Music Analysis

Josiah Boning

TJHSST Senior Research Project
Computer Systems Lab, 2007-2008

Purpose

- Apply machine learning algorithms to audio data
  - Neural Networks
- Autonomously identify what is music

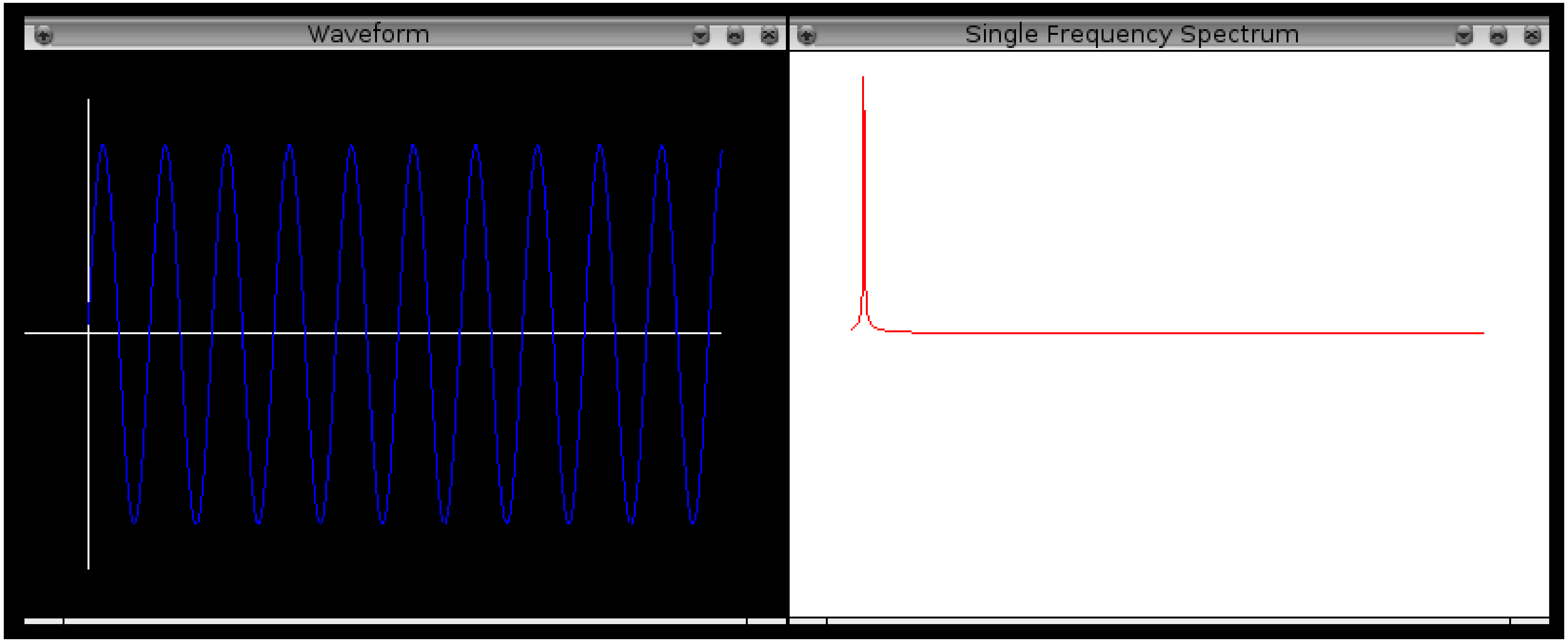
Background

- Bigarelle and Iost (1999)
  - Music genre can be identified by fractal dimension
- Basilie et al. (2004)
  - Music genre can be identified by machine learning algorithms
  - Used discrete MIDI data

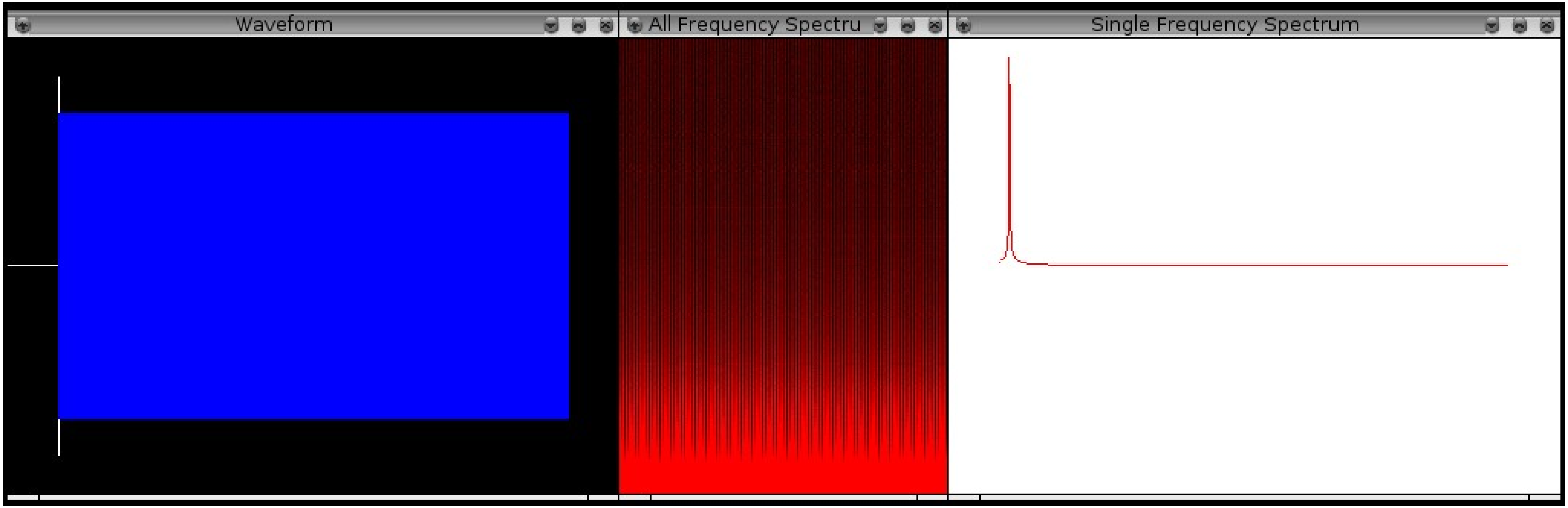
Methods

- Data Processing
  - Spectral Analysis: Fourier Transform
  - Fractal Dimension: Variation and ANAM Methods
- Machine Learning
  - Feed-Forward Neural Network

Fourier Transform



Fourier Transform



Fractal Dimension

$$D = \lim_{\epsilon \to 0} \frac{\log N(\epsilon)}{\log \frac{1}{\epsilon}}$$

- “a statistical quantity that gives an indication of how completely a fractal appears to fill space” -- Wikipedia
- Audio data is set of discrete sample points, not a function
- Therefore, fractal dimension can only be estimated

Fractal Dimension

- Variation Method:

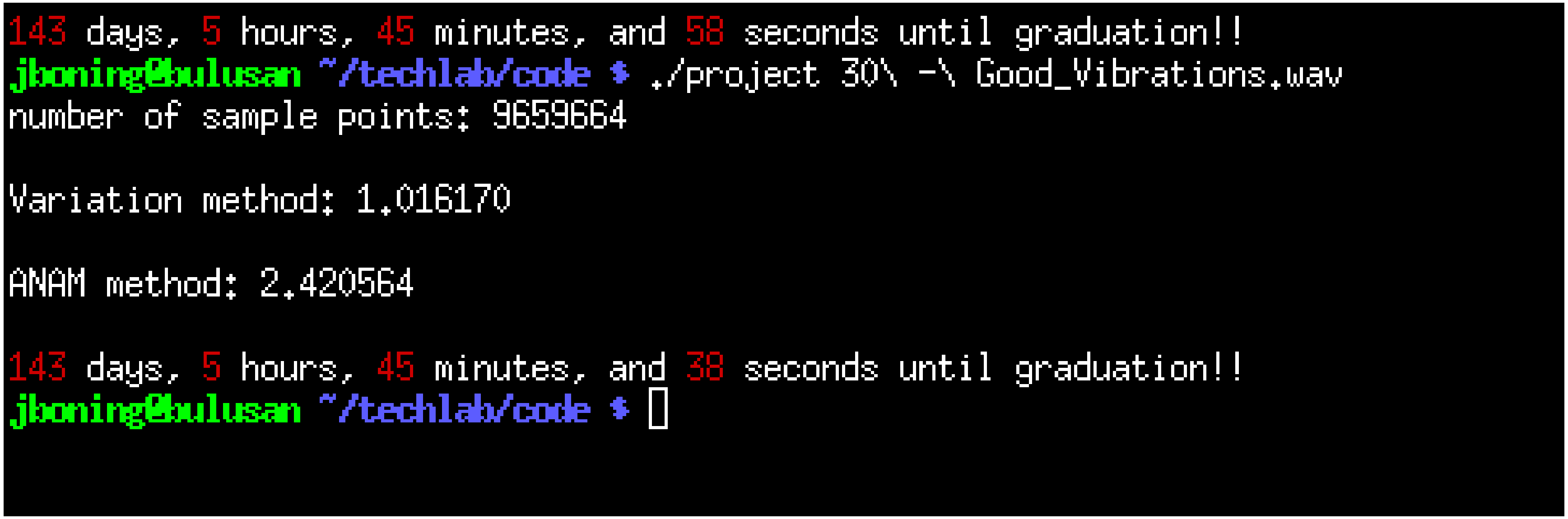
$$\lim_{\tau\to 0}\left(2-\frac{\log\left(\frac{1}{b-a}\int_{a}^{b}\left|max\left(\underset{|x-t|<\tau}{f}(t)\right)-min\left(\underset{|x-t|<\tau}{f}(t)\right)\right|dx\right)}{\log\tau}\right)$$

- ANAM Method:

$$\lim_{\tau\to 0}\left(2-\frac{\log\left(\frac{1}{b-a}\int_{x=a}^{x=b}\left[\frac{1}{\tau^{2}}\int_{t_1=0}^{\tau}\int_{t_2=0}^{\tau}\left|f(x+t_1)-f(x-t_2)^{\alpha}\right|dt_1\,dt_2\right]^{1/\alpha}dx\right)}{\log\tau}\right)$$

Fractal Dimension

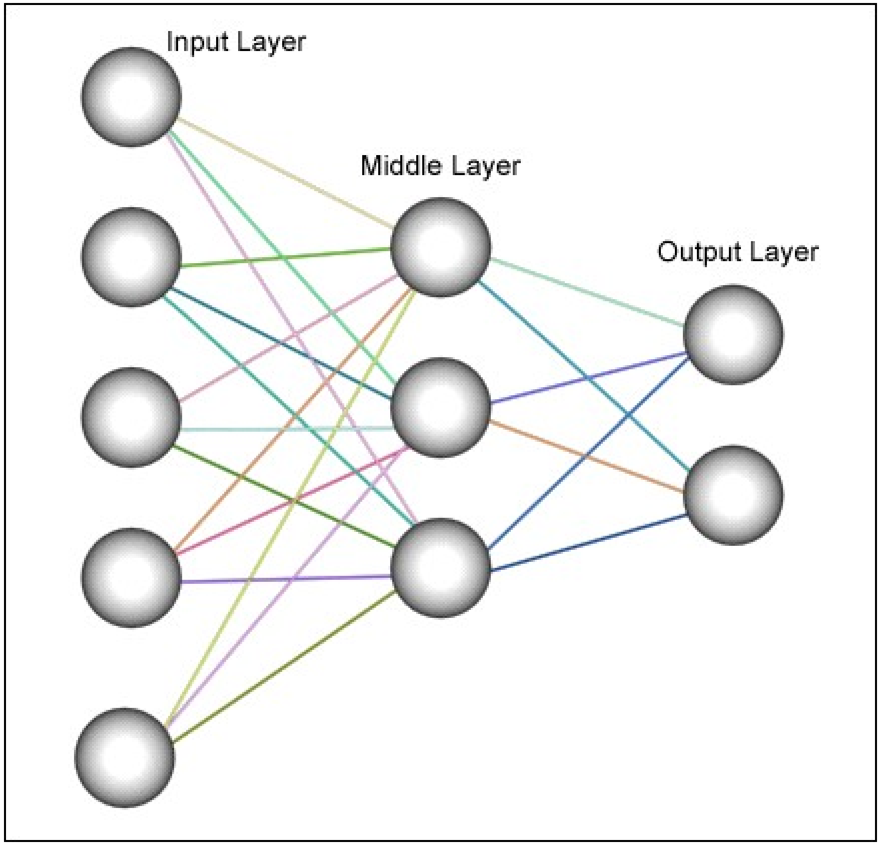
- Variation and ANAM methods are two methods of calculating/estimating the same value
- Should yield similar results
- They don't...



```
143 days, 5 hours, 45 minutes, and 58 seconds until graduation!!
jboning@bulusan ~/techlab/code $ ./project 30\ -\ Good_Vibrations.wav
number of sample points: 9659664

Variation method: 1.016170

ANAM method: 2.420564

143 days, 5 hours, 45 minutes, and 38 seconds until graduation!!
jboning@bulusan ~/techlab/code $ 
```

Machine Learning

- Neural networks
- Feed-Forward



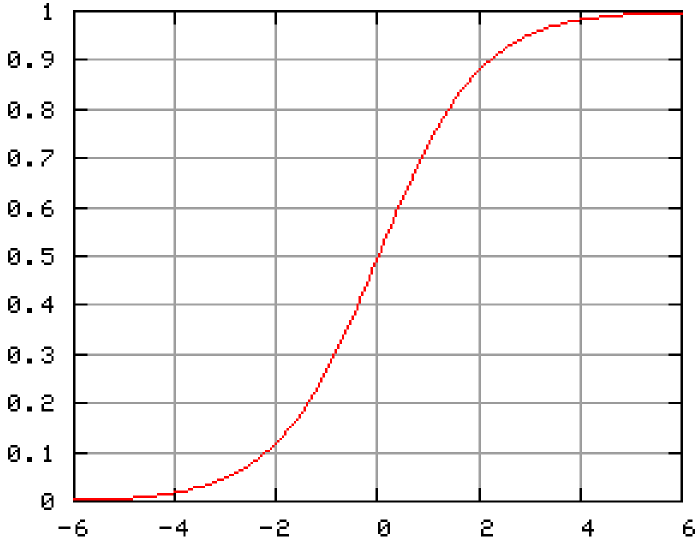
Neural Network Data Structures

```
typedef struct _neuron {
    double value;
    struct _edge* weights;
    double num_weights;
} neuron;

typedef struct _edge {
    struct _neuron* source;
    double weight;
} edge;

// sizeof(neuron) == 20
// sizeof(edge) == 16
```

Neural Network Pseudo-Code

```
For each layer:
   For each node:
      value = 0
      For each node in the previous layer:
         value += weight * value of other node
      value = sigmoid(value)
```

$$P(t) = \frac{1}{1 + e^{-t}}.$$



Neural Network Challenges: Memory

- Audio data: 44100 samples/sec
- Processing 1 second of data
- 44100 input, 44100 hidden nodes, 1 output node
  - Memory: (44100 * 2 + 1) * 20 bytes = 1.7 MB
- 44100 ^ 2 + 44100 edges
  - Memory: (44100 ^ 2 + 44100) * 16 bytes = **31 GB**

Neural Network Challenges: Training

- Training Algorithms
- Training Data
- Backwards Propagation