method called the Kodály Method (Choksy, 1974), and the last two participants receive training according to a new method called *Augment*. The different types of training are referred to as treatments. The performance of participants is assessed by a post-training aural test, and the response variable we are considering for each participant is labelled *improvement*: post-training test score minus pre-training test score. The design matrix for

the regression will be

$$\begin{array}{l}
 \text{baseline} \quad \text{skilled} \quad \text{kod\'aly} \quad \text{augment} \\
 \hline
 1 \quad 1 \quad 0 \quad 0 \quad 0 \\
 2 \quad 1 \quad 0 \quad 0 \quad 0 \\
 3 \quad 1 \quad 0 \quad 1 \quad 0 \\
 4 \quad 1 \quad 0 \quad 1 \quad 0 \\
 5 \quad 1 \quad 0 \quad 0 \quad 1 \\
 6 \quad 1 \quad 0 \quad 0 \quad 1 \\
 7 \quad 1 \quad 1 \quad 0 \quad 0 \\
 8 \quad 1 \quad 1 \quad 0 \quad 0 \\
 9 \quad 1 \quad 1 \quad 1 \quad 0 \\
 10 \quad 1 \quad 1 \quad 1 \quad 0 \\
 11 \quad 1 \quad 1 \quad 0 \quad 1 \\
 12 \quad 1 \quad 1 \quad 0 \quad 1
 \end{array} = \mathbf{X}. \quad (\text{A.93})$$

Regression calculations analogous to (A.87)-(A.91) can be performed to derive a formula for the improvement between post- and pre-training aural test scores, based on whether a participant was skilled or unskilled, and whether they received no training, training in the Kod\'aly Method, or training in the Augment Method. The formula will be

$$\text{improvement} = \alpha + \beta_1 \cdot \text{skilled} + \beta_2 \cdot \text{kod\'aly} + \beta_3 \cdot \text{augment}, \quad (\text{A.94})$$

and so-called *contrasts*, such as $\beta_j - \beta_i$, will tell us about the effectiveness of one block or treatment over another. For example, if $\beta_3 - \beta_2$ is significantly greater than zero, this constitutes evidence that the new Augment Method

is more effective for aural skills training than the Kodály Method. As with the previous example, this model is flawed due to too few data for the experimental design, and assumptions about $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{12}$ must be checked after the regression.

Multiple linear regression and ANOVA are treated in much more detail elsewhere (Lunn, 2007b; Daly et al., 1995; Davison, 2003).



Introduction to music representations

Music has been defined as ‘humanly generated sounds that are good to listen to, and that are so for themselves and not merely for the message they convey’ (Cook, 1998, p. 4). The first part of the quotation precludes examples such as birdsong and a dripping tap, and the second part speech. ‘Organized sound’ (Goldman, 1961, p. 133) is another definition of music, attributed to the composer Edgard Varèse (1883-1965). Organisation can be thought of as occurring on several levels: a composer may have organised sounds that are realised subsequently by performers, or organisation and performance may be almost simultaneous, as in the case of improvisation; the mechanisms of the ears and brain of a listener organise the incident sound waves into percepts; and as well as sounds being organised, sounds organise and enhance various activities, such as physical exercise and religious ceremonies.

As one-sentence definitions of music go, the above quotations are acceptable, but for me, music is the experience of attending or performing a concert or gig, as well as private listening, playing, or singing. The sounds created during such experiences may or may not exist in other formats, e.g. recordings and transcriptions, but the experience of music itself is ephemeral and ineffable. Nevertheless, academics and critics attempt to understand and describe these experiences, sometimes without recourse to recordings or tran-

scriptions. In Chapter 5 (p. 84) the reader encountered critic and composer Robert Schumann’s (1810-1856) reaction to a premiere of a piece by Chopin. No doubt the evolution of staff notation (more about which in Sec. B.2.1) and recording have shaped the way music and musicology have developed. Simultaneously, documentation allows one to discuss certain parts of a piece of music unambiguously, while fostering an over-reliance on the score/recording as separate from the experience of music. Despite many genres of music being wholly or substantially orally transmitted, ‘the way of thinking about music that is built into schools and universities—and most books about music, for that matter—reflects the way music was in nineteenth-century Europe rather than the way it is today, anywhere. The result is a kind of credibility gap between music and how we think about it’ (Cook, 1998, p. ii).

I would argue that the main reason for this bias towards nineteenth-century European music is *convenience*. To show Fig. B.1, a portion from the song ‘To the end’ by Blur (1994), it is a legal requirement to obtain written permission from the copyright holder. Similarly with Fig. B.2, a transcription of this portion into staff notation. Transcriptions of non-European and jazz/pop/rock music are not always available for purchase, and making transcriptions oneself involves time and effort, and the potential for errors. Some transcriptions are more faithful to the audio than others. If played on a piano, Fig. B.2 probably contains the minimum amount of information necessary for a listener who is familiar with ‘To the end’ to recognise what is being played, but Fig. B.2 contains no reference to the range of instruments that are present in the audio. Other transcriptions, such as that shown in Fig. B.3, give a more accurate account of the audio. For instance, Fig. B.3

indicates which different instruments are playing, as well as what they are playing. If it is possible to express the same point or argument by quoting from a piece of nineteenth-century music, freely and without obtaining permission, then no wonder examples from this period overshadow examples from non-European genres and jazz/pop/rock, etc.

FIGURE REMOVED
FROM ELECTRONIC VERSION
FOR COPYRIGHT REASONS

Figure B.1: A portion of the audio signal from the song 'To the end' by Blur (1994). © Copyright 1994 by EMI Music Publishing. Used by permission.

**FIGURE REMOVED
FROM ELECTRONIC VERSION
FOR COPYRIGHT REASONS**

Figure B.2: Bars 1-8 of 'To the end' by Blur (1994). This is a transcription into staff notation of the audio signal shown in Fig. B.1. © Copyright 1994 by Universal Music Group.

FIGURE REMOVED
FROM ELECTRONIC VERSION
FOR COPYRIGHT REASONS

Figure B.3: Bars 12-13 (with upbeat) of 'Little wing' by Jimi Hendrix Experience (1967). © Copyright 1998 by Hal Leonard Corporation.

B.1 Audio, and mathematical definitions

In order to discuss audio in any detail, some mathematical notation and definitions need to be introduced. For a non-mathematician wishing to understand this and subsequent chapters, Appendix A contains many supplementary definitions and examples. Although audio representation of music forms only a small part of this thesis, the following definitions act as a theoretical foundation.

Definition B.1. Commonly used sets have their own labels, such as the natural numbers,

$$\mathbb{N} = \{1, 2, \dots\}, \quad (\text{B.1})$$

the integers

$$\mathbb{Z} = \{\dots, -2, -1, 0, 1, 2, \dots\}, \quad (\text{B.2})$$

and the rational numbers \mathbb{Q} , which is the set of all fractions a/b such that a is an integer and b is a natural number. The rational numbers can be expressed in set notation as

$$\mathbb{Q} = \left\{ \frac{a}{b} : a \in \mathbb{Z}, b \in \mathbb{N} \right\}. \quad (\text{B.3})$$

It can be shown (but will not be shown here) that between any two different elements x and z of \mathbb{Q} , there exists a number y that is not an element of \mathbb{Q} . We say that y is an irrational number. A well-known example of an irrational number is $\pi = 3.141\dots$, the ratio of a circle's circumference to its diameter. There is no standard notation for the set of all irrational numbers, so \mathbb{I} will be used. The real numbers are defined as

$$\mathbb{R} = \mathbb{Q} \cup \mathbb{I}. \quad (\text{B.4})$$

Sometimes it is necessary to consider real numbers that are between certain limits. For instance: \mathbb{R}_+ is the set of all real numbers that are greater than or equal to zero; (a, b) is the set of all real numbers that are greater than a and less than b ; $[a, b]$ is the set of all real numbers that are greater than or equal to a and less than or equal to b ; combinations of brackets are acceptable as well, so $[a, b)$ is the set of all real numbers that are greater than or equal to a and less than b , etc. Often these sets are called intervals, but the term will be avoided here, to avoid confusion with musical intervals. It should be clear from the context whether $(0, 1)$ is an interval or a vector/list consisting of two numbers.

The set of all integers that are greater than or equal to zero and less than a is denoted \mathbb{Z}_a . In set notation,

$$\mathbb{Z}_a = \{m \in \mathbb{Z} : 0 \leq m < a\} = \{0, 1, \dots, a-2, a-1\}. \quad (\text{B.5})$$

For a set A , sometimes I use A^* to denote ‘ A with zero removed’, or $A \setminus \{0\}$ (cf. Def. B.1). Finally, the set with no elements is called the empty set, written \emptyset . ■

Definition B.2. Audio and signal. When music is recorded using a microphone, the membrane of the microphone is displaced by the incident sound waves and this displacement is mapped to a window, typically $[-1, 1]$, by the computer hardware to which the microphone is connected (Rumsey and McCormick, 2009). The displacement of the membrane is measured at regularly spaced time points (44 100 measurements per second is a standard) and stored as a vector

$$\mathbf{y} = (y_0, y_1, \dots, y_{n-1}). \quad (\text{B.6})$$

An arbitrary element y_i of \mathbf{y} , where i is some integer from 0 to $n - 1$, is called the i th sample. It is an element of $[-1, 1]$, and corresponds to the displacement at time $i/44\,100$ seconds. If the membrane is not displaced at time $i/44\,100$ s, then $y_i = 0$. When more than one microphone is being employed simultaneously during a recording, there will be a vector for each. Using an electromagnetic transducer to drive a speaker's membrane so that it is displaced according to a vector \mathbf{y} will give a listener the impression that they are hearing the music that was originally recorded and stored as \mathbf{y} . As the changes in displacement occur so quickly, the perception is one of continuous sound. The term *audio* refers to the sounds produced by playing a recording (also by live performance). The corresponding vector \mathbf{y} is often referred to as the *signal*. ■

FIGURE REMOVED FROM ELECTRONIC VERSION FOR COPYRIGHT REASONS

Figure B.4: A zoomed-in portion (compared to Fig. B.1) of the audio signal from the song 'To the end' by Blur (1994). © Copyright 1994 by EMI Music Publishing. Used by permission.

Example B.3. An example of a signal was given in Fig. B.1. It corresponds to the same excerpt of music as represented by Fig. B.2, although this corre-

spondence is not visually obvious. Figure B.4 shows a zoomed-in portion of the same signal. The curve here seems to exhibit some periodicity, meaning that the values plotted between samples 20 250 and 20 500 look as though they are repeated approximately between samples 20 500 and 20 750, and samples 20 750 and 21 000, etc. ■

The field known as *signal processing* involves manipulating vectors such as \mathbf{y} in (B.6) and those plotted in Figs. B.1 and B.4 (Mitra, 2010). One well-known topic in signal processing is data compression without loss of audio quality, so that more recordings can be stored in the same amount of computer memory (Painter and Spanias, 2000). The field known as music information retrieval (MIR), which in some cases draws on signal processing, involves extracting pertinent information from audio and other representations of music (Downie, 2003). Some digestion, extraction, summarisation is required, if a signal is to be understood as anything other than a very long vector of numbers between -1 and 1 , with as many as 44 100 numbers to consider per second. The human ear and brain have evolved to digest incoming audio into percepts, and some topics within MIR, such as pitch estimation (de Cheveigné, 2006), can be thought of as computational attempts at audio-percept conversion. In the final part of this section, a definition of *pitch* will be worked towards that makes perceptual sense, that introduces an element of symbolic music representation, and that explains the periodicity evident in Fig. B.4.

Jean le Rond d'Alembert (1717-1783) is generally credited with demonstrating that when a hypothetical string stretched between two end points 0 and L is plucked, its vertical deflection y as a function of time t and horizon-

tal position x obeys a certain equation called the *wave equation* (Kreyszig, 1999, pp. 585-597 for more details). Focusing on the deflection y at a fixed point on the x -axis, one of the components of a function that satisfies the wave equation is

$$f[(t, n, L, c)] = \sin\left(\frac{2\pi nct}{L}\right), \quad (\text{B.7})$$

where $t \in \mathbb{R}_+$ is time, $n \in \mathbb{N}$ is called the *n th harmonic*, $L \in \mathbb{R}_+$ is the length between the end points, and $c \in \mathbb{R}_+$ can be thought of as the *tension* in the string. Definitions of function (Def. A.11) and sigma notation (p. 299) are required in order to fully appreciate (B.7).

Definition B.4. Pitch. Pitch is the perceived attribute of a vibrating string (more generally, body) that can be varied by changing the length L or tension c in (B.7). Decrease the length and/or increase the tension, and there is an increase in the perceived pitch. When one string is plucked, many different *modes of vibration* are in evidence simultaneously, due to the first harmonic $n = 1$, the second harmonic $n = 2$, etc. in (B.7). Although it is possible to train the ear to discern the different harmonics present, what tends to be perceived is unified continuous sound of a certain pitch. ■

Definition B.5. Frequency and octave. The quantity nc/L from (B.7) is called *frequency*, and measured in hertz (Hz or s^{-1}). For $n = 1$, $nc/L = c/L$ is called the *fundamental frequency*. Nominally, a sound that exhibits a fundamental frequency of 440 Hz is said to have pitch A4. A sound with fundamental frequency double this (880 Hz) is said to have pitch A5. A sound with fundamental frequency quadruple this (1760 Hz) is said to have pitch A6, and so on. Similary, a sound with fundamental frequency half this (220 Hz) is said to have pitch A3, etc. Two sounds, one with fundamental

frequency ω Hz and another with fundamental frequency 2ω Hz have many harmonics in common. For instance, the second harmonic of the ω Hz sound has the same frequency as the first harmonic of the 2ω Hz sound. They are perceived as having a certain equivalence—called *octave equivalence*—and their corresponding pitches will bear the same alphabetic labels but different octave numbers. This was the case with A4 and A5 above, where the ‘A’ of A4 or A5 is referred to as *pitch class*, and the 4 and 5 as the *octave number*. ■

Definition B.6. Equal Temperament. Apart from octave relationships, arguably the most closely related pitch to one with fundamental frequency ω Hz is the pitch with fundamental frequency $\frac{3}{2}\omega$ Hz, as the third and second harmonics respectively have the same frequency. The prevalent tuning system called *equal temperament* is a compromise between this relationship and the parsimony of a small number of pitch labels. Pitches in equal temperament have fundamental frequencies that are tuned to $440 \cdot 2^{m/12}$ Hz, where $m \in \mathbb{Z}$ is the number of *semitone intervals* from A4. There are twelve pitches from A4 up to but not including A5, which are labelled using some combination of the seven letters A, B, . . . , G, possibly one of the accidental

symbols ‘ \natural ’, ‘ \sharp ’, ‘ \flat ’, ‘ \times ’, and ‘ $\flat\flat$ ’, and an octave number. For instance,

$$\text{A4 has fundamental frequency } 440 \cdot 2^{0/12} = 440 \text{ Hz}, \quad (\text{B.8})$$

$$\text{A}\sharp 4 \text{ or B}\flat 4 \text{ have fundamental frequency } 440 \cdot 2^{1/12} \approx 466 \text{ Hz}, \quad (\text{B.9})$$

$$\text{B4 has fundamental frequency } 440 \cdot 2^{2/12} \approx 494 \text{ Hz}, \quad (\text{B.10})$$

$$\text{C5 has fundamental frequency } 440 \cdot 2^{3/12} \approx 523 \text{ Hz}, \quad (\text{B.11})$$

$$\text{C}\sharp 5 \text{ or D}\flat 5 \text{ have fundamental frequency } 554 \cdot 2^{3/12} \approx 466 \text{ Hz}, \quad (\text{B.12})$$

$$\text{D5 has fundamental frequency } 440 \cdot 2^{5/12} \approx 587 \text{ Hz}, \quad (\text{B.13})$$

$$\text{D}\sharp 5 \text{ or E}\flat 5 \text{ have fundamental frequency } 554 \cdot 2^{6/12} \approx 622 \text{ Hz}, \quad (\text{B.14})$$

$$\text{E5 has fundamental frequency } 440 \cdot 2^{7/12} \approx 659 \text{ Hz.} \quad (\text{B.15})$$

■

The reason why only seven letters are used to label twelve semitones per octave and why some fundamental frequencies are associated with more than one label (e.g., A \sharp 4 and B \flat 4) is to do with the definition of *scale* (cf. Def. B.8). In the above list of pitches, it is A4 and E4 that bear the close 3/2 relationship, as $\frac{3}{2} \cdot 440 = 660 \approx 659$.

In proper mathematical notation, the naming of a pitch consists of an element of the set of strings $S = \{\text{'A}', \text{'B}', \dots, \text{'G}'\}$, an element of the set of strings $T = \{\text{''}, \text{'\natural'}, \text{'\sharp'}, \text{'\flat'}, \text{'\times'}, \text{'\flat\flat'}\}$, and an integer octave number. The set of pitch classes is denoted $U = \{\text{conc}(s, t) : s \in S, t \in T\}$, and \mathbb{Z} is the set of octave numbers (cf. Def. A.6). A shorthand such as C \sharp 5 has and will continue to be used instead of the proper mathematical notation, in this case

(‘C♯’, 5) ∈ (U × ℤ).

Example B.7. Some harmonics for the pitch F♯3 with fundamental frequency $440 \cdot 2^{-15/12} \approx 185$ Hz are shown separately in Fig. B.5A, and superposed (as they deflect the string at a point over time) in Fig. B.5B. The strength of the different harmonics in the superposition is given by a vector $\mathbf{a} = (a_1, a_2, \dots) \in [0, 1]^\infty$ called the *amplitudes*, so that the general form of the function plotted in Fig. B.5B is given by

$$g [(t, n, \omega, \mathbf{a})] = \sum_{n=1}^{\infty} a_n \sin (2\pi\omega nt). \quad (\text{B.16})$$

In the particular case of Fig. B.5B, $\omega \approx 185$ Hz and

$$\mathbf{a} \approx (4.8 \cdot 10^{-2}, 7.5 \cdot 10^{-3}, 0, 1.9 \cdot 10^{-3}, 0, 0, 0, 0, 0, 0, 6.1 \cdot 10^{-2}, 0, 0, \dots). \quad (\text{B.17})$$

The remarkable method by which a signal (e.g., Fig. B.4) can be approximated using superposed sine curves (such as in Fig. B.5B) is named *Fourier analysis*, after Jean Baptiste Joseph Fourier (1768-1830). When the approximation is close, the frequencies of the superposed harmonics can be used to estimate what pitch(es) the signal contains. While this introductory discussion of audio, signal processing, and pitch has overlooked some important aspects of a signal, such as phase and sound pressure level (Rumsey and McCormick, 2009), the reader now has working definitions of the percept of pitch, as well as the underlying mathematics. ■

FIGURE
REMOVED
FROM
ELECTRONIC
VERSION
FOR
COPYRIGHT
REASONS

Figure B.5: (A) Separate harmonics for the pitch F♯3 with fundamental frequency $440 \cdot 2^{-15/12} \approx 185$ Hz; (B) Superposed harmonics for the pitch F♯3 with fundamental frequency $440 \cdot 2^{-15/12} \approx 185$ Hz are shown in blue. Also plotted is the portion of the audio signal from Fig. B.4 ('To the end' by Blur, 1994). The blue signal appears to be a good approximation to the black signal. © Copyright 1994 by EMI Music Publishing. Used by permission.

B.2 Symbolic representation of music

B.2.1 Staff notation

A *musical score* (or just score) is any attempt to represent a piece of music graphically, of which *staff notation* is one instance. For a particular piece (or excerpt) of music, the elements of staff notation are also referred to as the *musical surface* (Lerdahl and Jackendoff, 1983).

Compared with audio, staff notation achieves higher structural generality as a music representation, but is less expressively complete (Wiggins, Miranda, Smaill, and Harris, 1993). Staff notation developed as a succession of instructions to be read from left to right, indicating which pitches should be sung when, to what words, and for how long. The earliest surviving examples of staff notation would not have been used by singers to learn or perform music; rather the luxurious manuscripts served to document compositions, and some appear to have been taken from place to place as a means of conveying the music of one school to another. The excerpt shown in Fig. B.6A is a facsimile from a manuscript known as F (Dittmer, 1966-7). The two-part excerpt would have been sung as part of the liturgy at the Cathedral of Notre Dame, between about 1150 and 1250 (Wright, 1989). This is an example of some of the earliest polyphony (music where more than one pitch is sung/played simultaneously) to be written down, although much earlier sources of monophony (music where only one pitch is sung/played at a time) exist. An attempt by Baltzer (1995) to transcribe the excerpt from Fig. B.6A into modern notation is shown in Fig. B.6B.

Staff notation is an evolving representation of music. Stone (1980) gives a

**FIGURE
REMOVED
FROM
ELECTRONIC
VERSION
FOR
COPYRIGHT
REASONS**

Figure B.6: (A) A facsimile of 'Regnat' from the manuscript F (166v V, Dittmer, 1966-7). This is an example of some of the earliest surviving polyphony to be written down; (B) A transcription of 'Regnat' from Fig. B.6A into modern staff notation (Baltzer, 1995). © Copyright 1995 by Éditions de l'Oiseau-Lyre. Reproduced by permission.

wide range of examples from the twentieth century, some of which bear little if any resemblance to Fig. B.6A. One such example by Sylvano Bussotti (*b* 1931) is reproduced in Fig. B.7. Having asserted that conventions in notation are transient, and shown both an early and recent example, an excerpt from the nineteenth century—from ‘L’invitation au voyage’ (1870) by Duparc—is given in Fig. B.8, and will be used to discuss the elements of staff notation in more detail.

FIGURE REMOVED

FROM ELECTRONIC VERSION
FOR COPYRIGHT REASONS

Figure B.7: The first movement from *Five pieces for David Tudor* no.4 (piano, 1959) by Bussotti. © Copyright 1959 by Casa Ricordi.

B.2.2 The elements of staff notation

1. *Staff*. The staff consists of parallel lines (usually five). Staves are grouped together to form systems. For instance, there are three staves

[Presque lent] (2)

50 *più f*

S. (4) C'est pouras - sou - vir, _____ Ton moin - dre dé - sir Qu'ils vien -

(3) (5) (6) (10) (11) (10) (10)

Pno. *più f* Expressif (10) (10) (10)

54 (9) - nent du bout (9) du mon - - -

f (7, 8) (10)

57 Un peu plus vite (2) *mf*

de (3) Les so -

Figure B.8: Bars 50-58 of 'L'invitation au voyage' (1870) by Henri Duparc. Annotations are shown in red, and correspond to the numbered list in Sec. B.2.2.

per system in Fig. B.8, and once a system has been played (or read) from left to right, attention switches to the system on the next line down. Typically, a melody instrument (such as the voice, violin) is represented by one staff, and an instrument where two hands are pressing keys or plucking strings (such as the piano, harp) is represented by two staves. The excerpt in Fig. B.8 is scored for soprano (S.) and piano (Pno.).

2. *Tempo* is an indication of how the piece ought to be played. Although often accompanied by an instruction in beats per minute, Cooper and Meyer (1960) point out that ‘two pieces of music may move at the same absolute speed, but one of the pieces may seem faster than the other. This is possible because the psychological tempo, which we shall call ‘pace’, depends upon how time is filled’ (p. 3). The tempo in Fig. B.8 is inherited from the beginning of the piece, hence the editorial brackets. There is a tempo change towards the end of this excerpt.
3. *Clef*. At the beginning of each staff a clef indicates the height of a reference pitch. For instance, the soprano part in Fig. B.8 contains a treble clef. The starting point of the treble clef, indicated by the arrow, marks the position of pitch G4. The piano’s right-hand part also contains a treble clef. The left-hand part contains a bass clef, whose starting point, indicated by the arrow, marks the position of pitch F3. Clefs are repeated on subsequent systems, and can change during a system, as happens towards the end of this excerpt.
4. *Key signature*. The key signature follows the clef, and is a succession

of sharp ('♯'), flat ('♭'), or natural ('♮') signs indicating that each pitch written on this line or in this space will be played as a sharp, flat, or natural respectively, unless altered by local accidental signs. Three flats indicate that the excerpt in Fig. B.8 is in the key of E♭ major or C minor. It is quite difficult to determine which from this short excerpt, but overall the song is in C minor. Key signatures are repeated on subsequent systems, and can change during a system, as indicated by the three natural signs towards the end of this excerpt. The concept of key is revisited in Sec. 2.4.

5. *Time signature.* The two numbers appearing to the right of the key signature are referred to collectively as the time signature. The top number (six in Fig. B.8) is the number of beats of music that will be represented before a barline (vertical line) marks the end of the current bar and the beginning of the next. As such, the term *bar* refers to all the information on and immediately above and below the staff that occurs between barlines. Bars are numbered for ease of reference, with the excerpt in Fig. B.8 consisting of bars 50-58. The bottom number of the time signature (in this case eight) refers to the type of note in which beats are counted. The number one corresponds to a semibreve, two corresponds to a minim, four to a crotchet, eight to a quaver, etc. The next item in this list gives the relative durations of these notes.
6. *Note.* One note is highlighted in the left-hand of the piano in bar 50 of Fig. B.8. A note consists of a notehead and possibly a dot to the right (in this case there is a dot to the right of the black ellipse). A note may also consist of a stem and a tail (in this case just a stem),

and an accidental symbol appearing to the left of the notehead (not in this case). Various other elements of notation (see items 9-12 below) affect how the note is performed.

A note has an *ontime*—a theoretical point in time when the note is meant to be played.¹ Staff notation is *relative*, meaning the *ontime* of a note has to be inferred from the *ontimes* of previous notes. Counting in quavers from zero, the highlighted note in bar 50 of Fig. B.8 has *ontime* 0. To determine the *ontime* of the next note in the left hand, durations need to be introduced.

A note has a *duration*—a theoretical amount of time for which a note is meant to be sustained. A semiquaver is worth half of a quaver, a quaver worth half of a crotchet, a crotchet worth half of a minim, and a minim worth half of a semibreve. Semiquavers, quavers, and crotchets have black ellipses for noteheads, whereas minims and semibreves have empty ellipses. If a dot appears to the right of a notehead, the note's duration is increased by half. For instance, ordinarily a crotchet is worth two quavers, but the highlighted note in bar 50 of Fig. B.8 is a dotted crotchet, so has duration three quavers. This means, still counting *ontimes* in quavers from zero, that the *ontime* of the next note in the left hand is 3. This note itself is a crotchet, lasting two quavers, meaning that the final note in the left hand of bar 50 (a quaver) has *ontime* 5. This final note, a quaver, has one tail emerging from its stem. A semiquaver has two tails emerging from its stem. The right hand of the piano part is full of semiquavers, but rather than being

¹The term *ontime* is used as distinct from *onset*, which is a percept.

written with separate tails, the tails are joined (or beamed) for ease of reading. A dotted minim appears in bar 56 of the soprano part, but Fig. B.8 contains no semibreves, which look like minims without stems. If longer durations cannot be notated adequately using some combination of the above notes and dotting, then two or more notes are tied together: the note is played once and held for a duration equal to the sum of its constituent notes' durations. Ties appear in bars 54-56 of the piano left hand, and should not be confused with slurs (see item 11 below). If shorter durations cannot be notated adequately, a *tuplet* may be used. For instance, a 9:6 tuplet appears in bar 58, meaning that the nine sixteenth notes beamed together must be performed in the time ordinarily occupied by six sixteenth notes. Sometimes a tuplet appears without the second number of the ratio, in which case it is assumed to be a two. The duration of a note, like its ontime, is relative to the tempo indication: a crotchet in a piece played slowly has a longer duration than one in a piece played fast.

The pitch of a note is given by its relative height on the staff, and may be modified by accidental signs. The pitches of various notes are given in Fig. B.9, as well as two integer representations of pitch that will be revisited in Sec. 2.2. From Fig. B.9 it can be seen that each line and space on the staff is associated with a particular element of $S = \{‘A’, ‘B’, \dots, ‘G’\}$, and that these are recycled every seven steps. An accidental from the set of strings $T = \{‘\flat’, ‘\sharp’, ‘\flat\flat’, ‘\sharp\sharp’, ‘\flat\flat\flat’, ‘\sharp\sharp\sharp’\}$ may be placed to the left of a notehead, and modifies one of the integer representations (MNN) in Fig. B.9 but not the other (MPN). Returning

to Fig. B.8, local accidentals (those placed next to a notehead rather than in a key signature) apply for the remainder of the bar in which they occur. Ledger lines are short extra staff lines that indicate the pitch of particularly high or low notes. For example, a ledger line is used to notate the soprano's A \flat 5 in bar 51 of Fig. B.8.

7. *Rest.* As staff notation is relative, silent durations must be notated explicitly, as opposed to the staff being left empty. Otherwise it is assumed that the next written note begins immediately. Each of the main note durations (semiquaver, quaver, crotchet, minim, semibreve) has its equivalent rest symbol, and, like notes, rests may be dotted to increase their duration by half. A semiquaver rest is annotated in bar 54 of Fig. B.8, a quaver rest is annotated in bar 57 and is followed by a dotted crotchet rest. Minim rests are small black rectangles that appear on a staff line, and semibreve rests look the same but appear to hang from a stave line. The symbol for a semibreve rest can also be used to indicate a bar-long rest, even if the length of the bar is different to a semibreve.
8. *Voicing.* Notes that are meant to be sung or played simultaneously (and rests that are meant to be observed simultaneously) should appear in vertical alignment. Some simultaneous notes share stems (such as in bars 50-53 of the piano right hand in Fig. B.8) whereas others have their own stems. In bars 54-57 of the right hand, some notes share stems and some have their own. The two groups of notes in these bars, one with upward stems and the other with downward stems, are referred to as two voices. Whether or not a listener would hear two

independent voices in an audio version of this piece depends on several perceptual factors (Bregman, 1990; Cambouropoulos, 2008). It is possible for additional voices to appear midway through an excerpt of staff notation, and for existing voices to disappear. A voice may also cross from one staff to another for instruments like the piano or harp. For instance, in the piano left hand of bar 54 of Fig. B.8, an additional voice appears, indicated by the semiquaver rest. The voice then crosses to the right-hand staff, where a D4 is notated. Cross-staff notes appear up to and including bar 58 of the piano part.

9. *Lyrics* are written below the note of a vocal part to which they should be sung. The syllables of multisyllabic words are joined by hyphens, and the continuation of a single syllable for multiple notes is also indicated by a hyphen, or an underscore if it is the final syllable.
10. *Dynamics*. Relative levels of strength or loudness in a piece of music are indicated in staff notation by dynamics. For example, ***pp*** is short for *pianissimo* (very soft), ***p*** is short for *piano* (soft), ***mp*** for *mezzo piano* (medium softness), ***mf*** for *mezzo forte* (medium strength or loudness), ***f*** for *forte* (strong), and ***ff*** for *fortissimo* (very loud). A transition between one dynamic level to a stronger level is indicated by the word *crescendo* (*cresc.*) or an opening hairpin (as annotated in bar 53 of Fig. B.8). A corresponding decrease in dynamic level is indicated by *decrescendo* (*decresc.*), or *diminuendo* (*dim.*), or a closing hairpin (annotated in bar 56 of Fig. B.8). Alterations in tempo (*accelerando*, *ritardando*, *rubato*) also come under the heading of dynamics. Expression marks, such as *expressif* in bar 50 of Fig. B.8 tend to be more open

to interpretation. Duparc instructs the pianist to play this excerpt expressively, but two different performers are unlikely to interpret this instruction in the same way.

11. *Articulation and phrasing.* The various elements of staff notation already introduced give clear instructions for what should be sung/played when, until when, and with what strength or relative loudness. Articulation and phrasing give further details of *how* notes should be sung/played. A slur is a curved line ‘extending over or under a succession of notes to indicate their grouping as a coherent unit, for example in legato performance [where note durations are extended slightly but sometimes are not, creating a bound or joined-up effect], or for purposes of phrasing’ (Chew, 2001, p. 526). A slur is annotated in the piano left hand of bar 52 of Fig. B.8. There are also articulation marks—lines, dots, hats, etc.—that are written above or below individual notes, but I am not concerned with these here. Different instruments have their own idiosyncratic articulation marks, such as bowing directions for the family of string instruments.
12. *Ornamentation.* An ornament is a symbol written above, below, or to the side of one or more notes, indicating that certain additional notes should be played at this point, or that the written notes should be played in a certain way. For instance, the letters *tr*, standing for trill, indicate that the written note should be played in rapid alteration with one above or below for a period of time. As another example, a group of vertically-aligned notes prefaced by a coiled line should be arpeggiated (played rapidly one after the other), rather than played exactly simul-

taneously. Ornamentation varies considerably with historical period and geographical location.

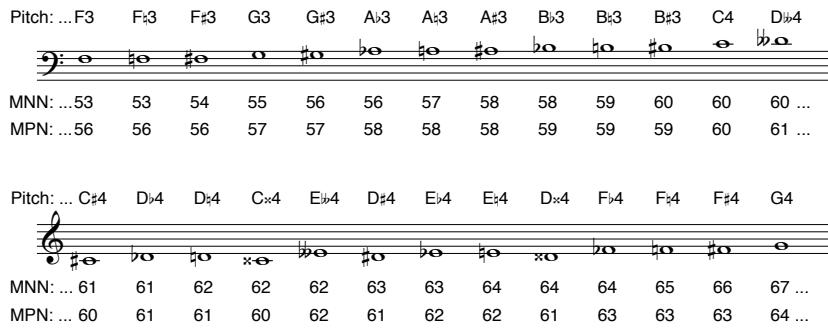


Figure B.9: A collection of notes annotated above with their pitch names, and below with two integer representations. The first integer representation is called MIDI note number (MNN), and discussed in Sec. 2.1. The second integer representation is defined by Meredith (2006a) as morphetic pitch number (MPN).

According to Fallows (2001), ‘[t]empo and expression marks may be the most consistently ignored components of a musical score... partly because only the notes are objective facts, but also because musicians tend to look first at the music [notes], only later checking the markings to see whether they agree with initial impressions’ (p. 271). Arguably, this observation might be applied to lyrics, to other aspects of dynamics, and to articulation, phrasing, and ornamentation. In an attempt to refine understanding, some researchers have been drawn to the more ambiguous elements of staff notation and their interpretation, such as *rubato* (Spiro, Gold, and Rink, 2008). On the other hand, it can be beneficial to focus on the minimal amount of information necessary for a listener who is familiar with a piece to recognise it as such. Knowing the ontime, pitch, duration, and staff of each note, for example, is considered an adequate starting point in Chapters 3-9.

B.2.3 MusicXML and kern

While the graphics file shown in Fig. B.8 is an appropriate representation of an excerpt of music for the purposes of performance and discussion, it is not straightforward for a notation program (such as Noteflight, Sibelius, or Finale), a recording/performance program (e.g., a sequencer, such as Logic or Cubase), an education program, or a database program to convert this graphics file into anything useful. Programs for extracting the elements of staff notation from a graphics file are an area of research in their own right (Rebelo, Capela, and Cardoso, 2010). Further, if the aim is to transfer a file representing a piece of music from one program to another, conversion to or from a graphics file is unnecessary. MusicXML (Extensible Markup Language) and kern are text-based, computer-readable data formats for representing one or more pieces of music (Recordare LLC, 2010; Huron, 2002, 1999).

Here is a musicXML representation for some of the excerpt shown in Fig. B.8. A gap in the code is indicated by an ellipsis, ‘...’.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE score-partwise PUBLIC "-//Recordare//DTD ...
3   "http://www.musicxml.org/dtds/partwise.dtd">
4 <score-partwise version="2.0">
5   <work>
6     <work-title>L'invitation au voyage</work-title>
7   </work>
8   <identification>
9     <creator type="composer">Henri Duparc</creator>
```

```
10    </identification>
11
12    ...
13
14    <part id="P1">
15        <measure number="50">
16            <attributes>
17                <divisions>24</divisions>
18                <key>
19                    <fifths>-3</fifths>
20                    <mode>minor</mode>
21                </key>
22                <time>
23                    <beats>6</beats>
24                    <beat-type>8</beat-type>
25                </time>
26                <clef number="1">
27                    <sign>G</sign>
28                    <line>2</line>
29                </clef>
30            </attributes>
31            <note>
32                <pitch>
33                    <step>C</step>
34                    <octave>5</octave>
```

```
35      </pitch>
36
37      <duration>24</duration>
38
39      <voice>1</voice>
40
41      <type>quarter</type>
42
43      <notehead>normal</notehead>
44
45      <lyric number="1">
46
47          <syllabic>single</syllabic>
48
49          <text>C'est</text>
50
51          <color>#000000</color>
52
53      </lyric>
54
55      </note>
56
57      <note>
58
59          <pitch>
```

35 `</pitch>

36

37 `<duration>24</duration>

38

39 `<voice>1</voice>

40

41 `<type>quarter</type>

42

43 `<notehead>normal</notehead>

44

45 `<lyric number="1">

46

47 ` `<syllabic>single</syllabic>

48

49 ` `<text>C'est</text>

50

51 ` `<color>#000000</color>

52

53 `</lyric>

54

55 `</note>

56

57 `<note>

58

59 ` `<pitch>

```
60      </note>

61      <note>
62          <pitch>
63              <step>E</step>
64              <alter>-1</alter>
65              <octave>5</octave>
66          </pitch>
67          <duration>24</duration>
68          <voice>1</voice>
69          <type>quarter</type>
70          <notehead>normal</notehead>
71          <lyric number="1">
72              <syllabic>begin</syllabic>
73              <text>as</text>
74              <color>#000000</color>
75          </lyric>
76      </note>

77      ...
78      </measure>

79      ...
80  </part>
81  <part id="P2">
82      ...
83  </part>
84  </score-partwise>
```

It is not vital to provide an exhaustive account of musicXML or kern for the purposes of subsequent chapters, but some general remarks will be made, and it is an interesting exercise to try to relate what appears in the above code to Fig. B.8. Lines 1-10 of the above code are preamble for document type and bibliographic information. Line 4 states that the music is encoded partwise, as opposed to timewise. In a partwise musicXML file, the entire top part is encoded (in this case, the soprano), then the entire second part (in this case, the piano), etc. Reading such code, it can be difficult to tell which notes are simultaneous. It is also possible to produce timewise encodings in musicXML, in which it is easier to tell which notes are simultaneous. Functions that convert from one encoding to the other are available. Ordinarily, lines 11-13 would contain a complete list of the parts to be encoded, and then the encoding of the first part (soprano) begins on line 14. Lines 16-30 contain information that relates to the time signature, key signature, and clef. The first actual note of the soprano part is specified by lines 31-44. One can imagine writing a program to extract certain elements of notation from this musicXML file. For instance, representing the code as a list of strings $L_1 = (\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_{84})$, a function f might take L_1 as its argument, and output the pitches specified for the first part, giving another list $L_2 = (\text{C}5, \text{C}5, \text{E}\flat\text{5})$.

A kern representation for some of the excerpt shown in Fig. B.8 is given below. Again, a gap in the code is indicated by an ellipsis, ‘...’.

```
1 !!!COM: Duparc, Henri
2 !!!OTL: L'invitation au voyage
3 !!!ODT: 1870//...
4 !!!OMD: Presque lent
```

```

5  **kern  -***kern  -***dynam  -***kern  -***dynam  -***text
6  *staff3  -*staff2  -*staff2/3  -*staff1  -*staff1  -*staff1
7  *Ipiano  -*Ipiano  -*Ipiano  -*Ivox  -*Ivox  -*Ivox
8  *clefF4  -*clefG2  -*clefG2  -*clefG2  -*clefG2  -**
9  *k[b-e-a-]  -*k[b-e-a-]  -*...  -*k[b-e-a-]  -**
10 *c:  -*c:  -*c:  -*c:  -*c:  -*c:
11 *M6/8  -*M6/8  -*M6/8  -*M6/8  -*M6/8  -**
12 =50  -*=50  -*=50  -*=50  -*=50  -*=50
13 (4.G\  -*16cc/ (16aa/LL  -*piu f  -*4cc\  -*piu f  -*C'est
14 .  -*16g/ 16ee-/  -*.  -*.  -*.  -*.
15 .  -*16cc/ 16aa/  -*.  -*.  -*.  -*.
16 .  -*16g/ 16ee-/  -*.  -*.  -*.  -*.
17 .  -*16cc/ 16aa/  -*.  -*8cc\  -*.  -*pour
18 .  -*16g/ 16ee-/JJ  -*.  -*.  -*.  -*.
19 4c\  -*16cc/ 16aa/LL  -*.  -*4ee-\  -*.  -*as-
20 .  -*16g/ 16ee-/  -*.  -*.  -*.  -*.
21 .  -*16cc/ 16aa/  -*.  -*.  -*.  -*.
22 .  -*16g/ 16ee-/  -*.  -*.  -*.  -*.
23 8e-\  -*16cc/ 16aa/  -*.  -*8cc\  -*.  -*sou-
24 .  -*16g/ 16ee-)/JJ  -*.  -*.  -*.  -*.
25 =51  -*=51  -*=51  -*=51  -*=51  -*=51
26 ...
27 =59  -*=59  -*=59  -*=59  -*=59  -*=59
28 *_  -*_*  -*_*  -*_*  -*_*  -*_
```

Lines 1-4 of the above code contain bibliographic information. Kern files

are tab delimited, which is evident from line 5 onwards. Column one of the above code encodes the piano left hand, column two the right hand, column three the dynamic markings for the piano, column four encodes the soprano staff, column five the dynamic markings for the soprano, and column six the lyrics. As with musicXML, kern can manage additional voices appearing and/or existing voices disappearing midway through an excerpt. Lines 8-11 contain information that relates to the clef, key signature, and time signature. Reading down the first column of lines 13-23, the pitches G3 (line 13), C4 (line 19), and E \flat 4 (line 23) of the piano left hand are evident, although kern is using a different pitch nomenclature with lowercase and uppercase letters. Reading across the columns of line 13, the pitches of the first simultaneity of bar 50 of Fig. B.8 are evident, as well as some dynamic markings and a lyric, ‘C'est’. The extra symbols attached to pitch letters—numbers and dots, parentheses, slashes, etc.—refer to durations, slurs, beams etc. respectively. Kern is a terser representation than musicXML, and the data in a kern file can be read partwise or timewise without recourse to a conversion function. Again one can imagine writing a program to extract certain elements of notation from this kern file.

When designing a format for music representation, it is important to consider the compromise between ease of use for the greatest number of users and catering for highly customised scores (such as in Fig. B.7). Using musicXML or kern, it might be difficult to take control of the position of each element on or around the staff, such as the *expressif* direction in bar 50 of Fig. B.8, but how much does this matter? Arguably, a scenario where one is able to conduct research by selecting from thousands of pieces in musicXML and/or

kern formats, containing the occasional mistake or inaccuracy, is preferable to a scenario where one is able to choose from only a handful of very accurately represented pieces.

B.2.4 An object-oriented approach

The notation programs Sibelius and Noteflight use object-oriented approaches for music representation (Budd, 1991). SmOKE (Pope, 1992) is another example of an object-oriented music representation. In a Sibelius file, ‘Score’ is an *object class*, of which ‘Staff’ is a *subclass*, of which ‘Bar’ is a subclass, . . . , of which ‘Note’ is a subclass (Finn et al., 2009). Each subclass has relevant *methods* (or functions) and *variables* associated with it. For instance, one of the variables of ‘Note’ is ‘Tied’, which takes the value ‘True’ if there is a tie between this note and the next, and ‘False’ otherwise. One of the methods of ‘Note’ is ‘Transpose’, which enables the user to change the pitch of the note (and perhaps other notes all at once) by a specified amount.

In Sibelius, a user can create music in staff notation by dragging and dropping notes on to a graphic containing the staves, so does not see or edit a text file such as those shown above. The combination of an object-oriented approach with a graphical interface results in ease of use and flexibility. For example, if the user drags the latter notehead of a pair of tied notes to the right to give it more space, the tie itself is elongated and the spacing of other elements is adjusted, automatically. High-end notation software has many impressive features, such as the ability to synthesise audio files from symbolic representation, and to transcribe a played melody into a symbolic representation. Most notation software allows users to export files as musicXML.

B.3 Primary concepts of music theory

Figure B.10 demonstrates the primary concepts of *scale*, *cycle of fifths*, and *chord*.

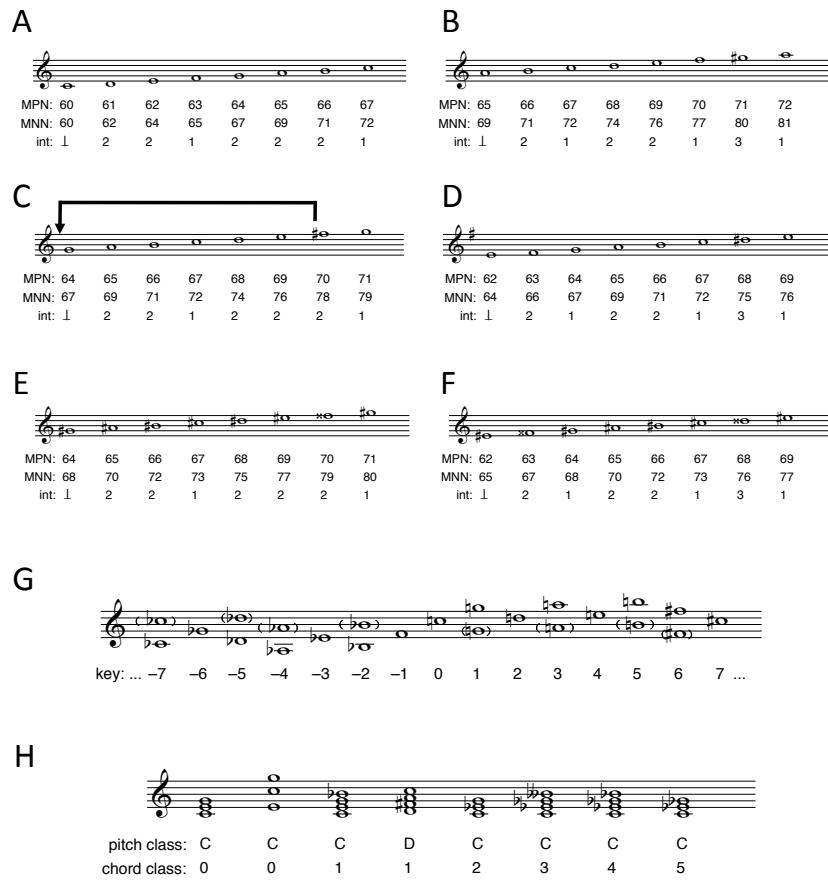


Figure B.10: The primary music-analytic concepts of scale, cycle of fifths, and chord. (A) C major scale; (B) A harmonic minor scale; (C) G major scale, written without key signature; (D) E harmonic minor scale, written with key signature; (E) G \sharp major scale; (F) E \sharp harmonic minor scale; (G) Cycle of fifths, with distance from pitch class C on the cycle indicated by integers underneath the staff; (H) Various classes of chords, with more details in Def. B.10.

Definition B.8. Scale and key. A scale is a list of pitches, with the pitches being regularly spaced according to either MNN or MPN. Figure

B.10A contains a major scale beginning on C4. It is called the C major scale, as it does not matter on which C it begins. Morphetic pitch numbers (MPN) of notes are written below the staff, from which it can be seen that they are regularly spaced. The MIDI note numbers (MNN) are not regularly spaced, however, and it is the list of semitone intervals (2, 2, 1, 2, 2, 2, 1) that identifies a major scale. A scale that shares all but one pitch class in common with the C major scale is the A harmonic minor scale, shown in Fig. B.10B. Comparing Figs. B.10A and B, only the G♯ is different. As with major scales, harmonic minor scales are regularly spaced according to MPN. The list of semitone intervals (2, 1, 2, 2, 1, 3, 1) identifies a harmonic minor scale. The A harmonic minor scale is said to be the relative minor of the C major scale, as the scales are so similar.

Two more pairs of major and harmonic minor scales appear in Figs. B.10C and B.10D, and Figs. B.10E and B.10F. Figure B.10C contains the G major scale. Its penultimate note has to be sharpened in order to adhere to the semitone intervals (2, 2, 1, 2, 2, 2, 1). The sharp symbol of F♯ could have been written in the key signature, as indicated by the arrow. In Fig. B.10D the relative minor scale, E harmonic minor, is written with a key signature. A piece or excerpt of music that uses mainly pitch classes from the G major or E harmonic minor scales is said to be in the key of G major or E minor. If the first or last pitch classes of the piece emphasise G rather than E, then it is probably in G major, and vice versa.

The strict observation of the scale definition leads to double sharps and flats: if a major scale begins on G♯, is spaced regularly according to MPN, and observes the semitone intervals (2, 2, 1, 2, 2, 2, 1), then the penultimate pitch

class will be $F\ddot{x}$. Similarly for double flats. Theoretically, triple, quadruple, etc. flats and sharps are possible. Some sharpened and flattened pitches are more commonly known by natural *enharmonic equivalents*. For instance, $C\flat 4$ and $C\flat 5$ in Fig. B.10G might be written as $B3$ and $B4$ instead. ■

Definition B.9. Cycle of fifths. It was mentioned (cf. Def. B.6) that next to octave equivalence, the pitch that is considered most equivalent to one with fundamental frequency ω Hz is the pitch with fundamental frequency $\approx \frac{3}{2}\omega$ Hz. Beginning on $C5$, this pitch is $G5$. Beginning on $G5$, this pitch is $D6$, etc. In the other direction, beginning on $F4$, this pitch is $C5$. Beginning on $B\flat 3$, this pitch is $F4$, and so on. A so-called cycle of pitch classes is thus established, and shown in Fig. B.10G. For historical reasons, it is called the cycle of fifths. The distance from pitch class C on the cycle of fifths is indicated by integers underneath the staff. An integer below a pitch class indicates the number of sharps (positive integers) or flats (negative integers) that ought to appear in the key signature of a piece in that major key. This is why the viewpoint list for *key* in Fig. 2.2 takes the value -3 repeatedly: this excerpt has three flats in the key signature, so overall it is in the key of $E\flat$ major or the relative minor, C minor. ■

Definition B.10. Chord. Pitches that sound either simultaneously or in succession can be collected together, and written on top of one another, as shown in Fig. B.10H. Such collections of two or more pitches are called chords. As with scales, the identity of a chord is determined by the semitone intervals between adjacent pitches. Even if the pitches are exchanged for (or supplemented with) octave-equivalent pitches, as demonstrated by the first two chords in Fig. B.10H, the identity of the chord is unchanged. There are

many classes of chord, so Fig. B.10H is limited to the six classes given by Pardo and Birmingham (2002):

0. *Major triad* is the name for a chord whose pitches can be arranged to give the semitone intervals (3, 4). The first two chords in Fig. B.10H are C major triads, as pitch-class C is the *root*—the lowest pitch in the chord when rearrangement gives the semitone intervals (3, 4).
1. *Dominant 7th*. This chord class is identified by the semitone intervals (4, 3, 3). The second and third chords in Fig. B.10H are C and D dominant 7th respectively.
2. *Minor triad*. This chord class is identified by the semitone intervals (3, 4). The fourth chord in Fig. B.10H is a C minor triad.
3. *Fully diminished 7th*. This chord class is identified by the semitone intervals (3, 3, 3). The fifth chord in Fig. B.10H is C fully diminished 7th.
4. *Half diminished 7th*. This chord class is identified by the semitone intervals (3, 3, 4). The penultimate chord in Fig. B.10H is C half diminished 7th.
5. *Diminished*. This chord class is identified by the semitone intervals (3, 3). The last chord in Fig. B.10H is C diminished. ■

Definition B.11. Scale degree and roman numeral notation. The first note of a major or minor scale is known as the *tonic*, or *first degree*. The second note is the *supertonic*, or *second degree*, followed by the *mediant*, *subdominant*, *dominant*, *submediant*, and *leading note*. The *tonic triad* is a

triad with the tonic note as its root. The *supertonic triad* is a triad with the supertonic as its root, etc., as shown in Fig. B.11. With the exception of the mediant triad in the harmonic minor scale, the remaining chord notes are members of the appropriate scale.

Roman numeral notation is a shorthand for referring to triads built on different scale degrees, with uppercase for major triads, lowercase for minor triads, and lowercase with a superscript circle for diminished triads, as shown in Fig. B.11. ■

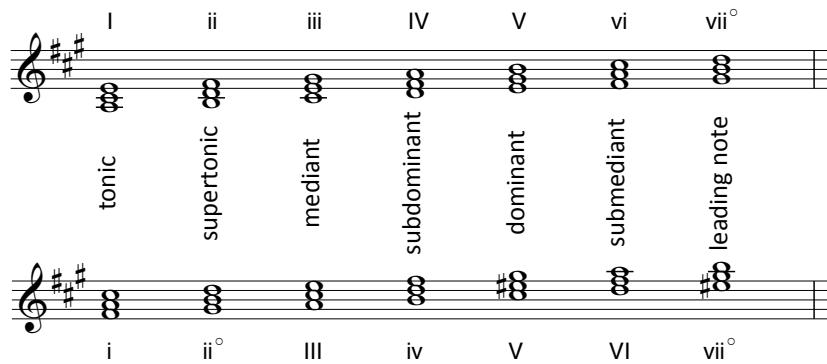


Figure B.11: Names for different degrees of the major and minor scales, and Roman numeral notation.

As an example, the progression E minor triad, G major triad, A minor triad, E minor triad, in the key of E minor, sounds equivalent to the progression A minor triad, C major triad, D minor triad, A minor triad, in the key of A minor, so it makes sense to refer to both concisely as i-III-iv-i. When a passage of music is labelled with Roman numerals, there may be some ambiguities due to non-chord notes. That said, certain chord progressions are regularly observed, especially at phrase endings. These progressions are given special names, for instance IV-I is a *plagal cadence*. Chord progressions

that establish certain keys, perhaps by using regular progressions, may be perceived as having *tonal direction*, whereas less common chord progressions may not.

Once a concept is comprehended, it becomes a potential tool for the analysis of music. For instance, the second half of bar 56 in Fig. B.8 consists of a D dominant 7th chord, and bar 57 consists of a G major triad. Different tools are more or less suited to the analysis of music from different genres/periods, and very rarely does a piece remain entirely within primary conceptual bounds. For example, I mentioned the second half of bar 56 in Fig. B.8, as the first half of bar 56 does not correspond to any of the chord classes discussed so far. The determination of key presents a similar issue: I hear bars 53-57 of Fig. B.8 as being in G major, but only retrospectively, given the resounding G major triad in bar 57. Bars 50-52 and 58 are more ambiguous with respect to key. A piece/excerpt that either disregards or is ambivalent toward conceptual bounds, gives analysts ample essay-writing material.

C

Definitions of the explanatory variables included in the regressions

An intuitive definition is given for each variable unless one is given in an earlier chapter. A mathematical definition is then given where appropriate. I adopt the notation from Meredith et al. (2002). That is, a *dataset* D (containing *datapoints* or just *points*) is a representation of an excerpt of music. A *translational pattern* P_1 is a subset of D , for which there exist $m - 1$ translations in D , written P_2, P_3, \dots, P_m . Meredith et al. (2002) call $\{P_1, P_2, \dots, P_m\}$ the *translational equivalence class* of P_1 in D , or $TEC(P_1, D)$ for short.

Cardinality is the number of notes contained in one occurrence of a pattern.

I write $|P|$ for the cardinality of the pattern, or sometimes l .

Occurrences refers to the number of times that a pattern occurs in an excerpt. This will be denoted by $|TEC(P, D)|$ or simply m .

Mean-centered cardinality \times occurrences: The intuition behind this variable is that a pattern containing many notes and/or occurring many times is likely to be relatively noticeable/important. It is common practice to use mean-centered values when forming such interactions, so if $\overline{|P|}$ is the mean of the cardinalities of all discovered patterns and

\bar{m} is the mean of the occurrences of all discovered patterns, then the *mean-centered cardinality* \times *occurrences* for a pattern P in a dataset D is given by

$$\text{mc_cardinality} \times \text{occurrences}(P, D) = (|P| - \overline{|P|})(m - \bar{m}). \quad (\text{C.1})$$

Coverage: The *coverage* of a pattern in a dataset is ‘the number of data-points in the dataset that are members of occurrences of the pattern’ (Meredith et al., 2003, p. 7).

$$\text{coverage}(P, D) = \left| \bigcup_{Q \in TEC(P, D)} Q \right|. \quad (\text{C.2})$$

The variables *coverage* and *mean-centered cardinality* \times *occurrences* are likely to be highly positively correlated as, before mean-centering, the latter is an upper bound for the former.

Compactness: The mathematical definition is

$$\text{compactness}(P, D) = l / |\{\mathbf{d}_i \in D : \mathbf{p}_1 \preceq \mathbf{d}_i \preceq \mathbf{p}_l\}|. \quad (\text{C.3})$$

A second definition is used for patterns that appear in one staff only. If the pattern P occurs in the top staff only, say, then D in (7.3) should be replaced by D^\dagger , the set of all datapoints occurring in the top stave.

Compression ratio: The mathematical definition is

$$\text{compression_ratio}(P, D) = \frac{\text{coverage}(P, D)}{|P| + |TEC(P, D)| - 1}. \quad (\text{C.4})$$

ThreeCs: The variables *coverage*, *compactness* and *compression ratio* are combined by Meredith et al. (2003) so as to order discovered patterns, but the nature of this combination is not made explicit. Meredith kindly confirmed by personal correspondence that the combination takes the form

$$\text{coverage}^a \cdot \text{compactness}^b \cdot \text{compression_ratio}^c, \quad (\text{C.5})$$

where a , b , and c are parameters to be chosen by the user. Stipulating these parameters a priori is not always appropriate, so Forth and Wiggins (2009) offer some useful alternatives. Forth and Wiggins (2009), having calculated the *coverage*, *compactness* and *compression ratio* for all the discovered patterns in a dataset, normalise the values for each variable independently and linearly to $[0, 1]$. One of their proposed combinations, which I refer to as the *threeCs* variable, is again multiplicative, with $a = b = c = 1$.¹

Expected occurrences: Let Z be a random variable giving the number of times the pattern $P \subseteq D$ or one of its translations occur over the excerpt (dataset), and let $\mathbb{E}(Z)$ be the expectation of Z .

Let D^* , D' , and π be as described in Def. 3.1. It is supposed that the music can be modeled by independent and identically distributed random variables X_1, X_2, \dots , each having the distribution given in π . This is what is meant by a ‘zero-order model’ (Conklin and Bergeron, 2008, p. 64). The probability of seeing the pattern $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_l\} \subseteq D$

¹It should be noted that Forth and Wiggins (2009) actually use an altered version of *coverage* as the first term in their product.

is the probability of the event

$$A_1 = \{X_1 = \mathbf{p}'_1 = \mathbf{d}'_{i_1}, X_2 = \mathbf{p}'_2 = \mathbf{d}'_{i_2}, \dots, X_l = \mathbf{p}'_l = \mathbf{d}'_{i_l}\}, \quad (\text{C.6})$$

so that

$$\mathbb{P}(A_1) = \prod_{j=1}^l \pi_{i_j}. \quad (\text{C.7})$$

There may also be translations P_2, P_3, \dots, P_M of P with corresponding events A_2, A_3, \dots, A_M that have nonzero probability. The probability of seeing the pattern $P \subseteq D$ or one of its translations is the probability of the event $A = A_1 \cup A_2 \cup \dots \cup A_M$. These events are mutually exclusive,

so

$$\mathbb{P}(A) = \sum_{i=1}^M \mathbb{P}(A_i). \quad (\text{C.8})$$

Let Y be an indicator variable for the event A , and Z be the random variable for the number of times A happens across a dataset consisting of n datapoints. It follows that $Z = Y_1 + Y_2 + \dots + Y_K$, where K is yet to be determined. It is $\mathbb{E}(Z)$, the expected number of times the event A happens across a dataset, that is required:

$$\mathbb{E}(Z) = \mathbb{E}\left(\sum_{i=1}^K Y_i\right) \quad (\text{C.9})$$

$$= \sum_{i=1}^K \mathbb{E}(Y_i) \quad (\text{C.10})$$

$$= K\mathbb{E}(Y) \quad (\text{C.11})$$

$$= K\mathbb{P}(A), \quad (\text{C.12})$$

where (C.10) follows by linearity of expectation, (C.11) follows by

Y_1, Y_2, \dots, Y_K being identically distributed, and (C.12) follows as Y is an indicator variable for the event A .

If it is only nonoverlapping patterns with no *interpolation* that are allowed, then $K = n/l$, or $\lfloor n/l \rfloor$, as this is the number of nonoverlapping patterns of size l with no gaps that can fit in a dataset of size n . If overlaps are allowed, $K = n - l + 1$. If any amount of interpolation is permitted then K could be as large as $\binom{n}{l}$, the number of ways of choosing l objects from n . This makes the last definition of K too lenient: it leads to counting instances of patterns that would not be heard. For example suppose $P = \{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3\}$ and n is large. Even if $X_1 = \mathbf{p}'_1, X_2 = \mathbf{p}'_2$, and $X_n = \mathbf{p}'_3$, then this should not count as an instance of P , as the gap between X_2 and X_n is too large. It seems reasonable to limit the amount of interpolation according to the *span* of the original pattern, denoted $s(P, D)$, given by the denominator of (7.3). So K should be the number of ways of choosing l objects from n such that these objects do not exceed the span:

$$K = \binom{s(P, D)}{l} + [n - s(P, D)] \binom{s(P, D) - 1}{l - 1}. \quad (\text{C.13})$$

It is common to transform likelihoods and expectations, as the untransformed values can have a large range—in the case of the *expected occurrences* variable, 10^{-183} to 10^6 . Temperley (2009, p. 14) uses a log transform, whereas I map $[10^{-183}, 10^6]$ to $[10^{-2}, 10^2]$ using a power law

$$\text{expected_occurrences}(P, D) = \|\mathbb{E}(Z)\| = a\mathbb{E}(Z)^b, \quad (\text{C.14})$$

where $a \approx 71.12$ and $b \approx 0.02$ are constants determined by the chosen interval $[10^{-2}, 10^2]$. The reason for not using a log transform is that *expected occurrences* is derived from a nonnegative random variable. Therefore, it too should be nonnegative, but $\log x < 0$ for $x < 1$. It would be inappropriate to include negative values in subsequent variables (see *interest* and *score*), hence the use of a power law instead. I acknowledge that $[10^{-2}, 10^2]$ is an arbitrary—and therefore somewhat unsatisfactory—choice of interval. A second, less arbitrary solution was also considered (see *geometric mean likelihood* below), but this variable did not emerge as significant.

Geometric mean likelihood is a slight variant on (C.7):

$$\text{geom_mean_likelihood}(P, D) = \left(\prod_{j=1}^l \pi_{i_j} \right)^{(1/l)}, \quad (\text{C.15})$$

where $\pi_{i_1}, \pi_{i_2}, \dots, \pi_{i_l}$ are the individual probabilities defined a few lines before (C.6).

Interest: The mathematical definition (Conklin and Bergeron, 2008, p. 64)

is

$$\text{interest}(P, D) = \frac{\text{occurrences}(P, D)}{\text{expected_occurrences}(P, D)} = \frac{|TEC(P, D)|}{\|\mathbb{E}(Z)\|}. \quad (\text{C.16})$$

Score: The score of a pattern in a dataset is the squared difference between observed and expected occurrences, divided by expected occurrences

(Conklin and Anagnostopoulou, 2001).

$$\text{score}(P, D) = \frac{(|TEC(P, D)| - \|\mathbb{E}(Z)\|)^2}{\|\mathbb{E}(Z)\|}. \quad (\text{C.17})$$

This variable bears a very close resemblance to Pearson's statistic (Davison, 2003), and it seems (from Conklin, 2008) that *interest* may be based on the related concept of likelihood ratio.

Prominence (also ‘selection function’) is the name given to a formula by Cambouropoulos (2006), involving the variables *cardinality*, *occurrences*, and *coverage*.

$$\text{prominence}(P, D) = |P|^a \cdot m^b \cdot 10^{c \frac{\text{coverage}(P, D) - m|P|}{\text{coverage}(P, D)}}, \quad (\text{C.18})$$

where $a = 1$, $b = 2$, and $c = 3$.

Alternative prominence: There are now several explanatory variables that involve a product of *cardinality* and *occurrences* (e.g., *mean-centred cardinality* \times *occurrences*, *coverage*, and *prominence*). In (C.18) the variable *occurrences* is squared (the m^b term) but it could be argued that *cardinality* is the dominant factor, and that this variable ought to be squared instead. This is the intuition behind the variable *alternative prominence*:

$$\text{alt_prominence}(P, D) = \frac{\text{occurrences} \cdot (\text{cardinality} - 1)^2}{\text{cardinality_of_dataset}} \quad (\text{C.19})$$

$$= m(|P| - 1)^2/n. \quad (\text{C.20})$$

Maximum pitch centre: The previous variables have not really taken into account the *musical* attributes of a pattern, and whether they might make it noticeable/important. Pearce and Wiggins (2007) quantified specific musical attributes in order to tease out deficiencies in their generated chorale melodies. The next ten variables are adapted from their work. *Pitch centre* is defined as ‘the absolute distance, in semitones, of the mean pitch of a [pattern] . . . from the mean pitch of the dataset’ (Pearce and Wiggins, 2007, p. 78; see also von Hippel, 2000). By taking the maximum *pitch centre* over all occurrences of a pattern, I hope to isolate either unusually high, or unusually low occurrences. Denoting the mean MIDI note number of a subset $Q \subseteq D$ by \bar{y}_Q ,

$$\text{max_pitch_centre}(P, D) = \max\{|\bar{y}_Q - \bar{y}_D| : Q \in TEC(P, D)\}. \quad (\text{C.21})$$

Signed pitch range is defined as the ‘distance, in semitones, of the pitch range of a [pattern] . . . from the mean pitch range of [other discovered patterns from the same dataset]’ (Pearce and Wiggins, 2007, p. 78; see also von Hippel, 2000). This variable is based on the suggestion that the larger a pattern’s pitch range, the more noticeable it is. Denoting the range in semitones of a pattern P in a dataset D by $r(P)$, and the other discovered patterns in D by P_1, P_2, \dots, P_M , with ranges $r(P_1), r(P_2), \dots, r(P_M)$,

$$\text{signed_pitch_range}(P, D) = r(P) - \frac{1}{M} \sum_{i=1}^M r(P_i). \quad (\text{C.22})$$

Unsigned pitch range is defined as the absolute value of the *signed pitch*

range of a pattern. The previous variable took account of patterns with unusually large pitch ranges, based on findings that such patterns are more noticeable/important. But one could argue instead that unusually large *or* small pitch ranges give rise to noticeable patterns. Including the variable *unsigned pitch range* in the regression allows this argument to be considered as well.

Small intervals: This variable counts the number of small intervals (less than two steps on the stave) that are present in the melody line of a pattern. The intuition is that scalic, static or stepwise melodies may be rated as more noticeable/important. A *top-line* definition of melody is applied: at each of the pattern's distinct ontimes there will be at least one datapoint present. At this ontime the melody takes the value of the highest pitch present. If a melody consists of the morphetic pitch numbers y_1, y_2, \dots, y_l , then

$$\text{small_intervals}(P, D) = \left| \left\{ y_i : |y_i - y_{i-1}| < 2, i = 2, 3, \dots, l \right\} \right|. \quad (\text{C.23})$$

Intervallic leaps: This variable counts the number of intervallic leaps (more than two steps on the stave) that are present in the melody line of a pattern, the intuition being that leaping melodies may be rated as more noticeable/important. The same *top-line* rule as above is applied. If a melody consists of the morphetic pitch numbers y_1, y_2, \dots, y_l , then

$$\text{intervallic_leaps}(P, D) = \left| \left\{ y_i : |y_i - y_{i-1}| > 2, i = 2, 3, \dots, l \right\} \right|. \quad (\text{C.24})$$

Chromatic: The variable *chromatic* is the maximum number of nonkey notes present, taken over all occurrences of a pattern. A particularly chromatic pattern is likely to be more noticeable—and rated higher therefore—than one that remains entirely in key.

Cadential: If a pattern contains a cadential figure, the variable *cadential* takes the value 1, and 0 otherwise. Cadences are often mentioned in music-analytical essays, as they help to segment long passages. Therefore if a pattern contains a cadential figure it may be rated as noticeable/important. An algorithmic definition of *cadence* was not used, as it is more convenient to assign values by hand.

Phrasal: This variable takes high values for patterns that coincide with phrase marks. Take an occurrence of a pattern: its first note may coincide with the opening of a phrase mark and its last note may coincide with the closing of a phrase mark. A point is scored for each so that, per occurrence, a score of 0, 1, or 2 is possible. These scores are then averaged across all occurrences of a pattern.

Rhythmic density is defined as ‘the mean number of events per tactus beat’ (Pearce and Wiggins, 2007, p. 78; see also Eerola and North, 2000). It is likely that this variable and *compactness* will be highly positively correlated, but being a specific musical property, *rhythmic density* may be a more suitable variable. In a pattern $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_l\}$, let \mathbf{p}_i have ontine x_i , $i = 1, 2, \dots, l$. The tactus beats are then the integers from $a = \lfloor x_1 \rfloor$ to $b = \lfloor x_l \rfloor$, assuming that beats coincide with integer ontimes and that the bottom number in the time signature does

not change over the course of the pattern. The rhythmic density of the pattern at beat $c \in [a, b]$, denoted $\rho(P, c)$, is given by

$$\rho(P, c) = |\{\mathbf{p}_i \in P : \lfloor x_i \rfloor = c\}|, \quad (\text{C.25})$$

so that

$$\text{rhythmic_density}(P) = \frac{1}{b - a + 1} \sum_{c \in [a, b]} \rho(P, c). \quad (\text{C.26})$$

Rhythmic variability is defined as ‘the degree of change in note duration (*i.e.*, the standard deviation of the log of the event durations)’ (Pearce and Wiggins, 2007, p. 78; see also Eerola and North, 2000). While it has been suggested that patterns with much rhythmic variation are more difficult to perceive (Eerola and North, 2000), such patterns could actually be more distinctive/important. For a pattern $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_l\}$, denote the durations of each datapoint by z_1, z_2, \dots, z_l . Then

$$\text{rhythmic_variability}(P) = \sqrt{\frac{1}{l} \sum_{i=1}^l (\log z_i - \overline{\log z})^2}. \quad (\text{C.27})$$

Signed dynamic level: The last four variables attempt to cover musical aspects that have not been taken into account already. Participants listened to recordings of excerpts and had access to the entire score of each piece (Paderewski, 1953). So consciously or unconsciously, participants may rate as more noticeable patterns that are marked and performed louder. The variable *signed dynamic level* involves mapping dynamic levels to numbers, and summing over the dynamic levels

that apply to a pattern. Where occurrences of a pattern have different *signed dynamic levels*, the maximum value is taken. The mapping is given by

$$\begin{aligned} \mathbf{pp} &\mapsto -3, \quad \mathbf{sp} \mapsto -2, \quad \mathbf{p} \mapsto -1, \\ & \qquad \qquad \qquad \textit{mezza voce} \mapsto 0, \\ \mathbf{f} &\mapsto 1, \quad \mathbf{sf} \mapsto 2, \quad \mathbf{ff} \mapsto 3. \end{aligned} \tag{C.28}$$

Chopin seems not to have distinguished between \mathbf{mp} and \mathbf{mf} , preferring the term *mezzo voce*. Crescendi and diminuendi are mapped to $1/2$ and $-1/2$ respectively. The terms *sotto voce* and *dolce* are mapped to -1 and 0 respectively.

Unsigned dynamic level: One could argue that it is naïve to assume that louder patterns are more noticeable/important. Could not a pattern that was played suddenly very softly be just as noticeable/important? The variable *unsigned dynamic level* takes the absolute values of the mapping used in the previous variable. In all other respects it is the same, and should assume large values for patterns that contain extreme dynamics, one way or the other.

Tempo fluctuation: Given fluctuations in tempo are used by composers and performers to emphasise aspects of the music, it seems reasonable to assume that patterns containing a pause mark, accelerando, ritardando, or rubato would be more noticeable/important than those that did not. The *tempo fluctuation* of a pattern is defined to be the number of tempo directions that apply to the pattern. Where occurrences of a

pattern have different *tempo fluctuation* values, the maximum is taken.

Metric syncopation: The term *hemiola* applies to a scenario in which six beats are arranged as three groups of two, contrary to the prevailing arrangement of two groups of three, as with pattern *H* in Fig. 6.9. *Hemiola* is a little too specific to be a variable in itself, so I define *metric syncopation* to apply to a scenario in which the prevailing arrangement of beats is contradicted. From the point of view of rating patterns, a pattern that contains a metric syncopation is likely to be noticeable/important. If a pattern contains a metric syncopation, then the variable *metric syncopation* takes the value 1, and 0 otherwise.

The next eight variables are used in Chapter 10 (not in Chapter 6) to determine whether certain attributes of a music excerpt are useful predictors of stylistic success.

Maximum metric weight entropy. For an excerpt with at least one bar containing syncopations (or otherwise ambiguous metric hierarchy), the *maximum metric weight entropy* should be large. For an excerpt with a very consistent sense of metre, on the other hand, the *maximum metric weight entropy* should be small. So this variable is a more nuanced version of the variable *metric syncopation*. Let an excerpt have dataset representation D and consist of N bars. For notes in the i th bar of the excerpt, let the distinct ontimes be $t_{i,1}, t_{i,2}, \dots, t_{i,n_i}$. Each ontione has a *general metric weight* $W_{l,p}(t_{i,j})$, where $j = 1, 2, \dots, n_i$, as defined in Def. 4.7 and originally by Volk (2008). The weights for the i th bar can be scaled so that they sum to one, and thought of as a probability

vector \mathbf{p}_i . If X_i is a discrete random variable with distribution \mathbf{p}_i , then the *maximum metric weight entropy* for the excerpt is

$$\text{max_metric_weight_entropy}(D, l, p) = \max\{H(X_i) : i = 1, 2, \dots, N\}, \quad (\text{C.29})$$

where the entropy $H(X_i)$ was defined in Def. A.41, and $l = p = 2$ are parameters in Inner Metric Analysis (Volk, 2008).

Mean metric weight entropy. This variable is very similar to *maximum metric weight entropy*. Carrying over definitions of D, l, p , and X_i from the previous variable,

$$\text{mean_metric_weight_entropy}(D, l, p) = \frac{1}{N} \sum_{i=1}^N H(X_i). \quad (\text{C.30})$$

It is useful to look at both types of summary (maximum and mean) for an excerpt of music, because a listener's rating of stylistic success could be based on either their sensitivity to the most extreme instance of some aspect (in which case the maximum is appropriate) or the general level of the aspect (in which case the mean is appropriate), or some combination of both.

Relative metric weight entropy. Let the *maximum metric weight entropy* of an excerpt be denoted by $H_{\max}(X)$. Suppose I want to compare the maximum metric weight entropy for one excerpt, represented by the random variable X , with the maximum metric weight entropies of several excerpts in a database, represented by the random variables Y_1, Y_2, \dots, Y_n . This is the intuition behind *relative metric weight entropy*.

tropy, and it is defined by

$$\text{rel_metric_weight_entropy}(X, \mathbf{Y}) = \left| H_{\max}(X) - \frac{1}{n} \sum_{i=1}^n H_{\max}(Y_i) \right|. \quad (\text{C.31})$$

Maximum chord weight entropy. This variable applies *maximum metric weight entropy* to a more abstract (chord) representation of an excerpt. For an excerpt with at least one bar containing syncopated chord changes (or otherwise ambiguous harmonic rhythm), the *maximum chord weight entropy* should be large. For an excerpt with a very consistent sense of harmonic rhythm, on the other hand, the *maximum chord weight entropy* should be small. When the HarmAn algorithm (Pardo and Birmingham, 2002) is applied to an excerpt of music, let E be the resultant *chord dataset*, as discussed in Sec. 4.4 and previously on p. 23. For chord datapoints in E referring to the i th bar of the excerpt, let the distinct ontimes be $t_{i,1}, t_{i,2}, \dots, t_{i,n_i}$. Each ontime has a *general metric weight* $W_{l,p}(t_{i,j})$, where $j = 1, 2, \dots, n_i$, as defined in Def. 4.7 and originally by Volk (2008). The weights for the i th bar can be scaled so that they sum to one, and thought of as a probability vector \mathbf{p}_i . If X_i is a discrete random variable with distribution \mathbf{p}_i , then the *maximum chord weight entropy* for the excerpt is

$$\text{max_chord_weight_entropy}(E, l, p) = \max\{H(X_i) : i = 1, 2, \dots, N\}, \quad (\text{C.32})$$

where $l = p = 2$ are parameters in Inner Metric Analysis (Volk, 2008).

Mean chord weight entropy. This variable is very similar to *maximum*

chord weight entropy. Carrying over definitions of E, l, p , and X_i from the previous variable,

$$\text{mean_chord_weight_entropy}(E, l, p) = \frac{1}{N} \sum_{i=1}^N H(X_i). \quad (\text{C.33})$$

Relative chord weight entropy. This variable is analogous to *relative metric weight entropy*, but applied to the *chord dataset* E , rather than the dataset D for the musical surface. Let the *maximum chord weight entropy* of an excerpt's chord dataset be denoted by $H'_{\max}(X)$, and suppose I want to compare the maximum chord weight entropy for one excerpt, represented by the random variable X , with the maximum chord weight entropies of several excerpts in a database, represented by the random variables Y_1, Y_2, \dots, Y_n . The *relative chord weight entropy* for a chord dataset is defined by

$$\text{rel_chord_weight_entropy}(X, \mathbf{Y}) = \left| H'_{\max}(X) - \frac{1}{n} \sum_{i=1}^n H'_{\max}(Y_i) \right|. \quad (\text{C.34})$$

Keyscape entropy. An excerpt of music may not establish a strong sense of key, in which case its *keyscape plot* (Sapp, 2005, and as discussed on p. 26) is likely to contain boxes of many different colours. The *keyscape entropy* of such an excerpt should be large. An excerpt that does establish a strong sense of key, on the other hand, will have a keyscape plot where *one* colour predominates, and should have a small *keyscape entropy*. As a box in a keyscape plot can be one of 24 colours (one colour for each major and minor key), the relative frequency of occurrence of

each colour can be written as a probability vector $\mathbf{p} = (p_1, p_2, \dots, p_{24})$.

If a dataset representation D of an excerpt gives rise to a keyscape with the probability vector \mathbf{p} , then the *keyscape entropy* of the dataset D is defined by

$$\text{keyscape_entropy}(D) = H_K(D) = - \sum_{i=1}^{24} p_i \log_2 p_i. \quad (\text{C.35})$$

Relative keyscape entropy. This variable is analogous to *relative metric weight entropy*, but applied to the concept of *keyscape* (Sapp, 2005). Let the *keyscape entropy* of an excerpt's dataset be denoted by $H_K(D)$, and suppose I want to compare the keyscape entropy for one excerpt with the keyscape entropies of several excerpts in a database, represented by the datasets D_1, D_2, \dots, D_n . The *relative keyscape entropy* for a dataset is defined by

$$\text{rel_keyscape_entropy}(D, D_1, D_2, \dots, D_n) = \left| H_K(D) - \frac{1}{n} \sum_{i=1}^n H_K(D_i) \right|. \quad (\text{C.36})$$



Ten top-rated, SIACT-discovered patterns in Chopin's op.56 no.1

Pattern A, 1st occurrence

Allegro non tanto rating 7.82 ontime, MNN

Musical score for Pattern A, 1st occurrence, bars 1-5. The score is in 3/4 time, B major (two sharps). The piano part consists of two staves. The top staff shows a series of eighth-note chords: G major (G-B-D), C major (C-E-G), F major (F-A-C), and B major (B-D-F#). The bottom staff shows a series of eighth-note chords: E major (E-G-B), A major (A-C-E), D major (D-F#-A), and G major (G-B-D). The notes are black, indicating they are part of the pattern.

Musical score for Pattern A, 1st occurrence, bars 6-10. The score continues in 3/4 time, B major. The piano part shows a series of eighth-note chords: E major (E-G-B), A major (A-C-E), D major (D-F#-A), and G major (G-B-D). The notes are black, indicating they are part of the pattern.

Musical score for Pattern A, 1st occurrence, bars 11-16. The score continues in 3/4 time, B major. The piano part shows a series of eighth-note chords: E major (E-G-B), A major (A-C-E), D major (D-F#-A), and G major (G-B-D). The notes are black, indicating they are part of the pattern.

Figure D.1: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. The first occurrence of pattern *A* is indicated by black noteheads. Output by SIACT for the projection on to ontime and MNN, and rated 7.82 by the formula in (6.4). In this appendix, SIACT parameters are $a = 4/5$, $b = 5$, and dynamic/expressive markings have been removed to aid clarity.

Pattern A, 2nd occurrence

Allegro non tanto

rating 7.82

ontime, MNN

The musical score displays three staves of music. The top staff is the treble clef, the middle staff is the bass clef, and the bottom staff is the bass clef. The key signature is four sharps. The time signature is 3/4. The tempo is Allegro non tanto. Measure 1 starts with a half note followed by eighth notes. Measure 2 starts with a quarter note followed by eighth notes. Measure 3 starts with a half note followed by eighth notes. Measure 4 starts with a quarter note followed by eighth notes. Measure 5 starts with a half note followed by eighth notes. Measure 6 starts with a half note followed by eighth notes. Measure 7 starts with a half note followed by eighth notes. Measure 8 starts with a half note followed by eighth notes. Measure 9 starts with a half note followed by eighth notes. Measure 10 starts with a half note followed by eighth notes. Measure 11 starts with a half note followed by eighth notes. Measure 12 starts with a half note followed by eighth notes. Measure 13 starts with a half note followed by eighth notes. Measure 14 starts with a half note followed by eighth notes. Measure 15 starts with a half note followed by eighth notes. Measure 16 starts with a half note followed by eighth notes.

Figure D.2: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. The second occurrence of pattern *A* is indicated by black noteheads.

Pattern B, 1st occurrence

Allegro non tanto

rating 7.75

ontime, MNN

A musical score for piano, featuring three staves. The top staff is treble clef, the middle staff bass clef, and the bottom staff bass clef. The key signature is A major (three sharps). The time signature is common time (indicated by 'C'). The score consists of three systems of music. System 1 (measures 1-5) shows the right hand playing eighth-note chords and sixteenth-note patterns, while the left hand provides harmonic support. System 2 (measures 6-10) continues this pattern with more complex harmonic movement. System 3 (measures 11-15) concludes the section with a final flourish of eighth-note chords and sixteenth-note patterns.

Figure D.3: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. The first occurrence of pattern *B* is indicated by black noteheads. Output by SIACT for the projection on to ontine and MNN, and rated 7.75 by the formula in (6.4).

Pattern B, 2nd occurrence

Allegro non tanto

rating 7.75

ontime, MNN

The musical score displays three staves of music. The top two staves are for the treble and bass voices, while the bottom staff provides harmonic information with Roman numerals. Measure 1 begins with a forte dynamic. Measure 6 follows with a change in harmonic context. Measure 11 marks the start of the second occurrence of pattern B. The music is set in 3/4 time and uses a key signature of four sharps.

Figure D.4: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. The second occurrence of pattern *B* is indicated by black noteheads.

Pattern C, 1st occurrence

Allegro non tanto

rating 7.69

ontime, MPN

A musical score for piano, featuring two staves. The top staff uses a treble clef and 3/4 time, with a key signature of four sharps. The bottom staff uses a bass clef and 3/4 time, also with a key signature of four sharps. Measure 6 begins with a forte dynamic. Measure 11 continues the musical line.

Figure D.5: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. The first occurrence of pattern C is indicated by black noteheads. Output by SIACT for the projection on to ontine and MPN, and rated 7.69 by the formula in (6.4).

Pattern C, 2nd occurrence

Allegro non tanto

rating 7.69

ontime, MPN

The musical score displays three staves of music. The top staff is the treble clef, the middle staff is the bass clef, and the bottom staff is also the bass clef. The key signature is four sharps. The time signature is 3/4. The tempo is Allegro non tanto. The rating is 7.69 and it is categorized as ontime, MPN. The score shows three distinct sections of music, likely corresponding to the introduction, first occurrence of pattern C, and second occurrence of pattern C. The second occurrence of pattern C is specifically highlighted with black noteheads.

Figure D.6: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. The second occurrence of pattern *C* is indicated by black noteheads.

Pattern D, 1st occurrence

Allegro non tanto

rating 7.45

ontime, MPN

A musical score for piano, featuring three staves. The top staff uses a treble clef, the middle staff an alto clef, and the bottom staff a bass clef. The key signature is A major (three sharps). The time signature is common time (indicated by a '4'). Measure 1 consists of a single note followed by a rest. Measures 2-5 show a repeating pattern of eighth-note chords. Measure 6 begins with a half note, followed by eighth-note chords. Measure 11 concludes the page with a half note.

Figure D.7: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. The first occurrence of pattern D is indicated by black noteheads. Output by SIACT for the projection on to ontine and MPN, and rated 7.45 by the formula in (6.4).

Pattern D, 2nd occurrence

Allegro non tanto

rating 7.45

ontime, MPN

The musical score displays three staves of music. The top staff is the treble clef, the middle staff is the bass clef, and the bottom staff is the bass clef. The key signature is four sharps. The time signature is 3/4. The tempo is Allegro non tanto. The rating is 7.45. The pattern is ontime, MPN. The score shows bars 1 through 16. Pattern D, the second occurrence, is indicated by black noteheads.

Figure D.8: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. The second occurrence of pattern *D* is indicated by black noteheads.

Pattern E, all occurrences

rating 7.01

ontime, MNN

Allegro non tanto

1st occurrence

2nd occurrence

3rd occurrence

11

Figure D.9: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. Occurrences of pattern E are indicated by black noteheads. Output by SIACT for the projection on to ontine and MNN, and rated 7.01 by the formula in (6.4).

Pattern F, all occurrences

Allegro non tanto

rating 6.95 ontime, MNN

The musical score consists of three staves of piano music. The first staff (treble clef) contains bars 1 through 4. The second staff (bass clef) contains bar 6. The third staff (treble clef) contains bar 11. In each staff, specific noteheads are highlighted in black to indicate occurrences of 'pattern F'. The music is in B major (two sharps) and 3/4 time. The tempo is Allegro non tanto.

1st occurrence 2nd occurrence 3rd occurrence

6 4th occurrence

11

Figure D.10: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. Occurrences of pattern *F* are indicated by black noteheads. Output by SIACT for the projection on to ontime and MNN, and rated 6.95 by the formula in (6.4).

Pattern G, all occurrences

Allegro non tanto

rating 6.90

ontime, MPN

1st occurrence 2nd occurrence

3rd occurrence

11

Figure D.11: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. Occurrences of pattern G are indicated by black noteheads. Output by SIACT for the projection on to ontine and MPN, and rated 6.90 by the formula in (6.4).

Pattern H, all occurrences

Allegro non tanto

rating 6.80 ontime, MPN

1st occurrence 2nd occurrence 3rd occurrence

4th occurrence

6

11

Figure D.12: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. Occurrences of pattern *H* are indicated by black noteheads. Output by SIACT for the projection on to ontime and MPN, and rated 6.80 by the formula in (6.4).

Pattern I, occurrences 1, 3, and 5-7



Figure D.13: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. Due to overlapping notes, two occurrences (second and fourth) of pattern *I* are not shown. Output by SIACT for the projection on to ontime and duration, and rated 6.69 by the formula in (6.4).

Pattern J, occurrences 1 and 3

Allegro non tanto

rating 6.69

ontime, duration

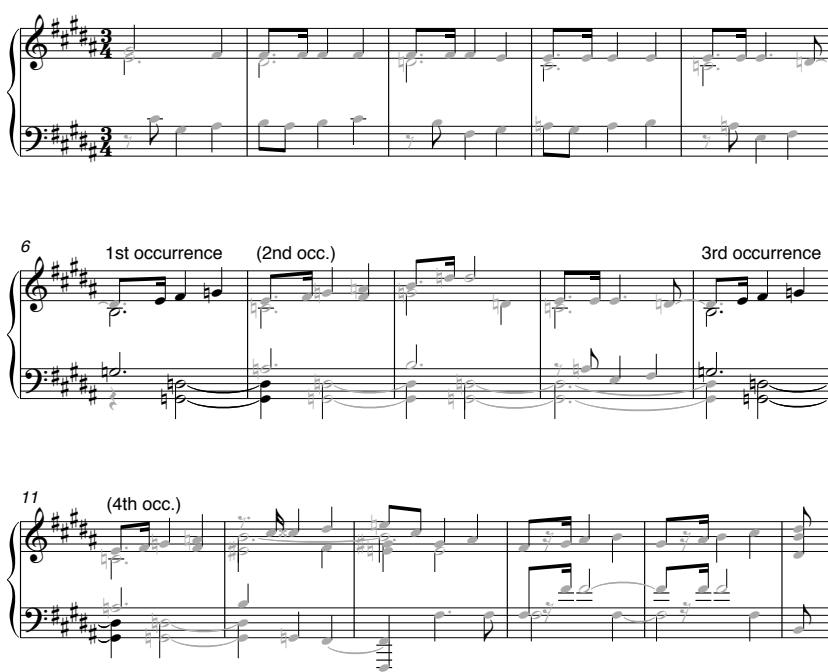


Figure D.14: Bars 1-16 from the Mazurka in B major op.56 no.1 by Chopin. Due to overlapping notes, two occurrences (second and fourth) of pattern J are not shown. Output by SIACT for the projection on to ontime and duration, and rated 6.69 by the formula in (6.4).

\mathcal{E}

Four more mazurka sections generated by the models in Chapter 9

A mazurka section generated by the model Racchmaninof-Oct2010 is shown in Fig. 9.6. This excerpt corresponds to stimulus 29 in Chapter 10 (System B*, p. 253). The parameter values were $c_{\text{absb}} = 10$, $c_{\text{src}} = 4$, $c_{\text{min}} = c_{\text{max}} = 12$, $\bar{c} = 19$, $c_{\text{prob}} = .2$, $c_{\text{beat}} = 12$, and $c_{\text{for}} = c_{\text{back}} = 3$. Four more computer-generated mazurka sections are given overleaf.



Figure E.1: Mazurka section generated by the model Racchman-Oct2010. This excerpt corresponds to stimulus 19 in Chapter 10 (System A, p. 252). Parameter values $c_{\text{absb}} = 10$, $c_{\text{src}} = 4$, $c_{\text{min}} = c_{\text{max}} = \bar{c} = 19$, $c_{\text{prob}} = .2$, and $c_{\text{beat}} = 12$, and $c_{\text{for}} = c_{\text{back}} = 3$.



Figure E.2: Mazurka section generated by the model Racchman-Oct2010. This excerpt corresponds to stimulus 20 in Chapter 10 (System A, p. 252). Parameter values $c_{\text{absb}} = 10$, $c_{\text{src}} = 4$, $c_{\text{min}} = c_{\text{max}} = \bar{c} = 19$, $c_{\text{prob}} = .2$, and $c_{\text{beat}} = 12$, and $c_{\text{for}} = c_{\text{back}} = 3$.



Figure E.3: Mazurka section generated by the model Racchmaninof-Oct2010. This excerpt corresponds to stimulus 27 in Chapter 10 (System B, p. 253). Parameter values $c_{absb} = 10$, $c_{src} = 4$, $c_{min} = c_{max} = \bar{c} = 31$, $c_{prob} = 1$, and $c_{beat} = 12$, and $c_{for} = c_{back} = 3$.



Figure E.4: Mazurka section generated by the model Racchmaninof-Oct2010. This excerpt corresponds to stimulus 28 in Chapter 10 (System B^* , p. 253). Parameter values $c_{absb} = 10$, $c_{src} = 4$, $c_{min} = c_{max} = \bar{c} = 19$, $c_{prob} = .2$, and $c_{beat} = 12$, and $c_{for} = c_{back} = 3$.

References

- Bret J. Aarden. *Dynamic melodic expectancy*. PhD thesis, School of Music, Ohio State University, 2003.
- Moray Allan. Harmonising chorales in the style of Johann Sebastian Bach. Master's thesis, School of Informatics, University of Edinburgh, 2002.
- Teresa M. Amabile. Children's artistic creativity: Detrimental effects of competition in a field setting. *Personality and Social Psychology Bulletin*, 8:573–578, 1982.
- Teresa M. Amabile. *Creativity in Context*. Westview Press, Boulder, CO, 1996.
- Torsten Anders and Eduardo R. Miranda. Constraint application with higher-order programming for modeling music theory. *Computer Music Journal*, 34(2):25–38, 2010.
- AQA. *Specification for AS and A level music*. AQA, Manchester, UK, 2007.
- AQA. *Specimen question paper for GCE music unit 5*. AQA, Manchester, UK, 2009.
- Christopher Ariza. The interrogator as critic: The turing test and the evaluation of generative music systems. *Computer Music Journal*, 33(2):48–70, 2009.
- Rebecca A. Baltzer, editor. *Le magnus liber organi de Notre-Dame de Paris: Les clausules à deux voix du manuscrit de Florence, Biblioteca Medicea Laurenziana, pluteus 29.1*, volume 5. Éditions de l'Oiseau-Lyre, Monaco, 1995.

- Bernard Bel and Jim Kippen. Bol Processor grammars. In Mira Balaban, Kemal Ebcioğlu, and Otto Laske, editors, *Understanding music with AI: Perspectives on music cognition*, pages 366–401. AAAI Press, Menlo Park, CA, 1994.
- Ian D. Bent and Anthony Pople. Analysis. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 1, pages 526–589. Macmillan, London, UK, 2nd edition, 2001.
- Margaret Bent. Isorhythm. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 12, pages 618–623. Macmillan, London, UK, 2nd edition, 2001.
- Mathieu Bergeron and Darrell Conklin. Structured polyphonic patterns. In Juan Pablo Bello, Elaine Chew, and Douglas Turnbull, editors, *Proceedings of the International Symposium on Music Information Retrieval*, pages 69–74, Philadelphia, PA, 2008. International Society for Music Information Retrieval. Retrieved 14 July, 2009 from http://ismir2008.ismir.net/papers/ISMIR2008_177.pdf.
- Idil Biret. *Chopin mazurkas*, volume 1/2. Naxos, Munich, 1990.
- Stephen Blum. Composition. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 6, pages 186–201. Macmillan, London, UK, 2nd edition, 2001.
- Blur. *Parklife*. Food Records, UK, 1994.
- Margaret A. Boden. Computer models of creativity. In Robert J. Sternberg, editor, *Handbook of creativity*, pages 351–373. Cambridge University Press, Cambridge, UK, 1999.
- Albert S. Bregman. *Auditory scene analysis: The perceptual organization of sound*. MIT Press, Cambridge, MA, 1990.
- Timothy Budd. *An introduction to object-oriented programming*. Addison-Wesley, Reading, MA, 1991.
- Emilios Cambouropoulos. Musical parallelism and melodic segmentation: A computational approach. *Music Perception*, 23(3):249–267, 2006.
- Emilios Cambouropoulos. Voice and stream: Perceptual and computational modeling of voice separation. *Music Perception*, 26(1):75–94, 2008.

-
- Cambridge University Faculty of Music. *Music Tripos examination, part IA*. Cambridge University Faculty of Music, Cambridge, UK, 2010a.
- Cambridge University Faculty of Music. *Music Tripos courses*. Cambridge University Faculty of Music, Cambridge, UK, 2010b.
- Peter J. Cameron. *Introduction to algebra*. Oxford University Press, Oxford, UK, 1998.
- Geoffrey Chew. Slur. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 23, pages 526–528. Macmillan, London, UK, 2nd edition, 2001.
- Shih-Chuan Chiu, Man-Kwan Shan, Jiun-Long Huang, and Hua-Fu Li. Mining polyphonic repeating patterns from music data using bit-string based approaches. In Regunathan Radhakrishnan and Rong Yan, editors, *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 1170–1173, New York, 2009. IEEE. Retrieved 9 February, 2010 from http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5202708.
- Lois Choksy. *The Kodály method: comprehensive music education from infant to adult*. Prentice-Hall, Englewood Cliffs, NJ, 1974.
- Parag Chordia, Avinash Sastry, Trishul Malikarjuna, and Aaron Albin. Multiple viewpoint modeling of tabla sequences. In J. Stephen Downie and Remco Veltkamp, editors, *Proceedings of the International Symposium on Music Information Retrieval*, pages 381–386, Utrecht, 2010. International Society for Music Information Retrieval. Retrieved 1 December, 2010 from <http://ismir2010.ismir.net/proceedings/ismir2010-65.pdf>.
- Raphaël Clifford, Manolis Christodoulakis, Tim Crawford, David Meredith, and Geraint A. Wiggins. A fast, randomised, maximal subset matching algorithm for document-level music retrieval. In Richard Dannenberg, Kjell Lemström, and Adam R. Tindale, editors, *Proceedings of the International Symposium on Music Information Retrieval*, pages 150–155, Victoria, Canada, 2006. International Society for Music Information Retrieval. Retrieved 12 October, 2009 from http://ismir2006.ismir.net/PAPERS/ISMIR06120_Paper.pdf.
- Nick Collins. Musical form and algorithmic composition. *Contemporary Music Review*, 28(1):103–114, 2009.
- Tom Collins, Robin Laney, Alistair Willis, and Paul H. Garthwaite. Using discovered polyphonic patterns to filter computer-generated music. In Dan

- Ventura, Alison Pease, Rafael Pérez y Pérez, Graeme Ritchie, and Tony Veale, editors, *Proceedings of the International Conference on Computational Creativity*, pages 1–10, Lisbon, 2010. University of Coimbra.
- Darrell Conklin. Representation and discovery of vertical patterns in music. In Christina Anagnostopoulou, Miguel Ferrand, and Alan Smaill, editors, *Music and Artificial Intelligence: Proceedings of the International Conference on Music and Artificial Intelligence*, volume 2445 of *Lecture Notes in Artificial Intelligence*, pages 32–42. Springer-Verlag, Berlin, 2002.
- Darrell Conklin. Music generation from statistical models. In *Proceedings of the AISB Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*, pages 30–35, Brighton, UK, 2003. SSAISB. Retrieved 28 March, 2010 from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.2086>.
- Darrell Conklin. Discovery of distinctive patterns in music. In *Proceedings of the International Workshop on Machine Learning and Music*, Helsinki, 2008. University of Helsinki. Retrieved July 14, 2009 from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.158.4152>.
- Darrell Conklin. Discovery of distinctive patterns in music. *Intelligent Data Analysis*, 14(5):547–554, 2010.
- Darrell Conklin and Christina Anagnostopoulou. Representation and discovery of multiple viewpoint patterns. In Andrew Schloss, Roger Dannenberg, and Peter Driessens, editors, *Proceedings of the International Computer Music Conference*, pages 479–485, San Francisco, CA, 2001. International Computer Music Association. Retrieved 16 July, 2009 from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.5364>.
- Darrell Conklin and Mathieu Bergeron. Feature set patterns in music. *Computer Music Journal*, 32(1):60–70, 2008.
- Darrell Conklin and Ian H. Witten. Multiple viewpoint systems for music prediction. *Journal of New Music Research*, 24(1):51–73, 1995.
- Nicholas Cook. *A guide to musical analysis*. J.M. Dent and Sons, London, UK, 1987.
- Nicholas Cook. Perception: A perspective from music theory. In Rita Aiello and John A. Sloboda, editors, *Musical perceptions*, pages 64–95. Oxford University Press, Oxford, UK, 1994.

-
- Nicholas Cook. *Music: A very short introduction*. Oxford University Press, Oxford, UK, 1998.
- Grosvenor Cooper and Leonard B. Meyer. *The rhythmic structure of music*. University of Chicago Press, Chicago, IL, 1960.
- David Cope. *Experiments in musical intelligence*. The Computer Music and Digital Audio Series. A-R Editions, Madison, WI, 1996.
- David Cope. *Mazurkas, after Chopin*. Spectrum Press, Paris, 1997.
- David Cope. *Virtual music: computer synthesis of musical style*. MIT Press, Cambridge, MA, 2001. (Includes essays by Douglas Hofstadter, Eleanor Selfridge-Field, Bernard Greenberg, Steve Larson, Jonathan Berger, and Daniel Dennett).
- David Cope. *The well-programmed clavier*. Spectrum Press, Paris, 2002.
- David Cope. *Computer models of musical creativity*. MIT Press, Cambridge, MA, 2005.
- Thomas H. Cormen. *Introduction to algorithms*. MIT Press, Cambridge, MA, 2nd edition, 2001. (First edition lists Charles E. Leiserson and Ronald L. Rivest as additional authors).
- Alastair Craft and Ian Cross. A n-gram approach to fugal exposition composition. In *Proceedings of the AISB Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*, pages 36–41, Brighton, UK, 2003. SSAISB. Retrieved 28 March, 2010 from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.96.3238>.
- Ian Cross. Music analysis and music perception. *Music Analysis*, 17(1):3–20, 1998.
- Carl Czerny. *School of practical composition*, volume 3. Robert Cocks & Co, London, UK, 1848. (Year of publication is approximate).
- Fergus Daly, David J. Hand, M. Chris Jones, A. Daniel Lunn, and Kevin J. McConway. *Elements of statistics*. Addison-Wesley, Wokingham, UK, 1995.
- Anthony C. Davison. *Statistical models*. Cambridge University Press, Cambridge, UK, 2003.

- Alain de Cheveigné. Multiple *F0* estimation. In DeLiang Wang and Guy J. Brown, editors, *Computational auditory scene analysis: Principles, algorithms, and applications*, pages 45–79. John Wiley & Sons, Hoboken, NJ, 2006.
- Peter Desain and Henkjan Honing. The quantization of musical time: A connectionist approach. *Computer Music Journal*, 13(3):56–66, 1989.
- Diana Deutsch. The processing of structured and unstructured tonal sequences. *Perception and Psychophysics*, 28(5):381–389, 1980.
- Luther A. Dittmer, editor. *Facsimile reproduction of the manuscript Firenze, Biblioteca Mediceo-Laurenziana, pluteo 29, I.* Institute of Medieval Music, Brooklyn, NY, 1966-7.
- Shyamala Doraisamy and Stefan Rüger. Robust polyphonic music retrieval with *N*-grams. *Journal of Intelligent Information Systems*, 21(1):53–70, 2003.
- Stephen Downes. Mazurka. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 16, pages 189–190. Macmillan, London, UK, 2nd edition, 2001.
- J. Stephen Downie. Music information retrieval. In Blaise Cronin, editor, *Annual review of information science and technology 37*, pages 295–340. Information Today, Medford, NJ, 2003.
- Kemal Ebcioğlu. An expert system for harmonizing chorales in the style of J.S. Bach. In Mira Balaban, Kemal Ebcioğlu, and Otto Laske, editors, *Understanding music with AI: Perspectives on music cognition*, pages 294–333. AAAI Press, Menlo Park, CA, 1994.
- Edexcel. *Question paper for GCE music unit 5*. Edexcel, London, UK, 2009.
- Tuomas Eerola and Adrian C. North. Expectancy-based model of melodic complexity. In Chris Woods, Geoff Luck, Renaud Brochard, Fred Seddon, and John A. Sloboda, editors, *Proceedings of the International Conference on Music Perception and Cognition*, Keele, UK, 2000. Department of Psychology, Keele University. Retrieved 12 June, 2010 from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.110.3186>.
- Arne Eigenfeldt and Philippe Pasquier. Realtime generation of harmonic progressions using constrained markov selection. In Dan Ventura, Alison Pease, Rafael Pérrez y Pérez, Graeme Ritchie, and Tony Veale, editors,

-
- Proceedings of the International Conference on Computational Creativity*, pages 16–25, Lisbon, 2010. University of Coimbra.
- Warren J. Ewens and Gregory R. Grant. *Statistical methods in bioinformatics: An introduction*. Springer-Verlag, New York, 2001.
- David Fallows. Tempo and expression marks. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 25, pages 271–279. Macmillan, London, UK, 2nd edition, 2001.
- Morwaread M. Farbood. *A quantitative, parametric model of musical tension*. PhD thesis, School of Architecture and Planning, Massachusetts Institute of Technology, 2006.
- Jonathan Finn, James Larcombe, Yasir Assam, Simon Whiteside, Mike Copperwhite, Paul Walmsley, Graham Westlake, Michael Eastwood, Andrew Davis, and Daniel Spreadbury. *Sibelius 6: Using the ManuScript language*. Avid Technology, Inc, Burlington, MA, 2009. Retrieved August 2, 2010 from <http://www.sibelius.com/download/software/win/ManuScriptLanguage.pdf>.
- Allen Forte and Steven E. Gilbert. *Introduction to Schenkerian analysis*. W. W. Norton and Company, New York, 1982.
- Jamie Forth and Geraint A. Wiggins. An approach for identifying salient repetition in multidimensional representations of polyphonic music. In Joseph Chan, Jackie Daykin, and M. Sohel Rahman, editors, *London algorithmics 2008: Theory and practice*, Texts in Algorithmics, pages 44–58. College Publications, London, UK, 2009.
- Andrew Gartland-Jones and Peter Copley. The suitability of genetic algorithms for musical composition. *Contemporary Music Review*, 22(3):43–55, 2003.
- Richard F. Goldman. Reviews of records (Varèse: *Ionisation*; *Density 21.5*; *Intégrales*; *Octandre*; *Hyperprism*; *Poème Electronique*. Instrumentalists, cond. Robert Craft. Columbia MS 6164 (stereo)). *The Musical Quarterly*, 47(1):133–134, 1961.
- James D. Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57(2):357–384, 1989.

- Stephen A. Hedges. Dice music in the eighteenth century. *Music and Letters*, 59(2):180–187, 1978.
- Maud Hickey. An application of amabile’s consensual assessment technique for rating the creativity of childrens musical compositions. *Journal of Research in Music Education*, 49(3):234–244, 2001.
- Hermann Hild, Johannes Feulner, and Wolfram Menzel. Harmonet: a neural net for harmonizing chorales in the style of J.S. Bach. In Richard P. Lippmann, John E. Moody, and David S. Touretzky, editors, *Advances in neural information processing systems 4*, pages 267–274. Morgan Kaufmann, Waltham, MA, 1992.
- Jia-Lien Hsu, Chih-Chin Liu, and Arbee L. P. Chen. Discovering nontrivial repeating patterns in music data. *IEEE Transactions on Multimedia*, 3(3):311–325, 2001.
- David Huron. *Music research using Humdrum: A user’s guide*. Center for Computer Assisted Research in the Humanities, Stanford, CA, 1999.
- David Huron. Tone and voice: A derivation of the rules of voice-leading from perceptual principles. *Music Perception*, 19(1):1–64, 2001a.
- David Huron. What is a musical feature? Forte’s analysis of Brahms’s opus 51, no.1, revisited. *Music Theory Online*, 7(4):69 numbered paragraphs, 2001b. Retrieved 14 July, 2009
http://mto.societymusictheory.org/issues/mto.01.7.4/mto.01.7.4.huron_frames.html.
- David Huron. Music information processing using the Humdrum Toolkit: Concepts, examples, and lessons. *Computer Music Journal*, 26(2):11–26, 2002.
- David Huron. *Sweet anticipation*. MIT Press, Cambridge, MA, 2006.
- Petr Janata, Jeffrey L. Birk, Barbara Tillmann, and Jamshed J. Bharucha. Online detection of tonal pop-out in modulating contexts. *Music Perception*, 20(3):283–305, 2003.
- Jimi Hendrix Experience. *Axis: Bold as love*. Track Records, UK, 1967.
- Neil C. Jones and Pavel A. Pevzner. *An introduction to bioinformatics algorithms*. MIT Press, Cambridge, MA, 2004.

-
- Maurice G. Kendall. *Rank correlation methods*. C. Griffin, London, UK, 1948.
- Anssi Klapuri and Manuel Davy, editors. *Signal processing methods for music transcription*. Springer-Verlag, New York, 2006.
- Ian Knopke and Frauke Jürgensen. A system for identifying common melodic phrases in the masses of palestrina. *Journal of New Music Research*, 38(2):171–181, 2009.
- Erwin Kreyszig. *Advanced engineering mathematics*. John Wiley & Sons, Hoboken, NJ, 8th edition, 1999.
- Carol L. Krumhansl and Roger N. Shepard. Quantification of the hierarchy of tonal functions within a diatonic context. *Journal of Experimental Psychology: Human Perception and Performance*, 5(4):579–594, 1979.
- Oliver Lartillot. Efficient extraction of closed motivic patterns in multi-dimensional symbolic representations of music. In Joshua D. Reiss and Geraint A. Wiggins, editors, *Proceedings of the International Symposium on Music Information Retrieval*, pages 191–198, London, UK, 2005. International Society for Music Information Retrieval. Retrieved 8 February, 2010 from <http://ismir2005.ismir.net/proceedings/1082.pdf>.
- Olivier Lartillot. A musical pattern discovery system founded on a modeling of listening strategies. *Computer Music Journal*, 28(3):53–67, 2004.
- Fred Lerdahl and Ray Jackendoff. *A generative theory of tonal music*. MIT Press, Cambridge, MA, 1983.
- Fred Lerdahl and Carol L. Krumhansl. Modeling tonal tension. *Music Perception*, 24(4):329–366, 2007.
- David Lewin. *Generalized interval systems and transformations*. Oxford University Press, Oxford, UK, 2007. (Original work published 1987 by Yale University Press, New Haven).
- H. Christopher Longuet-Higgins. The perception of melodies. *Nature*, 263(5579):646–653, 1976.
- Gareth Loy. *Musimathics: the mathematical foundations of music*, volume 1. MIT Press, Cambridge, MA, 2005.
- Daniel Lunn. Part A: Statistics lecture notes. 2007a. Retrieved 15 July, 2010 from http://www.stats.ox.ac.uk/~dlunn/A5_06/A5_2006.htm.

- Daniel Lunn. Part B: Applied statistics lecture notes. 2007b. Retrieved 15 July, 2010 from http://www.stats.ox.ac.uk/~dlunn/BS1_05/BS1_mt05.htm.
- Hugh Macdonald. Transformation, thematic. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 25, page 694. Macmillan, London, UK, 2nd edition, 2001.
- Christopher D. Manning and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, 1999.
- Andrei A. Markov. Extension of the limit theorems of probability theory to a sum of variables connected in a chain. In Ronald A. Howard, editor, *Dynamic probabilistic systems*, volume 1, pages 551–76. John Wiley & Sons, New York, 1976. (Translated by George Petelin. Original work published 1907).
- Rudolf Mathon and Alexander Rosa. 2-(v, k, λ) designs of small order. In Charles J. Colbourn and Jeffrey H. Dinitz, editors, *The CRC handbook of combinatorial designs*, pages 3–41. CRC Press, Boca Raton, FL, 1996.
- Colin Meek and William P. Birmingham. Automatic thematic extractor. *Journal of Intelligent Information Systems*, 21(1):9–33, 2003.
- David Meredith. The *ps13* pitch spelling algorithm. *Journal of New Music Research*, 35(2):121–159, 2006a.
- David Meredith. Point-set algorithms for pattern discovery and pattern matching in music. In Tim Crawford and Remco C. Veltkamp, editors, *Proceedings of the Dagstuhl Seminar on Content-Based Retrieval*, Dagstuhl, Germany, 2006b. IBFI. Retrieved 10 August, 2009 from <http://drops.dagstuhl.de/opus/volltexte/2006/652/pdf/06171.MeredithDavid.Paper.652.pdf>.
- David Meredith, Kjell Lemström, and Geraint A. Wiggins. Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music. *Journal of New Music Research*, 31(4):321–345, 2002.
- David Meredith, Kjell Lemström, and Geraint A. Wiggins. Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music. In *Cambridge Music Processing Colloquium*, Cambridge, UK, 2003. Department of Engineering, University of Cambridge. Retrieved 10 August, 2009 from <http://www.titanmusic.com/papers/public/cmpc2003.pdf>.

- Leonard B. Meyer. *Emotion and meaning in music*. University of Chicago Press, Chicago, IL, 1956.
- Tom M. Mitchell. *Machine learning*. McGraw-Hill, New York, 1997.
- Sanjit K. Mitra. *Digital signal processing: A computer-based approach*. McGraw-Hill, New York, 4th edition, 2010.
- David C. Moffat and Martin Kelly. An investigation into people's bias against computational creativity in music composition. In *Proceedings of the Joint Workshop on Computational Creativity*, Riva del Garda, 2006. European Coordinating Committee for Artificial Intelligence.
- Eugene Narmour. *The analysis and cognition of basic melodic structures: the implication-realization model*. University of Chicago Press, Chicago, IL, 1990.
- William S. Newman. Kirnberger's 'method for tossing off sonatas'. *The Musical Quarterly*, 47(4):517–525, October 1961.
- William S. Newman. *The sonata since Beethoven: The third and final volume of a history of the sonata idea*. University of North Carolina Press, Chapel Hill, NC, 1969. (Original quotation of Robert Schumann appears in *Neue Zeitschrift für Musik*, 14(10):39-40, February 1841).
- Gerhard Nierhaus. *Algorithmic composition: Paradigms of automated music generation*. Springer-Verlag, Vienna, 2009.
- James R. Norris. *Markov chains*. Cambridge University Press, Cambridge, UK, 1997.
- Jum C. Nunnally. *Psychometric theory*. McGraw-Hill, New York, 1967.
- Adam Ockelford. *Repetition in music: Theoretical and metatheoretical perspectives*. RMA Monographs No. 13. Ashgate, Aldershot, UK, 2005.
- François Pachet. Interacting with a musical learning system: The continuator. In Christina Anagnostopoulou, Miguel Ferrand, and Alan Smaill, editors, *Music and Artificial Intelligence: Proceedings of the International Conference on Music and Artificial Intelligence*, volume 2445 of *Lecture Notes in Artificial Intelligence*, pages 119–132. Springer-Verlag, Berlin, 2002.
- Ignacy J. Paderewski, editor. *Fryderyk Chopin: Complete works*, volume 10. Instytut Fryderyka Chopina, Warsaw, 1953.

- Ted Painter and Andreas Spanias. Perceptual coding of digital audio. *Proceedings of the IEEE*, 88(4):451–515, 2000.
- Bryan Pardo and William P. Birmingham. Algorithms for chordal analysis. *Computer Music Journal*, 26(2):27–49, 2002.
- Marcus T. Pearce. *The construction and evaluation of statistical models of melodic structure in music perception and composition*. PhD thesis, Department of Computing, City University, London, 2005.
- Marcus T. Pearce and Geraint A. Wiggins. Towards a framework for the evaluation of machine compositions. In *Proceedings of the AISB Symposium on Artificial Intelligence and Creativity in Arts and Sciences*, pages 22–32, Brighton, UK, 2001. SSAISB. Retrieved 24 April, 2009 from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.3026>.
- Marcus T. Pearce and Geraint A. Wiggins. Evaluating cognitive models of musical composition. In Amílcar Cardoso and Geraint A. Wiggins, editors, *Proceedings of the International Joint Workshop on Computational Creativity*, pages 73–80, London, UK, 2007. Goldsmiths, University of London.
- Marcus T. Pearce, David Meredith, and Geraint A. Wiggins. Motivations and methodologies for automation of the compositional process. *Musicæ Scientiae*, 6(2):119–147, 2002.
- Ted Pedersen. Empiricism is not a matter of faith. *Computational Linguistics*, 34(3):465–470, 2008.
- Geoffroy Peeters. Sequence representation of music structure using higher-order similarity matrix and maximum-likelihood approach. In Simon Dixon, David Bainbridge, and Rainer Typke, editors, *Proceedings of the International Symposium on Music Information Retrieval*, pages 35–40, Vienna, 2007. International Society for Music Information Retrieval. Retrieved 9 February, 2010 from http://ismir2007.ismir.net/proceedings/ISMIR2007_p035_peeters.pdf.
- Stephen Travis Pope. The SmOKE music representation, description language, and interchange format. In *Proceedings of the International Computer Music Conference*, pages 106–109, San Francisco, CA, 1992. International Computer Music Association. Retrieved 17 September, 2011 from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.63.9211>.

-
- Anthony Pople. Modeling musical structure. In Eric Clarke and Nicholas Cook, editors, *Empirical musicology: Aims methods, prospects*, pages 127–156. Oxford University Press, Oxford, UK, 2004.
- Keith Potter, Geraint A. Wiggins, and Marcus T. Pearce. Towards greater objectivity in music theory: information-dynamic analysis of minimalist music. *Musicae Scientiae*, 11(2):295–324, 2007.
- Lawrence R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.
- Ana Rebelo, Guilherme Capela, and Jamie S. Cardoso. Optical recognition of music symbols. *International Journal on Document Analysis and Recognition*, 13(1):19–31, 2010.
- Recordare LLC. *MusicXML 2.0 tutorial*. Recordare LLC, Los Altos, CA, 2010. Retrieved October 1, 2010 from <http://www.recordare.com/sites/default/files/musicxml-tutorial.pdf>.
- Nancy B. Reich. Schumann, Clara. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 22, pages 754–758. Macmillan, London, UK, 2nd edition, 2001.
- Xiaona Ren, Lloyd A. Smith, and Richard A. Medina. Discovery of retrograde and inverted themes for indexing musical scores. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, pages 252–253, New York, 2004. ACM. Retrieved 9 February, 2010 from <http://www.computer.org/portal/web/csdl/doi/10.1109/JCDL.2004.1336131>.
- John Rink. Tonal architecture in the early music. In Jim Samson, editor, *The Cambridge companion to Chopin*, pages 78–97. Cambridge University Press, Cambridge, UK, 1992.
- Curtis Roads. *The computer music tutorial*. MIT Press, Cambridge, MA, 1996.
- Pierre-Yves Rolland. FlExPat: Flexible extraction of sequential patterns. In Nick Cercone, Tsau Young Lin, and Xindong Wu, editors, *Proceedings of the IEEE International Conference on Data Mining*, pages 481–488, Washington, DC, 2001. IEEE Computer Society. Retrieved 9 February, 2010 from http://www.hemerafinanceconcept.fr/Documents/Rolland_ICDM'01.pdf.

- Christian André Romming and Eleanor Selfridge-Field. Algorithms for polyphonic music retrieval: The Hausdorff metric and geometric hashing. In Simon Dixon, David Bainbridge, and Rainer Typke, editors, *Proceedings of the International Symposium on Music Information Retrieval*, pages 457–462, Vienna, 2007. International Society for Music Information Retrieval. Retrieved 6 May, 2009 from http://ismir2007.ismir.net/proceedings/ISMIR2007_p457_romming.pdf.
- Charles Rosen. *The Romantic generation*. Harvard University Press, Cambridge, MA, 1995.
- Sheldon Ross. *A first course in probability*. Pearson Education, Inc, Upper Saddle River, NJ, 7th edition, 2006.
- Francis Rumsey and Tim McCormick. *Sound and recording: An introduction*. Focal Press, Oxford, UK, 6th edition, 2009.
- Craig Stuart Sapp. Visual hierarchical key analysis. *ACM Computers in Entertainment*, 3(4), 2005.
- Michael A. Savageau. Michaelis-menten mechanism reconsidered: implications of fractal kinetics. *Journal of Theoretical Biology*, 176:115–124, 1995.
- E. Glen Schellenberg. Simplifying the implication-realization model of melodic expectancy. *Music Perception*, 14(3):295–318, 1997.
- Heinrich Schenker. *Harmony*. MIT Press, Cambridge, MA, 1973. (Translated by Elisabeth Mann Borgese and edited by Oswald Jonas. Original work published 1906 by Cotta, Stuttgart).
- Heinrich Schenker. *Free composition*. Longman, New York, 1979. (Translated and edited by Ernst Oster. Original work published 1935 by Universal Edition, Vienna).
- Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.
- Robert Sherlaw Johnson. Composing with fractals. In John Fauvel, Raymond Flood, and Robin J. Wilson, editors, *Music and mathematics: From Pythagoras to fractals*, pages 163–172. Oxford University Press, Oxford, UK, 2003.
- Herbert A. Simon and Richard K. Sumner. Pattern in music. In Benjamin Kleinmuntz, editor, *Formal representation of human judgement*, pages 219–250. John Wiley & Sons, New York, 1968.

- Neta Spiro, Nicolas Gold, and John Rink. Plus ça change: Analyzing performances of chopin's mazurka op. 24 no. 2. In Ken'ichi Miyazaki, Yuzuru Hiraga, Mayumi Adachi, Yoshitaka Nakajima, and Minoru Tsuzaki, editors, *Proceedings of the International Conference on Music Perception and Cognition*, pages 418–427, Sapporo, Japan, 2008. JSMPC. Retrieved 27 May, 2009 from http://www.cs.ucl.ac.uk/staff/N.Gold/Nicolas_Gold/Dr_Nicolas_Gold_files/icmpc08.pdf.
- Kurt Stone. *Music notation in the twentieth century: A practical guidebook*. W. W. Norton & Company, New York, 1980.
- Siu-Lan Tan, Matthew P. Spackman, and Christy L. Peaslee. The processing of structured and unstructured tonal sequences. *Perception and Psychophysics*, 28(5):381–389, 1980.
- David Temperley. Bayesian models of musical structure and cognition. *Musicae Scientiae*, 8(2):175–205, 2004.
- David Temperley. *Music and probability*. MIT Press, Cambridge, MA, 2007.
- David Temperley. A unified probabilistic model for polyphonic music analysis. *Journal of New Music Research*, 38(1):3–18, 2009.
- William F. Thompson and Lola L. Cuddy. Sensitivity to key change in chorale sequences: a comparison of single voices and four-voice harmony. *Music Perception*, 7(2):151–168, 1989.
- Barbara Tillmann, Petr Janata, Jeffrey L. Birk, and Jamshed J. Bharucha. Tonal centers and expectancy: Facilitation or inhibition of chords at the top of the harmonic hierarchy. *Journal of Experimental Psychology: Perception and Performance*, 34(4):1031–1043, 2008.
- Laurel J. Trainor and Robert J. Zatorre. The neurological basis of musical expectations. In Susan Hallam, Ian Cross, Michael Thaut, and Aniruddh D. Patel, editors, *The Oxford handbook of music psychology*, pages 171–183. Oxford University Press, Oxford, UK, 2009.
- Dmitri Tymoczko. *A geometry of music: Harmony and counterpoint in the extended common practice*. Oxford Studies in Music Theory. Oxford University Press, Oxford, UK, 2011.
- Esko Ukkonen, Kjell Lemström, and Veli Mäkinen. Geometric algorithms for transposition invariant content-based music retrieval. In Holger H.

- Hoos and David Bainbridge, editors, *Proceedings of the International Symposium on Music Information Retrieval*, pages 193–199, Baltimore, MD, 2003. International Society for Music Information Retrieval. Retrieved 15 February, 2010 from <http://ismir2003.ismir.net/papers/Ukkonen.pdf>.
- Anja Volk. The study of syncopation using inner metric analysis: Linking theoretical and experimental analysis of metre in music. *Journal of New Music Research*, 37(4):259–273, 2008.
- Christian von Ehrenfels. On Gestalt qualities. In Barry Smith, editor, *Foundations of Gestalt theory*, pages 82–117. Philosophia Verlag, Munich, 1988. (Original work published 1890).
- Paul von Hippel. Redefining pitch proximity: Tessitura and mobility as constraints on melodic intervals. *Music Perception*, 17(3):315–327, 2000.
- Paul M. Walker. Exposition. In Stanley Sadie and John Tyrrell, editors, *The new Grove dictionary of music and musicians*, volume 8, pages 462–463. Macmillan, London, UK, 2nd edition, 2001.
- Raymond Whorley, Geraint A. Wiggins, Christophe Rhodes, and Marcus T. Pearce. Development of techniques for the computational modelling of harmony. In Dan Ventura, Alison Pease, Rafael Pérrez y Pérdez, Graeme Ritchie, and Tony Veale, editors, *Proceedings of the International Conference on Computational Creativity*, pages 11–15, Lisbon, 2010. University of Coimbra.
- Gerhard Widmer and Werner Goebl. Computational models of expressive music performance: the state of the art. *Journal of New Music Research*, 33(3):203–216, 2004.
- Geraint Wiggins, Eduardo R. Miranda, Alan Smaill, and Mitch Harris. A framework for the evaluation of music representation systems. *Computer Music Journal*, 17(3):31–42, 1993.
- Geraint A. Wiggins. A preliminary framework for description, analysis and comparison of creative systems. *Knowledge-Based Systems*, 19(7):449–458, 2006.
- Geraint A. Wiggins. Models of musical similarity. *Musicae Scientiae*, Discussion Forum 4A:315–338, 2007.
- Geraint A. Wiggins. Computer models of musical creativity: A review of computer models of musical creativity by David Cope. *Literary and Linguistic Computing*, 23(1):109–116, 2008.

- Geraint A. Wiggins, Mitch Harris, and Alan Smaill. Representing music for analysis and composition. In *Proceedings of the IJCAI Workshop on Music and Artificial Intelligence*, pages 63–71, Detroit, MI, 1989.
- Geraint A. Wiggins, Kjell Lemström, and David Meredith. SIA(M)ESE: an algorithm for transposition invariant, polyphonic content-based music retrieval. In *Proceedings of the International Symposium on Music Information Retrieval*, pages 283–284, Paris, 2002. International Society for Music Information Retrieval. Retrieved 29 October, 2009 from <http://ismir2002.ismir.net/proceedings/03-SP03-4.pdf>.
- Robin J. Wilson. *Introduction to graph theory*. Longman, Harlow, UK, 4th edition, 1996.
- Craig Wright. *Music and ceremony at Notre-Dame of Paris 500-1500*. Cambridge University Press, Cambridge, UK, 1989.

