Intelligent Audio Systems:
A review of the foundations and applications of semantic audio analysis and music information retrieval
These lecture notes contain hyperlinks to the CCRMA Wiki.

On these pages, you can find supplemental material for lectures - providing extra tutorials, support, references for further reading, or demonstration code snippets for those interested in a given topic.

Click on the symbol on the lower-left corner of a slide to access additional resources.

WIKI REFERENCES...
Review from Day 2

- BBQ Today
- Correction on knn formatting
- Name some spectral features
- What are the 3 major components of a MIR system?
- Why do we have to scale our extracted features?
- Which of these did we really not do at all in Lab 2? And, do you think this was a problem?

- How did the lab go?
- Let’s dig into some interesting observations from the lab
- Did you try other audio files – other instrument recognizers?
Basic system overview

Segmentation
(Frames, Onsets, Beats, Bars, Chord Changes, etc)

Feature Extraction
(Time-based, spectral energy, MFCC, etc)

Analysis / Decision Making
(Classification, Clustering, etc)
FEATURE EXTRACTION
Temporal Information

- Rise time or Attack time - time interval between the onset and instant of maximal amplitude
- Attack slope

Picture courtesy: Olivier Lartillot
Temporal Information

- Temporal Centroid
Frame 1

Energy

Octave

1  2  3  4  5  6  7  8  9

1  10 100 1000 10000

4,061.3  10,154.2  545.1  186.7  76.5  34.5  18.2  8.7  5.9
## Features – Frame 1

<table>
<thead>
<tr>
<th>Frame</th>
<th>ZC R</th>
<th>Centroid</th>
<th>BW</th>
<th>Skew</th>
<th>Kurtosis</th>
<th>E1</th>
<th>E2</th>
<th>E3</th>
<th>E4</th>
<th>E5</th>
<th>E6</th>
<th>E7</th>
<th>E8</th>
<th>E9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9</td>
<td>2.8kHz</td>
<td>5kHz</td>
<td>2.2</td>
<td>6.7</td>
<td>4000</td>
<td>10100</td>
<td>545</td>
<td>187</td>
<td>77</td>
<td>35</td>
<td>18</td>
<td>9</td>
<td>6</td>
</tr>
</tbody>
</table>
Frame 2

Energy

Octave

kick

1 2 3 4 5 6 7 8 9

24.8 32.8 5,308.1 1,366.4 360.4 180.2 194.5 68.6 5.3

10000 10000 1000 100 10 1
<table>
<thead>
<tr>
<th>Frame</th>
<th>ZC R</th>
<th>Centroid</th>
<th>BW</th>
<th>Skew</th>
<th>Kurtosis</th>
<th>E1</th>
<th>E2</th>
<th>E3</th>
<th>E4</th>
<th>E5</th>
<th>E6</th>
<th>E7</th>
<th>E8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9</td>
<td>2.8kHz</td>
<td>5kHz</td>
<td>2.2</td>
<td>6.7</td>
<td>4000</td>
<td>10100</td>
<td>545</td>
<td>187</td>
<td>77</td>
<td>35</td>
<td>18</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>423</td>
<td>3.1kHz</td>
<td>4kHz</td>
<td>2</td>
<td>7.2</td>
<td>24</td>
<td>33</td>
<td>5300</td>
<td>1366</td>
<td>360</td>
<td>180</td>
<td>194</td>
<td>68</td>
</tr>
</tbody>
</table>
MFCCs

The idea of MFCCs is to capture spectrum in accordance with human perception.

1. STFT
2. log(STFT)
3. Perform mel-scaling to group and smooth coefficients. (perceptual weighting)
4. Decorrelate with DCT

[...continued...]
MFCC

0 2 4 6 8 10 12 14

FFT magnitudes

0 50 100 150 200 250 300 350

0 1 2

mel filterbank output

-30 -20 -10 0 10

ceps = DCT of mel filter output

frame

0 50 100 150 200 250 300 350

-6 -4 -2 0 1

FFT magnitudes

x 10^4

guitar.wav
MFCC of Music
(Petruncio, 2003)

Piano

Saxophone

Tenor Opera Singer

Drums
Spectral Energy vs. MFCC
Features: Measuring changes

• $\Delta$ and $\Delta \Delta$
  – Change between frames
  – How quickly the change is occurring

• Spectral flux is the distance between the spectrum of successive frames
Feature extraction

- Feature design and creation uses one’s domain knowledge.
- Choosing discriminating features is critical.
- Smaller feature space yields smaller, simpler models, faster training, often less training data needed.
ANALYSIS AND DECISION MAKING
Supervised vs. Unsupervised

• Unsupervised - “clustering”
• Supervised – binary classifiers (2 classes)
• Multiclass is derived from binary
Clustering

- Unsupervised learning – find pockets of data to group together
- Statistical analysis techniques
Clustering

- $K = \# \text{ of clusters}$

- Choosing the number of clusters – note that choosing the “best” number of clusters according to minimizing total squared distance will always result in same $\# \text{ of clusters}$ as data points.
Clustering

The basic goal of clustering is to divide the data into groups such that the points within a group are close to each other, but far from items in other groups.

Hard clustering – each point is assigned to one and only one cluster.
Demo

- http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html
The key points relating to *k-means clustering* are:

- *k*-means is an automatic procedure for clustering unlabelled data;
- it requires a pre-specified number of clusters;
- Clustering algorithm chooses a set of clusters with the minimum within-cluster variance;
- Guaranteed to converge (eventually);
- Clustering solution is dependent on the initialization (You get different results with each running).
K-Means

The initialization method needs to be further specified. There are several possible ways to initialize the cluster centers:

- Choose random data points as cluster centers
- Randomly assign data points to K clusters and compute means as initial centers
- Choose data points with extreme values
- Find the mean for the whole data set then perturb into k means
- Find ground-truth for data
EVALUATION
Our classifier accuracy is 83.4%
Cross-validation

• Say, 10-fold cross validation
• Divide test set into 10 random subsets.
• 1 test set is tested using the classifier trained on the remaining 9.
• We then do test/train on all of the other sets and average the percentages. Helps prevent over fitting.
• Do not optimize too much on cross validation – you can severely overfit. Sanity check with a test set.
Cross-validation
Cross-validation

Fold 1: 70%

TRAINING SET

TEST
Cross-validation

Fold 1: 70%
Fold 2: 80%
Cross-validation

Fold 1: 76%
Fold 2: 80%
Fold 3: 77%
Fold 4: 83%
Fold 5: 72%
Fold 6: 82%
Fold 7: 81%
Fold 8: 71%
Fold 9: 90%
Fold 10: 82%
Mean = 79.4%
Stratified Cross-Validation

• Same as cross-validation, except that the folds are chosen so that they contain equal proportions of labels.
Spectral Bands
Frame 1

Energy

Octave

1  2  3  4  5  6  7  8  9

1  2  3  4  5  6  7  8  9

1  1  10  100  1000  10000

10,154.2 545.1 186.7 76.5 34.5 18.2 8.7 5.9
## Features – Frame 1

<table>
<thead>
<tr>
<th>Frame</th>
<th>ZC R</th>
<th>Centroid</th>
<th>BW</th>
<th>Skew</th>
<th>Kurtosis</th>
<th>E1</th>
<th>E2</th>
<th>E3</th>
<th>E4</th>
<th>E5</th>
<th>E6</th>
<th>E7</th>
<th>E8</th>
<th>E9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9</td>
<td>2.8kHz</td>
<td>5kHz</td>
<td>2.2</td>
<td>6.7</td>
<td>4000</td>
<td>10100</td>
<td>545</td>
<td>187</td>
<td>77</td>
<td>35</td>
<td>18</td>
<td>9</td>
<td>6</td>
</tr>
</tbody>
</table>
Energy

Octave

Frame 2

kick

Energy

Octave

24.8
32.8
5,308.1
1,366.4
360.4
180.2
194.5
68.6
5.3

Frame 2
# Features: SimpleLoop.wav

<table>
<thead>
<tr>
<th>Frame</th>
<th>ZCR</th>
<th>Centroid</th>
<th>BW</th>
<th>Skew</th>
<th>Kurtosis</th>
<th>E1</th>
<th>E2</th>
<th>E3</th>
<th>E4</th>
<th>E5</th>
<th>E6</th>
<th>E7</th>
<th>E8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9</td>
<td>2.8kHz</td>
<td>5kHz</td>
<td>2.2</td>
<td>6.7</td>
<td>4000</td>
<td>10100</td>
<td>545</td>
<td>187</td>
<td>77</td>
<td>35</td>
<td>18</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>423</td>
<td>3.1kHz</td>
<td>4kHz</td>
<td>2</td>
<td>7.2</td>
<td>24</td>
<td>33</td>
<td>5300</td>
<td>1366</td>
<td>360</td>
<td>180</td>
<td>194</td>
<td>68</td>
</tr>
</tbody>
</table>
Log Spectrogram

![Log Spectrogram Image]
The resulting graph indicate the cross-correlation score for each different tonality candidate.
http://www.chordpickout.com/index.html
Unsupervised learning – find pockets of data to group together

Statistical analysis techniques

Clustering
Day 3 Lab

• Get your Lab 2 working
  – Make sure that training data = 100% accurate
  – Try the test snares and test kicks
    • Write down your accuracy and parameters
    • Change the number of features
    • Add or replace current features with different values
      – (e.g., mirbrightness, mirrolloff)

• Demo - tonality
• Demo – tempo