Using Statistical Feature Extraction to Distinguish the Styles of Different Composers

Cory McKay (Marianopolis College)
Tristano Tenaglia (McGill University)
Julie Cumming (McGill University)
Ichiro Fujinaga (McGill University)

Topics

- Introduction to “features” (from a machine learning perspective)
  - And how they can be useful for musicologists and music theorists
- jSymbolic2
  - And how it can be useful to music theorists and musicologists
- Composer attribution study

Personal context

- I was originally trained as a physicist and as a jazz guitarist before changing careers and focusing on music information retrieval
- As a former physicist, I am deeply attached to:
  - Overarching abstract theoretical models
  - Empirical validation of those models
- I think we do a great job at the first of these in music theory and musicology
  - But there is still room for improvement with respect to the second

Empiricism, software & statistics

- Empiricism, automated software tools and statistical analysis techniques allow us to:
  - Study huge quantities of music very quickly
    - More than any human could reasonably look at
  - Empirically validate (or repudiate) our theoretical suspicions
  - Do purely exploratory studies of music
  - See music from fresh perspectives
    - Can inspire new ways of looking at music

Human involvement is crucial

- Of course, computers certainly cannot replace the expertise and insight of musicologists and theorists
  - Computers instead serve as powerful tools and assistants that allow us to greatly expand the scope and reliability of our work
- Computers do not understand musical experience
  - We must pose the research questions for them to investigate
  - We must interpret the results they present us with
- Music is, after all, defined by human experience, not some “objective” externality

jMIR

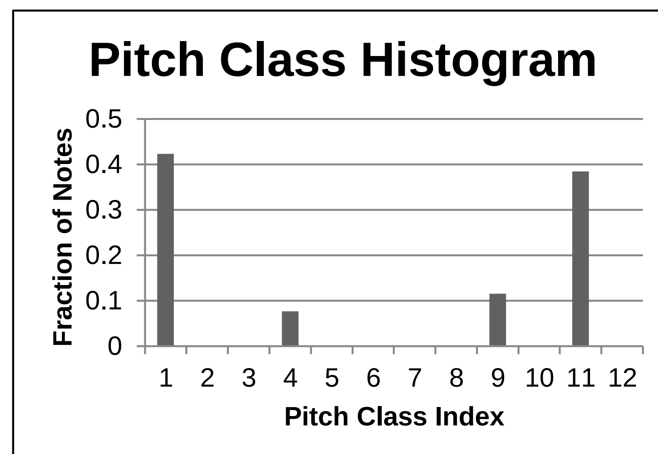
What are “features”?

- Pieces of information that can characterize something (e.g. a piece of music) in a simple way
- Usually numerical values
  - A feature can be a single value, or it can be a set of related values (e.g. a histogram)
- Can be extracted from pieces as a whole, or from segments of pieces

Example: Two basic features

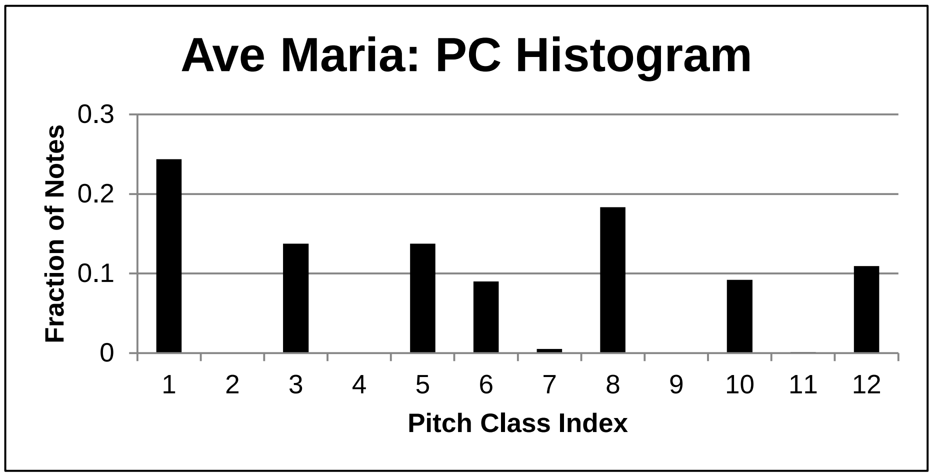
- Range (1-D): Difference in semitones between the highest and lowest pitches.
- Pitch Class Histogram (12-D): Each of its 12 values represents the fraction of notes with a particular pitch class. The first value corresponds to the most common pitch class, and each following value to a pitch class a semitone higher than the previous.

- Range = G - C = 7 semitones
- Pitch Class Histogram: see graph ->
  - Note counts: C: 3, D: 10, E: 11, G: 2
  - Most common note: E (11/26 notes)
    - Corresponding to 0.423 of the notes
  - E is thus pitch class 1, G is pitch class 4, C is pitch class 9, D is pitch class 11



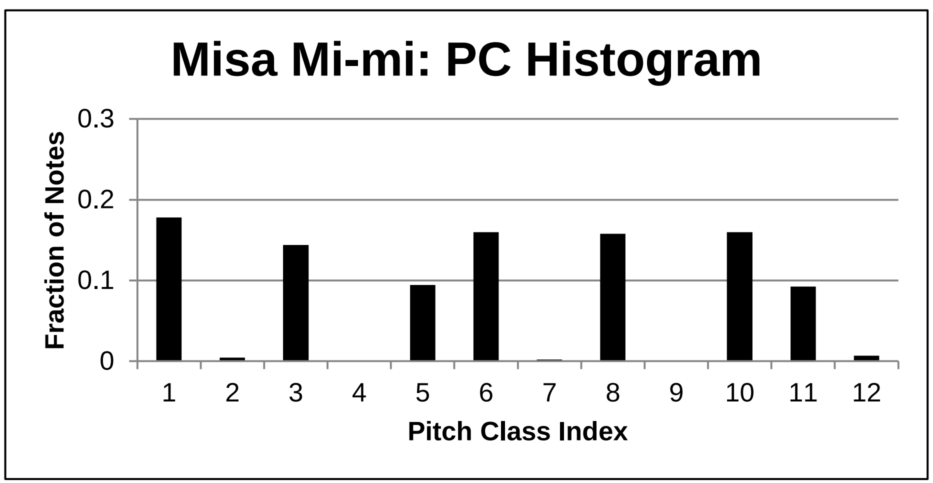
Josquin's *Ave Maria... Virgo serena*

- Range: 34
- Repeated notes: 0.181
- Vertical perfect 4ths: 0.070
- Rhythmic variability: 0.032
- Parallel motion: 0.039



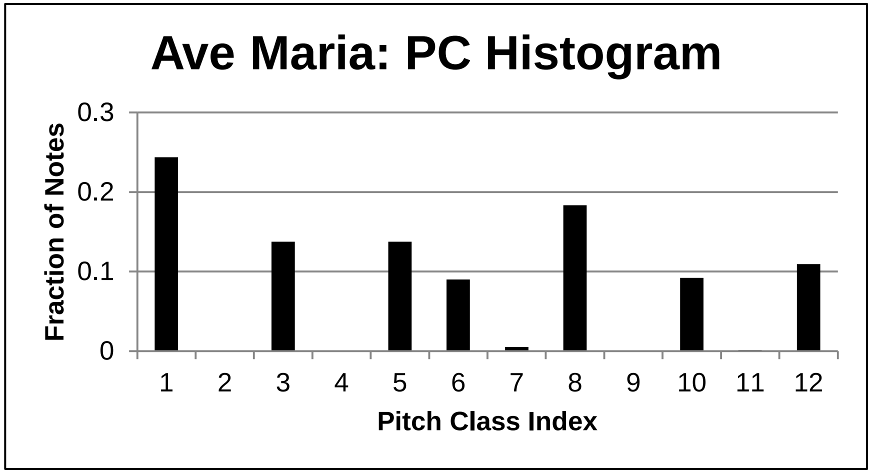
Ockeghem’s *Missa Mi-mi* (Kyrie)

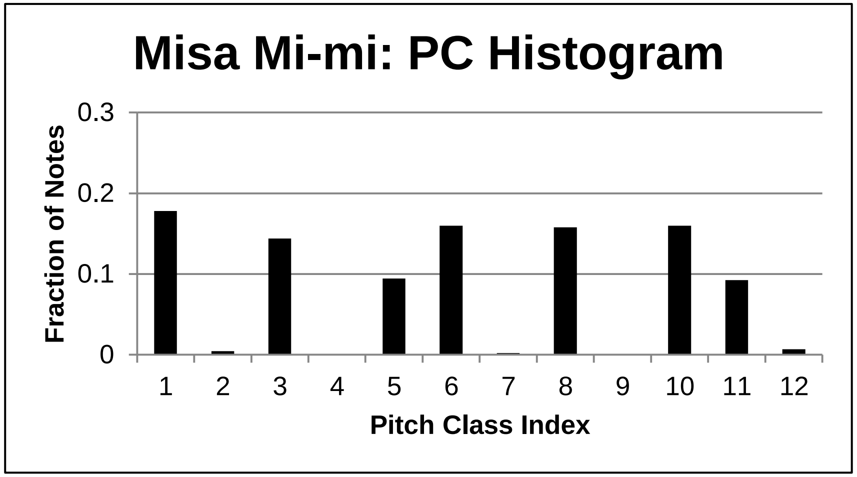
- Range: 26
- Repeated notes: 0.084
- Vertical perfect 4ths: 0.109
- Rhythmic variability: 0.042
- Parallel motion: 0.076



Feature value comparison

Feature	Ave Maria	Misa Mi-mi
Range	34	26
Repeated notes	0.181	0.084
Vertical perfect 4ths	0.070	0.109
Rhythmic variability	0.032	0.042
Parallel motion	0.039	0.076





How can we use features?

- Perform sophisticated searches of large musical databases
  - e.g. find all pieces with less than X amount of chromaticism and more than Y amount of contrary motion
- Use machine learning to classify or cluster music
  - e.g. identify the composers of unattributed musical pieces
- Apply statistical analysis and visualization tools to features extracted from large collections of music
  - Look for patterns
  - Study the relative musical importance of various features

jSymbolic2: Introduction

- jSymbolic2 is a software platform we have implemented for extracting features from symbolic music
  - Part of our much larger jMIR package
- Compatible with Macs, PCs and Linux computers

What does jSymbolic2 do?

- Extracts 172 unique features
- Some of these are multi-dimensional histograms, including:
  - Pitch and pitch class histograms
  - Melodic interval histograms
  - Vertical interval histograms
  - Chord types histograms
  - Beat histograms
  - Instrument histograms
- In all, extracts a total of 1230 separate values

jSymbolic2: Feature types (1/2)

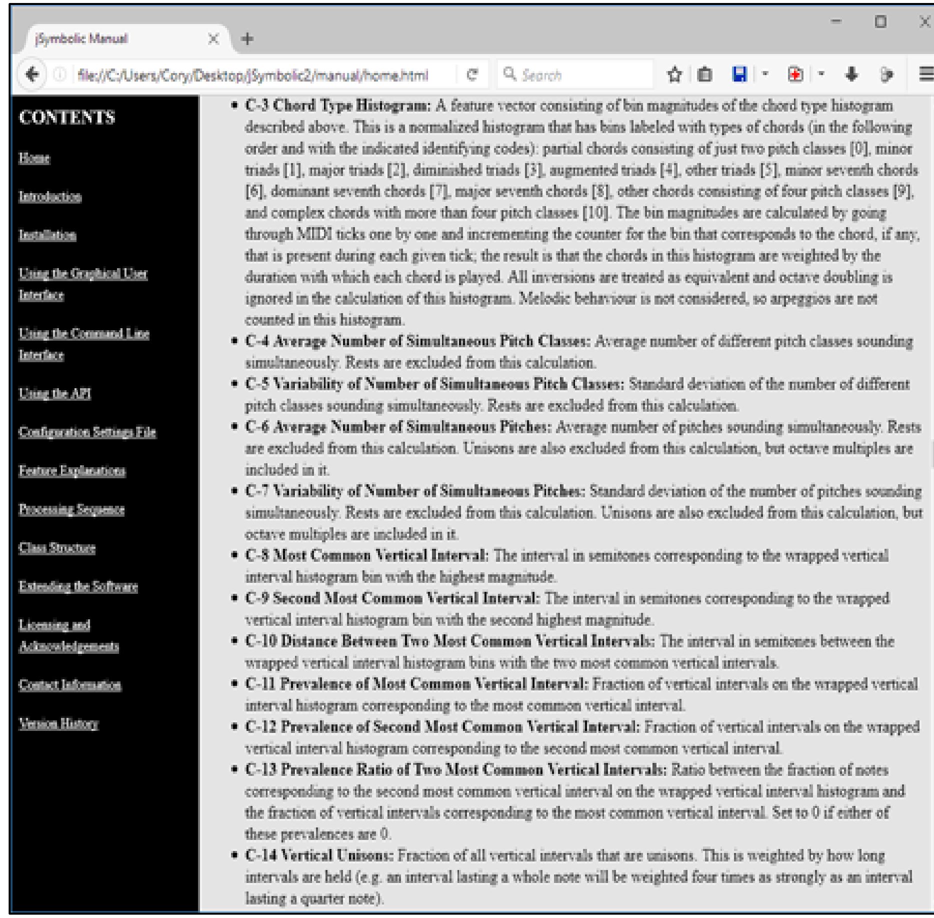
- Pitch Statistics:
  - What are the occurrence rates of different pitches and pitch classes?
  - How tonal is the piece?
  - How much variety in pitch is there?
- Melody / horizontal intervals:
  - What kinds of melodic intervals are present?
  - How much melodic variation is there?
  - What kinds of melodic contours are used?
  - What types of phrases are used?
- Chords / vertical intervals:
  - What vertical intervals are present?
  - What types of chords do they represent?
  - How much harmonic movement is there?

jSymbolic2: Feature types (2/2)

- Instrumentation:
  - What types of instruments are present and which are given particular importance relative to others?
- Texture:
  - How many independent voices are there and how do they interact (e.g., polyphonic, homophonic, etc.)?
- Rhythm:
  - Time intervals between the attacks of different notes
  - Duration of notes
  - What kinds of meters and rhythmic patterns are present?
  - Rubato?
- Dynamics:
  - How loud are notes and what kinds of dynamic variations occur?

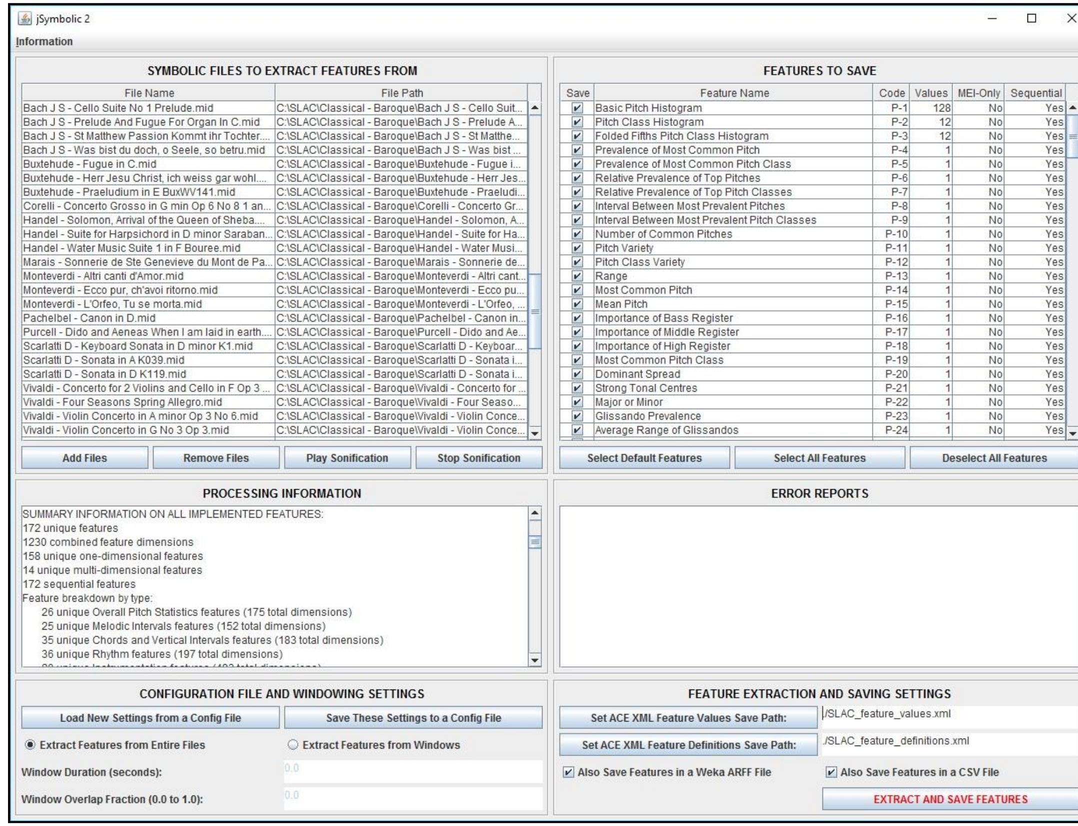
jSymbolic2: Manual

- Extensive manual includes:
  - Detailed feature descriptions
  - Detailed instructions on installation and use



jSymbolic2: User interfaces

- Graphical user interface
- Command line interface
- Java API
- Rodan workflow



What can you do with jSymbolic2's features?

- Empirically study huge collections of music in new ways
  - Search music databases based on feature values
  - Use machine learning
  - Analyze and visualize music based on feature values

Composer attribution study

- We used jSymbolic2 features to automatically classify pieces of Renaissance music by composer
  - As an example of the kinds of things that can be done with jSymbolic2
  - As a meaningful research project in its own right

RenComp7 dataset

- Began by constructing our "RenComp7" dataset:
  - 1584 MIDI pieces
  - By 7 Renaissance composers
- Combines:
  - Top right: Music drawn from the Josquin Research Project (Rodin, Sapp and Bokulich)
  - Bottom right: Music by Palestrina (Miller 2004) and Victoria (Sigler, Wild and Handelman 2015)

Composer	Pieces
Busnoys	69
Josquin *(only includes the 2 most secure Jesse Rodin groups)*	131
La Rue	197
Martini	123
Ockeghem	98

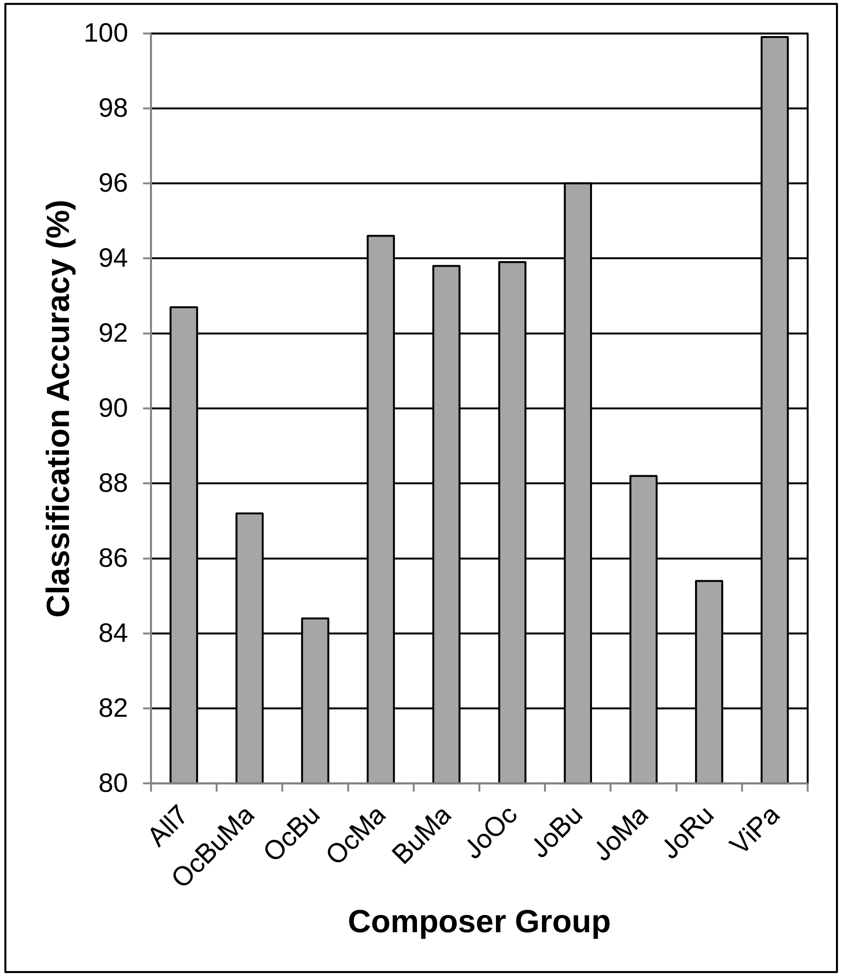
Composer	Pieces
Palestrina	705
Victoria	261

Methodology

- Extracted 721 feature values from each of the 1584 RenComp7 pieces using jSymbolic2
- Used machine learning to teach a classifier to automatically distinguish the music of the composers
  - Based on the jSymbolic2 features
- Used statistical analysis to gain insight into relative compositional styles
- Performed several versions of this study
  - Classifying amongst all 7 composers
  - Focusing only on smaller subsets of composers
    - Some more similar, some less similar

Classification results

Composer Group	Classification Accuracy
All 7	92.7%
Ockeghem / Busnoys / Martini	87.2%
Ockeghem / Busnoys	84.4%
Ockeghem / Martini	94.6%
Busnoys / Martini	93.8%
Josquin / Ockeghem	93.9%
Josquin / Busnoys	96.0%
Josquin / Martini	88.2%
Josquin / La Rue	85.4%
Victoria / Palestrina	99.9%



Direct applications of such work

- Validating existing suspected but uncertain attributions
- Helping to resolve conflicting attributions
- Suggesting possible attributions of currently unattributed scores

How do the composers differ?

- Some interesting questions:
  - What musical insights can we learn from the jSymbolic2 feature data itself?
  - In particular, what can we learn about how the music of the various composers differ from one another?
- Chose to focus on two particular cases:
  - Josquin vs. Ockeghem: Relatively different
  - Josquin vs. La Rue: Relatively similar

A priori expectations (1/3)

- What might an expert musicologist expect to differentiate the composers?
  - Before actually examining the feature values
- Once formulating these expectations, we can then see if the feature data confirms or repudiates these expectations
  - Both are useful!
- We can also then see if the feature data reveals unexpected insights

A priori expectations (2/3)

- What do you think might distinguish the composers?
  - Josquin vs. Ockeghem?
  - Josquin vs. La Rue?
- I consulted one musicologist (Julie Cumming) and one theorist (Peter Schubert), both experts in the period . . .

A priori expectations (3/3)

- Josquin vs. Ockeghem: Ockeghem may have . . .
  - Slightly more large leaps (larger than a 5th)
  - Less stepwise motion in some voices
  - More notes at the bottom of the range
  - Slightly more chords (or simultaneities) without a third
  - Slightly more dissonance
  - A lot more triple meter
  - More varied rhythmic note values
  - More 3-voice music
  - Less music for more than 4 voices
- Josquin vs. La Rue: La Rue may have . . . Hard to say!
  - Maybe more varied repetition (melodic and contrapuntal, including rhythm)?
  - Maybe more compressed ranges?

Were our expectations correct?

- Josquin vs. Ockeghem: Ockeghem may have . . .
  - OPPOSITE: Slightly more large leaps (larger than a 5^{th})
  - SAME: Less stepwise motion in some voices
  - SAME: More notes at the bottom of the range
  - SAME: Slightly more chords (or simultaneities) without a third
  - OPPOSITE: Slightly more dissonance
  - YES: A lot more triple meter
  - SAME: More varied rhythmic note values
  - YES: More 3-voice music
  - YES: Less music for more than 4 voices
- Josquin vs. La Rue: La Rue may have . . .
  - UNKNOWN: Maybe more varied repetition (melodic and contrapuntal, including rhythm)?
  - SAME: Maybe more compressed ranges?

Diving into the feature values

- There are a variety of statistical techniques for attempting to evaluate which features are likely to be effective in distinguishing between types of music
- We used seven of these statistical techniques to find:
  - The features and feature subsets most consistently statistically predicted to be effective at distinguishing composers
- We then manually examined these feature subsets to find the features likely to be the most musicologically meaningful

Novel insights revealed (1/2)

- Josquin vs. Ockeghem (93.9%):
  - Rhythm-related features are particularly important
    - Josquin tends to have greater rhythmic variety
      - Especially in terms of both especially short and long notes
    - Ockeghem tends to have more triple meter
      - As expected
    - Features derived from beat histograms also have good discriminatory power
  - Ockeghem tends to have more vertical sixths
  - Ockeghem tends to have more diminished triads
  - Ockeghems tends to have longer melodic arcs

Novel insights revealed (2/2)

- Josquin vs. La Rue (85.4%):
  - Pitch-related features are particularly important
    - Josquin tends to have more vertical unisons and thirds
    - La Rue tends to have more vertical fourths and octaves
    - Josquin tends to have more melodic octaves

Research potential (1/2)

- The results above are the product of an initial accurate but relatively simple analysis
- There is substantial potential to expand this study
  - Apply more sophisticated and detailed statistical analysis techniques
  - Perform a more detailed manual exploration of the feature data
  - Implement new specialized features
  - Look at more and different composer groups

Research potential (2/2)

- Composer attribution is just one small example of the many musicological and theoretical research domains to which features and jSymbolic2 can be applied
  - e.g. genre, such as madrigals vs. motets

Research collaborations (1/2)

- We enthusiastically welcome research collaborations with other musicologists and theorists
- In particular, we are always looking for ideas for interesting for new features to implement
  - jSymbolic2 makes it relatively easy to add bespoke features
  - Can iteratively build increasingly complex features based on existing features

Research collaborations (2/2)

- Please do not hesitate to speak to me if you would like demos of:
  - Using jSymbolic2
  - How one can apply statistical analysis or machine learning to extracted features
  - How feature values can be visualized and explored manually
- I am also more than happy to show you any of our data or code
  - jSymbolic2 is open-source and free

Thanks for your attention!

- **jSymbolic2:** http://jmir.sourceforge.net
- **E-mail:** cory.mckay@mail.mcgill.ca

McGill

Schulich School of Music
École de musique Schulich

CIRMMT
Centre for Interdisciplinary Research in Music Media and Technology

Social Sciences and Humanities Research Council of Canada

Conseil de recherches en sciences humaines du Canada

Canada

Marianopolis College

Fonds de recherche sur la société et la culture Québec

DDMAL
Distributed Digital Music Archives & Libraries Lab

SIMSSA
Single Interface for Music Score Searching and Analysis

jMIR