Machine Learning Term Project Write-up
Creating Models of Performers of Chopin Mazurkas

Marcello Herreshoff
In collaboration with Craig Sapp (craig@ccrma.stanford.edu)

1 Motivation

We want to generative models of pianists based on input features extracted from musical scores (such as the number of events in a beat, the position of a beat in a phrase, dynamics, harmony, form, *etc.*) Target features are tempo values for each beat of a performance. We try to extract performer style from these models to generate synthetic performances of different compositions. These models can also potentially be used to identify the performers of new or old recordings with unknown performers.

Training data consists of tempo curves extracted from audio recordings of 358 performances of five different mazurkas composed by Frédéric Chopin played by 118 different performers. Craig has demonstrated that a performer nearly always maintains a consistent performance rendering of the *same piece* over time (even after several decades) and numerical methods based on correlation can be used to identify audio recordings of the same piece played by the same pianist.[1]

We are interested in being able to transfer the performance style of a particular performer between *different pieces* for the purpose of synthetic performance in that performer's style, or to identify a performer in a recording of unknown or disputed origin.[2] Recent work has been done on attempting to address performer style in a machine-learning context, but state-of-the-art is "still rather speculative".[3] Automatically-generated performance rendering competitions have been held at several music-related conferences in the past few years.[4]

2 Input and Target Features

Target and input features for the project consist of data for five solo piano mazurkas composed by Frédéric Chopin (1810-1849). The mazurka is a folk dance from Chopin's native country of Poland in triple meter which is generally characterized by a weak, short first beat in each measure and an accented second or third beat. Chopin converted and popularized this folk dance into an abstract musical art form. Performance conventions for playing these compositions also show a general trend over time from a dance to more abstract/personal musical interpretations. In addition, performances of mazurkas tend to vary regionally, with Polish and Russian pianists influenced by the historical dance interpretations, while pianists more geographically distant from this central tradition tend to use a more individual and abstract playing style.

2.1 Target Features

The target data consists of tempo data for each beat in various performances by professional pianists, extracted by Craig as part of the *Mazurka Project* at Royal Holloway, University of London.[5] Performance data consists of absolute timings for beats in recordings of the mazurka, as well as loudness levels at the locations of the beats (which are not utilized in the current study). The tempo data used in this project is converted into beats per minute which is inversely proportional to the duration between absolute beat timings locations:

$$\text{tempo}^{(i)} = \frac{60}{\text{beat}^{(i+1)} - \text{beat}^{(i)}}$$

[1] *Hybrid Numeric/Rank Similarity Metrics for Musical Performance Analysis*, Craig Sapp, ISMIR 2008.
`http://ismir2008.ismir.net/papers/ISMIR2008\_240.pdf`

[2] *Fantasia for Piano*, Mark Singer, The New Yorker, 17 September 2007.
`http://www.newyorker.com/reporting/2007/09/17/070917fa\_fact\_singer?currentPage=all`

[3] *In search of the Horowitz factor*, Widmer, *et al.*, AI Magazine 24/3 (Sept. 2003), furlhttp://portal.acm.org/citation.cfm?id=958680

[4] `http://www.renconmusic.org/icmpc2008/`

[5] `http://mazurka.org.uk/info/revcond`, or in Microsoft Excel format: `http://mazurka.org.uk/info/excel/beat`

Beat timings are extracted manually with the assistance of audio analysis tools, using an audio editor called *Sonic Visualiser*.[6] Automatic beat extraction is not possible with current state-of-the art methods since mazurka beat-tempos can vary by 50% between successive beats (a characteristic of the mazurka genre) and most beat-extraction methods assume a narrower variation between beats. Each mazurka performance consists of a sequence about 200–300 beat-tempos. Figure 1 shows beat-tempo curves for several performers all playing mazurka in B minor, op. 30, No. 2.

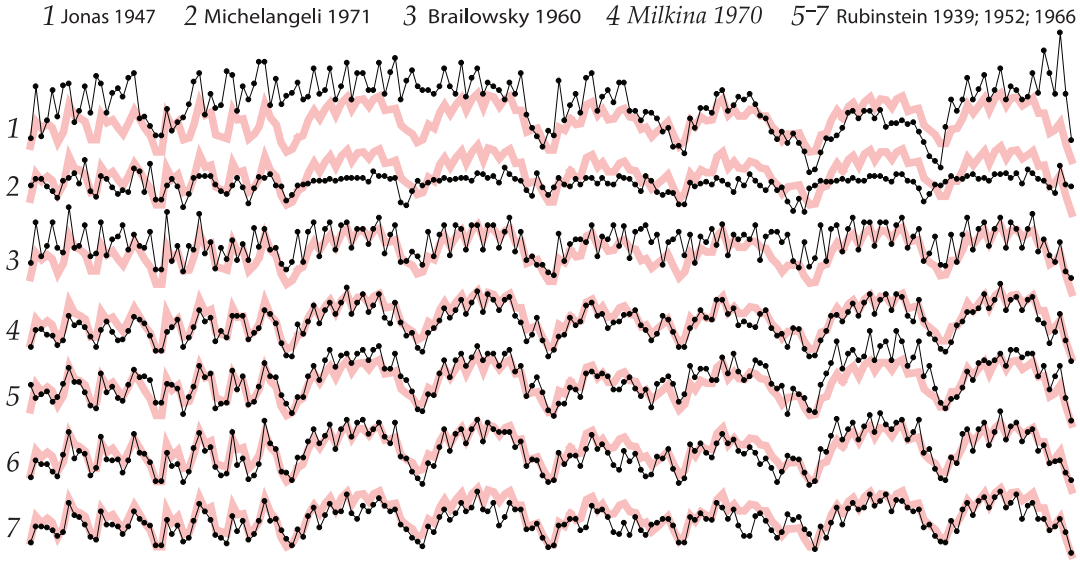

Figure 1: Six example beat-tempo curves for performances of mazurka 30/2. The light-gray curve is the average for 35 performances. Plot 1 shows a performer who concatenates phrases; plot 2 shows a performer who plays slower than average and does not do much phrase arching; plot 3 show a performer who exaggerates the metric cycle by switching between fast and slow beats; plot 4 shows someone who plays very close to the average; plots 5–7 show the same performer recorded on different dates.

Each of the five mazurkas utilized for this study have performance data for 30 to 90 performances. All mazurkas include data for three performances by Arthur Rubinstein, a well-known and prolific performer of the 20th-century, as well as occasional duplicate performers who record the same mazurka twice.

2.2 Input Features

Several input features were extracted from text-based musical scores for each mazurka.[7] We chose features which we thought would be likely to differ between different performers and might stay stable between the performances of an individual performer. The current set of features going from general to more musically specific:

1. The mean feature: This feature is always 1. We included it so that the linear regression algorithm can learn the constant offset. The theta value for this feature describes roughly the average tempo at which the performer plays.

2. The global position: This feature increases linearly as the piece progresses. The theta value for this feature describes roughly whether the performer accelerates or decelerates on average over the course of the entire piece.

3. The metrical position: This feature is the position of the beat in the measure. In this case, because all Mazurkas are in $\frac{3}{4}$ time, the position is either 1, 2 or 3. The theta value for this feature describes roughly whether the performer accelerates or decelerates inside each measure (averaged across the whole piece.)

4. The number of left hand events: This feature is the number of notes played by the performer's left hand in each beat. The theta value of this feature describes roughly whether the performer speeds up or slows down when playing beats with more ornate left hand parts.

[6] `http://www.sonicvisualiser.org`, `http://sv.mazurka.org.uk`
[7] `http://kern.ccarh.org/cgi-bin/ksbrowse?type=collection&l=/users/craig/classical/chopin/mazurka`

5. The number of right hand events (same as above.)

6. The "harmonic charge": a measurement of the local harmonic activity. The calculation method is described below. The theta value of this feature shows roughly whether the performer plays faster when the performance modulates up a fifth.

To calculate the "harmonic charge" we measure the interval between the global key of the piece and the local key of a analysis window around the current beat. The interval is described as a number of perfect-fifths between the key tonics. For example, if the global key is C major, and the local key is G major, then the harmonic charge is +1 since G major is close to C major. If the local key is B major, then the harmonic charge compared to C major is higher at +6 since it is a more distant key relation.

We calculate the local and global key measures using the Krumhansl-Schmuckler key-finding algorithm with Bellman-Budge key profiles.[8] The algorithm measures a chromatic histogram of notes in a musical selection, and then uses Pearson correlation to compare to expected prototypes for major and minor keys, taking the test key with the highest correlation as the answer:

$$\text{key} = \arg\max_k \frac{\sum_t [h(k,t) - \mu_h][p(t) - \mu_p]}{\sqrt{\sum_t [h(k,t) - \mu_h]^2 \ \sum_t [p(t) - \mu_p]^2}}$$

where h is a duration-weighted histogram of pitches in the analysis window in the music score; p is a pitch-class histogram expected for a major or minor key.

3 Linear Regression Model

Because we are trying to build an application, we decided to start out with a simple model and improve it incrementally. Our basic model states that the tempo with which a performer will play a beat is Gaussianly distributed with mean at an affine function of the absolute index of the measure containing the beat, the absolute index of the beat, the index of the beat in the measure (in this case, a number between 1 and 3, because Mazurkas have three-beat measures), the number of beats in the performer's left hand, the number of beats in the performer's right hand, and the harmonic charge.

In frequentist terms, our prediction for the performer will be an affine function of the features listed above, and our effort function will be the sum of squared errors between the prediction and the actual performance. In order to make the error output more comprehensible, we calculated the root mean squared (RMS) error, which is equivalent.

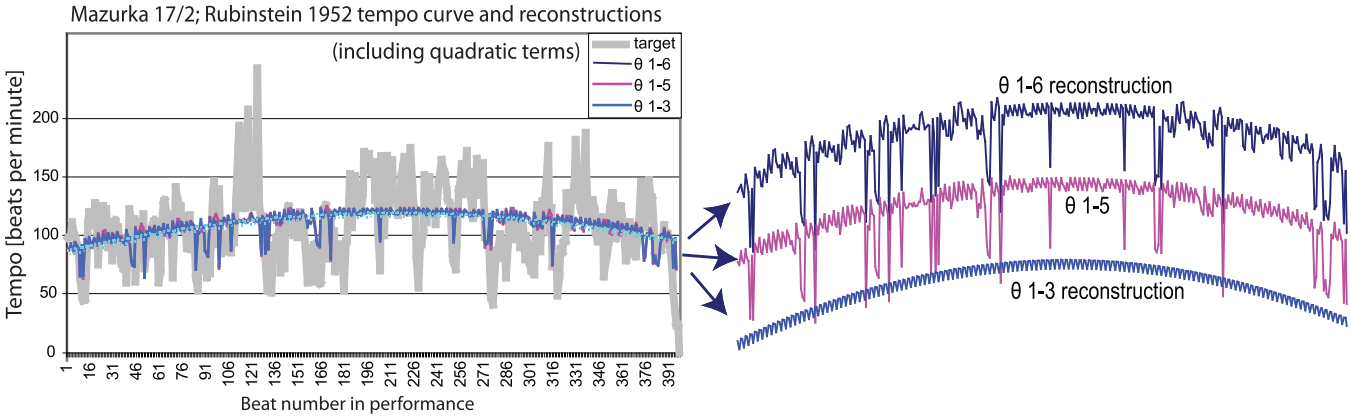

Figure 2: Three progressive reconstructions of Rubinstein's 1952 performance of Mazurka 17/4, using linear regression on the original features as well as quadratic features.

For each piece, the average of the RMS error between each recording and the average of all recordings of each piece was lower than the average of the RMS error between each recording and its reconstruction under my linear regression model. This means that the reconstructions are worse approximations to the recordings than the average recording. For example, the average RMS error for the reconstructions of mazurka 63/3 is 30.203, while the average RMS error for the average of all recordings of Mazurka 63/3 was 25.931.

[8] *Visual hierarchical key analysis*, Craig Sapp, in ACM CIE 3/4, October 2005. `http://portal.acm.org/citation.cfm?id=1095534.1095544`

Next, we did an ablative analysis. We started by stripping off all the features (except the constant) in order to get a base-line on the error. The error of this severely ablated model (which in effect approximated every recording with a flat line) produced an error which was not much higher than the error of the linear regression model which had all five features we listed as its input. For example, the average RMS error for the flat-line approximation of Mazurka 63/3 was 31.984. (On only one of the Mazurkas (Mazurka 242) was did the average RMS error for the flat-line approximation and the average RMS error for the full linear regression differ by as much as 4.5.)

This indicates not only that the algorithm is not extracting enough information from the data to be a better approximation than the average recording, but that none of these five features are not strongly correlated with the tempo data (because if they were, some values of theta would have significantly lowered the RMS error.) [9].

We have also done experiments adding quadratic terms to our existing features. For each feature $x^{(i)}$ we added another feature $x_{i+N} = (x^{(i)})^2$, the idea being that many of the stuctures in music have shapes that look like arches (see for example Figure 1). To test whether this was effective we trained both models on Rubinstein's 1952 performance of Mazurka 17/4 and we tested them on Rubinstein's 1966 performance of the same Mazurka. Adding these terms reduced the RMS error from 22.74 to 21.19 (this means that the error function, which is proportional to a the square of the RMS error, was reduced by an additional 10%.)

As a specific example consider Figure 2. This shows a progressive reconstruction of the piece, first using only the first three features, then using the first five features and then using all six features. The first reconstruction includes the global and metric position features. Here we see that it has roughly catured the tempo-arch in which Rubinstein plays the piece.

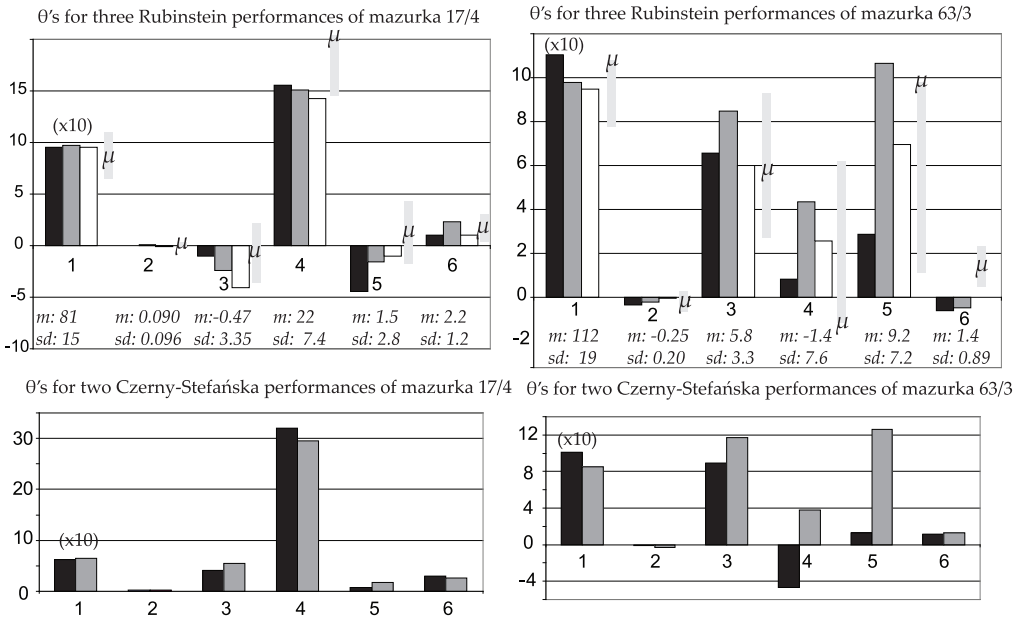

Figure 3: Weights trained for all the performances by Rubinstein and Czerny-Stefańska on Mazurkas 17/4 and 63/3. Each plot shows the six compoents of θ with different colored bars indicating different performances. (The values for $theta_1$ have been scaled by a factor of .1 to fit in the chart.) For brevity we did not include the squared features.

While the reconstriction is by no means a good fit, it does capture an interesting fact about Rubinstein's performance. The second and third reconstructions feature sharp downward spikes, which do seem to align well with downward spikes the target. Inspecting the score of the piece, we found that these downward spikes occur whenever there was a half-note in the left hand (which would cause the left-hand event count to drop.) These spikes all align well with spikes in the target recording, so Rubinstein really does slow down when the left hand plays a half-note.

As can be seen in Figure 3, the weights assigned to the features vary from Mazurka to Mazurka even when the performer is held constant. However, while the features have not characterized the style of a performer sufficiently to identify the perfomer of an arbitrary piece, the values of θ detected by the logistic regression usually seem relatively stable within performances by the same performer of the same piece. (The two recordings of Czerny's playing Mazurka 63/3 were fourty years apart.) An improved version of this technique could be useful for identifying the performer of a disputed recording.

[9]The full data-set and code is available here `http://www.stanford.edu/~marce110/cs229/linear-regression.tgz`

4 Future Directions

4.1 PCA Filtering

Another experiment we performed on the data was to do PCA filtering on the several of the recordings, to test the robustness of Craig's similarity algorithm to degradation of its input.[10]

By PCA Filtering, we mean that we did principle component analysis on the data, calculated the PC loadings for each recording and each Principle Component and then reconstructed each recording using only the first n principle components. We did this for $n = 1, 2, 3, ..., 8, 9, 10, 20, 40, 80$.

We found that by retaining the ten largest principle components of the data, the similarity algorithm was able to detect the true performer of the original recording with high accuracy. [11]

The similarity algorithm may not be taking advantage of all the information that may be available in the filtered recordings, however, it is able to correctly identify the performers of recordings even when the recording has had all but its first ten principle components filtered out. The fact that it is capable of correctly identifying the performer of the recording indicates that, in some sense, the performer's style can be distinguished from the styles of the other performers using ten numbers.

4.2 Linear Regression with More Features

It is clear that we need to extract more features from the scores. Possible features that might be helpful include

1. An average of many different performances of the piece (this would allow the linear regression algorithm to look for patterns in the way that the current preformer differs from the average.)

2. Phrasing information (the position of the beat in a musical phrase). These feature(s) would either be the locations of the measures in the phrase based on hand-generated phrase-boundary, or a collection of sine and cosine waves (i.e. $\sin\left(\frac{2\pi}{k}n\right)$ and $\cos\left(\frac{2\pi}{k}n\right)$ for various values of k (here n is the index of the measure in the piece.) For example, in Figure 1 there are eight regularly spaced phrases that are each twenty-four beats long.

3. Adding more detailed rhythmic information. For example, the number of half-notes, quarter-notes, eighth-notes, etc. in the left and right hands in the current measure as well as the current beat.

4. Music is an ordered squence of events which give rise to the interpretation. Our current models don't take this into account. We can include the features of neighboring beats in the feaature set of the current beat to gain musical context.

5. Nearby dynamics markers in the music (*e.g.* piano and forte markers, crescendos, *etc.*)

4.3 Kernelized Linear Regression

Kernelized linear regression could allow us to detect more complex relationships between the features for example a dissonant note at the end of a measure might warrant a different interpretation from a dissonant note at the start of a measure. Another way to use kernelized linear regression would be to construct a mapping from the musical score of a measure to a tree structure and then use a tree-similarity measure as the kernel.[12] This could have the advantage of automatically detecting which features of the score are relevant but the disadvantage that it could be difficult to tell which features those are.

4.4 Hidden Markov Models

Another possibility is to model the pianists as entities with hidden state using a hidden Markov model. Hidden Markov models have enjoyed success in areas such as speech recognition. Because musical interpretations may have similar structures to vocalized speech (both are acoustic processes designed to be processed by the human brain), this may be grounds for optimism that a hidden Markov model could characterize the playing style of a performer. Another reason that hidden Markov models might be useful models of performers is that pieces of music may contain distinct sections meant to be interpreted with different moods.

[10] *Hybrid Numeric/Rank Similarity Metrics for Musical Performance Analysis*, Craig Sapp, ISMIR 2008. `http://ismir2008.ismir.net/papers/ISMIR2008\_240.pdf`

[11] A graph can be found here `http://www.stanford.edu/~marce110/cs229/PCAandResidue-20081009.pdf`

[12] "A survey of kernels for structured data", Thomas Gärtner, ACM SIGKDD Explorations Newsletter `http://portal.acm.org/citation.cfm?doid=959242.959248`