Non-contextual musical analysis

Polansky, Rockmore, Shapira

with Will Haslett

June, 2006

Precedents
- Multi-dimensional scaling of timbre space (Grey, Krumhansl, etc.)
- Statisical measures of chromaticism (atonal theory)
- Style analysis and synthesis (Cope, others)
- Long history of statistically based computure-aided composition
- Recent work in Zipf's law analysis-, compression- (relative entropy) based similarity measures.

Purpose
- Forensic vs. non-forensic analysis
- Stylistically non-biased (ignorant) feature detection
- What does that say about musical style?
- What are the most important, or most "telling" non-stylistically based features?
- What are musical "glue" words?
- Always stop short of higher-level musical assumptions (themes, motives, "tension-release," "harmony")
- Several goals: evolution of a composers' style, comparison of one composer to another, comparison of one style to another

Statistical features

- A set of simple, easy to calculate, non-assumptive features that make up a multi-dimensional feature space
- The smaller the number of features, the better, and the simpler the features, the better.
  - Features should be unambiguous, simply defineable.
  - Distance between features should be as simple as possible.
- Must be easily countable, avoid invoking anything about "music theory"
  - But… even at the simplest level, this is hard to do. e.g One of our features is "pitch-class" which invokes an assumption about octave equivalance.

Methodology

- Use simplest body of data possible: laypersons's conception of "Mozart,", via MidiFiles
- Score as text (not soundfiles)
- How to compare? Problems: do we look at composers chronologically? Is early Mozart the same composer as late Mozart? Are all Mozart movements "Mozart"?, or is there a scherzo Mozart, an Allegro Mozart, etc. Do we look at averages of measures across composers, across types of pieces, across contemporaneity?
- Feature based approach
  - Extract features
  - Develop distance measures between features, and the set of features
    - Choice of metrics for each feature is individual, and somewhat arguable.
  - Use MDS or PCA or some similar measure to plot the spaces, determine nearness of stimuli to preexisting clusters
    - Weighting of different features a real problem (these measures are in no way normalized to each other)
    - Consider the issue of the clustering algorithm having its own assumptions about distance, weighting.

Some example features

- Pareto Graphs of simple statistical measures
  - Frequency ranking without paying attention, in this case, to what is being ranked
    - What is the frequency of the most common element, next most common element, etc., out to the number of elements.
    - Assumption that this will be some measure of a composer's signature
    - Trivial case: extreme atonality, should be more or less a straight line. Non-modulating tonal music: should be a simple decreasing function from tonic down to 2nd degree, tritone, etc.
  - More or less looking for the "power function," a la Zipf's Law, of the usage of elements
    - Two separate ranking graphs: PCR, PNR (see below, to start with, though there are others, see further below)
      - Possible alternative measure: Over a composer's work, or some segment of it, the mean frequency ranking for the 1st .. nth most common value, as well as the mean ranking for PC and PN
    - These graphs need to be logarithmic, or high populations at the top end of the statistics will disproportionately affect the measure between graphs.
      - The hypothesis is that to distinguish between two composers of a similar stylistic period, it is the outliers (the least populous places in the Pareto histogram) that may have the most importance.
      - Both the x and y values (or k and $p(k)$ must be log).
        - In this way, whatever the power of the power function, it will be found as a slope of a straight line.
  - *Slope*
    - $log(p(k))/log(k) = -a$
      - Determines slope of the best fit line of the logarithm of the Pareto graph prob. distribution.
  - R^2 values: measure the fit of the Pareto graph to the best power law function fit.

        - Use least-squares to measure the goodness of fit of a probability distribution to the compute slope (see above).
            - Maneris et al do this, but with averages across features.
        - Has some surprising ramifications, a composer's R^2 measures might be more related than their power laws: Schoenberg and Mozart may have more in common than Schoenberg and Berg, Mozart and Haydn (this is a measure of how well they adhere to a certain kind of regularity, or how "similar they are to themselves")
        -
- Single notes (things to count in the Pareto graphs)
    - Smallest "glue" in the piece, no relationships implied by counting single elements
        - Pitch class (PC)
        - Pitch "number" (PN)
            - Takes into account "range", but is somewhat redundant to pitch class
            - Will vary by composer technologies (e.g. pianos got bigger!). But, over a wide body of works, forms (orchestral, etc.), should normalize itself and also incorporate composer's orchestrational choices.
            - PN still has a few details to work out
        - Both PC and PN are normalized, given as a probability distribution.

Simple Experimental Procedure (so far)

- Take Pareto graphs of PC and PN as probability distributions.
- Take the log of the *p(k)* and the *log k*.
- Find the slope of the best line ($log\ p(k)/log(k) = -a$)
- Find the least squares best fit of the *p(k)* to the line with slope $–a$ (R^2)

- Inspect these graphs and values.

- Now, have a 6 feature vector:
  - (PC-vector, PN-vector, PC-slope, PN-slope, PCR^2, PNR^2)
    - Note: At this point, we are not really looking at the first two.

- Take the distance between each feature.
- Do a Euclidean metric on these 6 values, return one number (the distance between two pieces)
- Create a symmetric matrix of the differences between all pieces in the input set.
- Feed that matrix into MDS, PCA.

- Look for clustering.

- Add features.