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

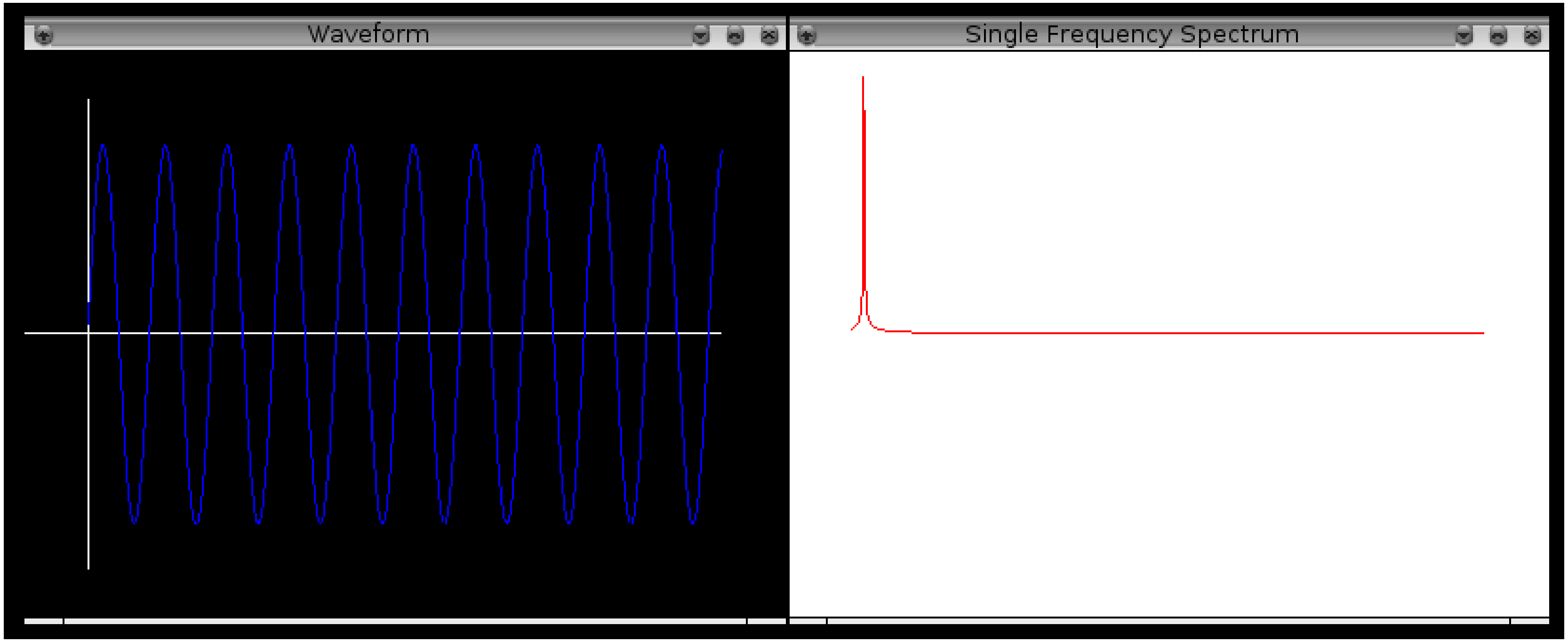
Background

- 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

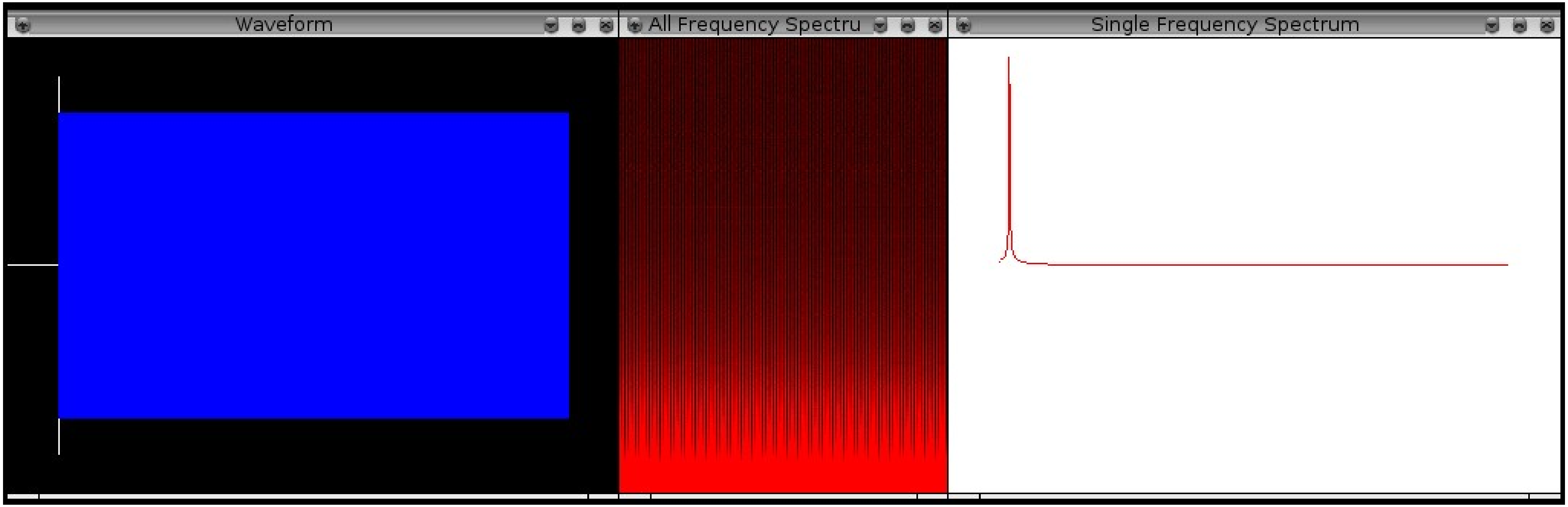
Methods

- 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:

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

- ANAM Method:

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

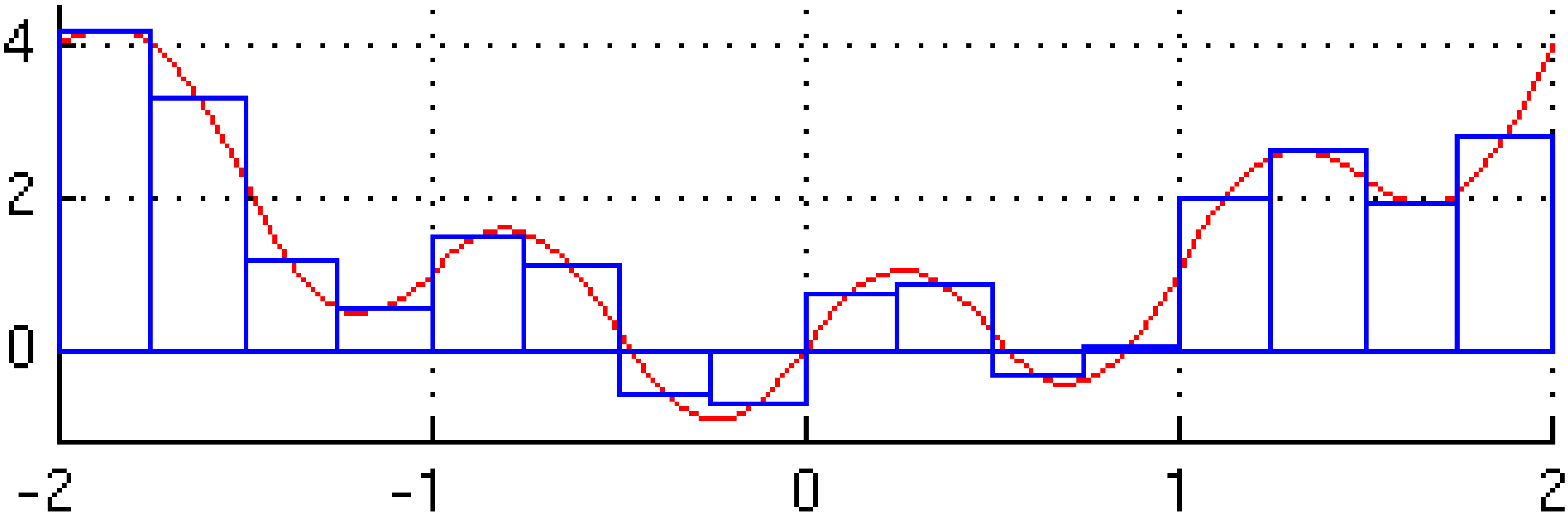
Fractal Dimension

- 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 $ 
```

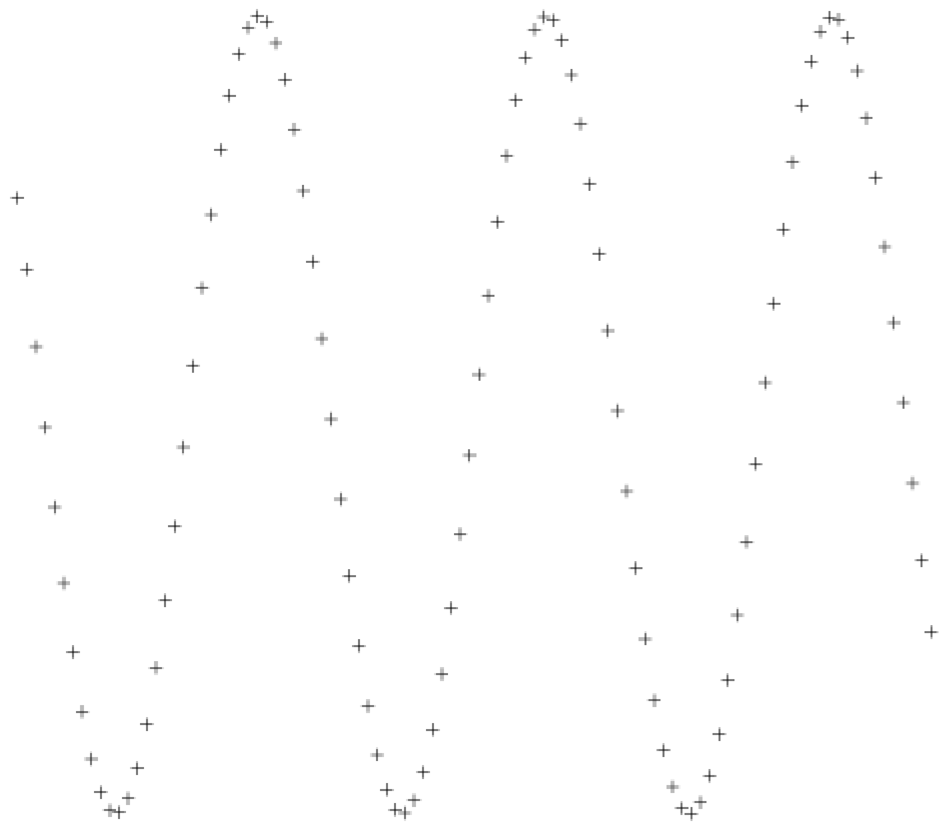
Integration Error

- Euler’s Method -- Rectangles



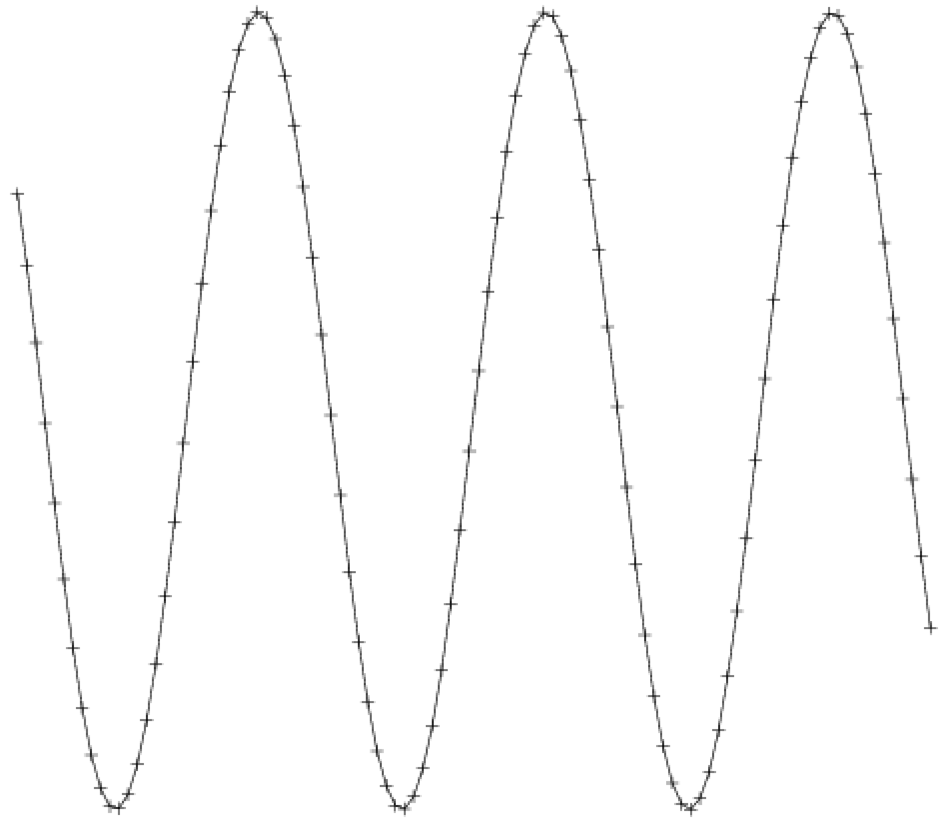
Fractal Dimension

- Discrete data – limited by sampling frequency



Integration Error

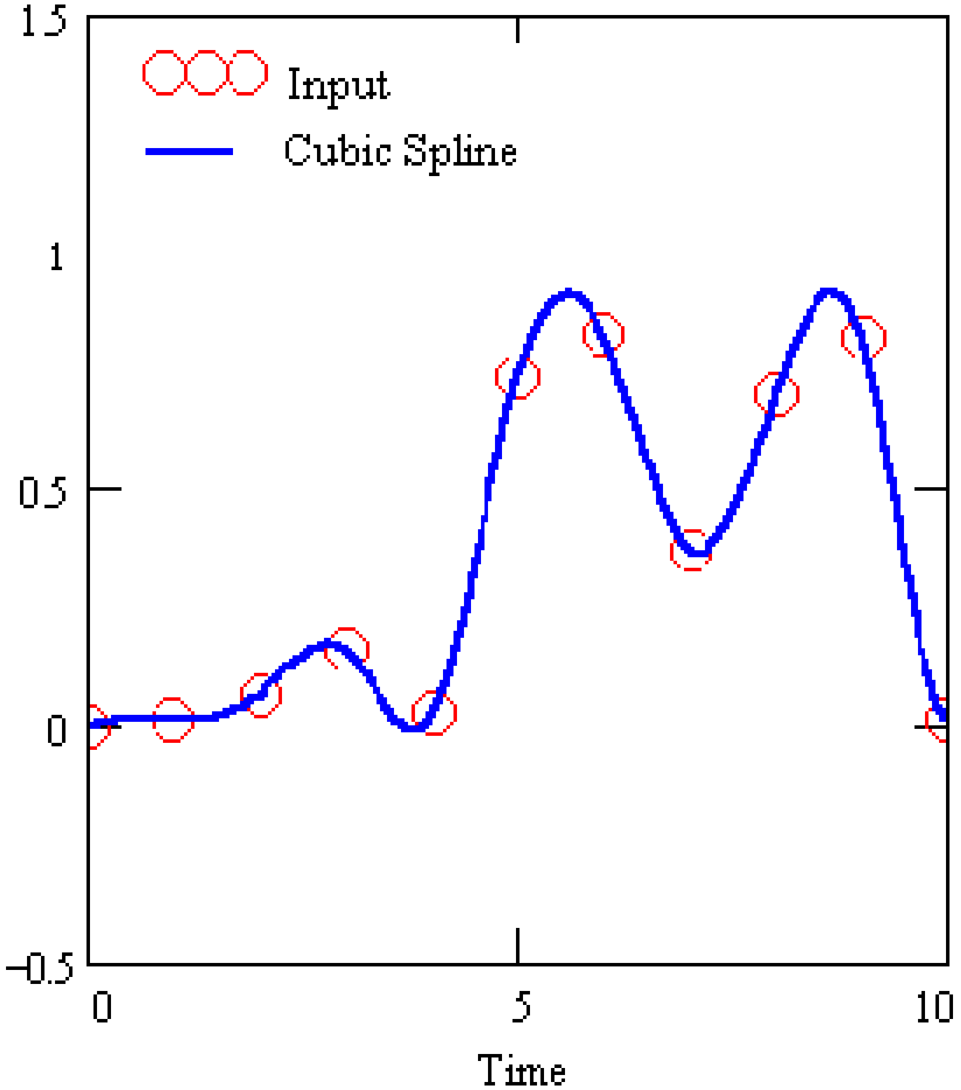
- Solution: Interpolation



Interpolation Methods

- Polynomial Interpolation – Single formula used to meet all data points
  - Lagrange Interpolation
  - Newton's Divided Differences Interpolation
  - Chebyshev Interpolation
- Splines – several formulas in segments
  - Linear Spline – data points connected by lines
  - Cubic Spline – data points connected by cubic polylnomials

Cubic Splines



Cubic Splines

A cubic spline $S(x)$ through the data points $(x_1, y_1),\ldots,(x_n, y_n)$ is a set of cubic polynomials

$$S_1(x) = y_1 + b_1(x - x_1) + c_1(x - x_1)^2 + d_1(x-x_1)^3 \text{ on } [x_1,x_2]$$
$$S_2(x) = y_2 + b_2(x - x_2) + c_2(x - x_2)^2 + d_2(x-x_2)^3 \text{ on } [x_2,x_3]$$
$$\vdots$$
$$S_{n-1}(x) = y_{n-1} + b_{n-1}(x - x_{n-1}) + c_{n-1}(x - x_{n-1})^2 + d_{n-1}(x-x_{n-1})^3 \text{ on } [x_{n-1},x_n]$$

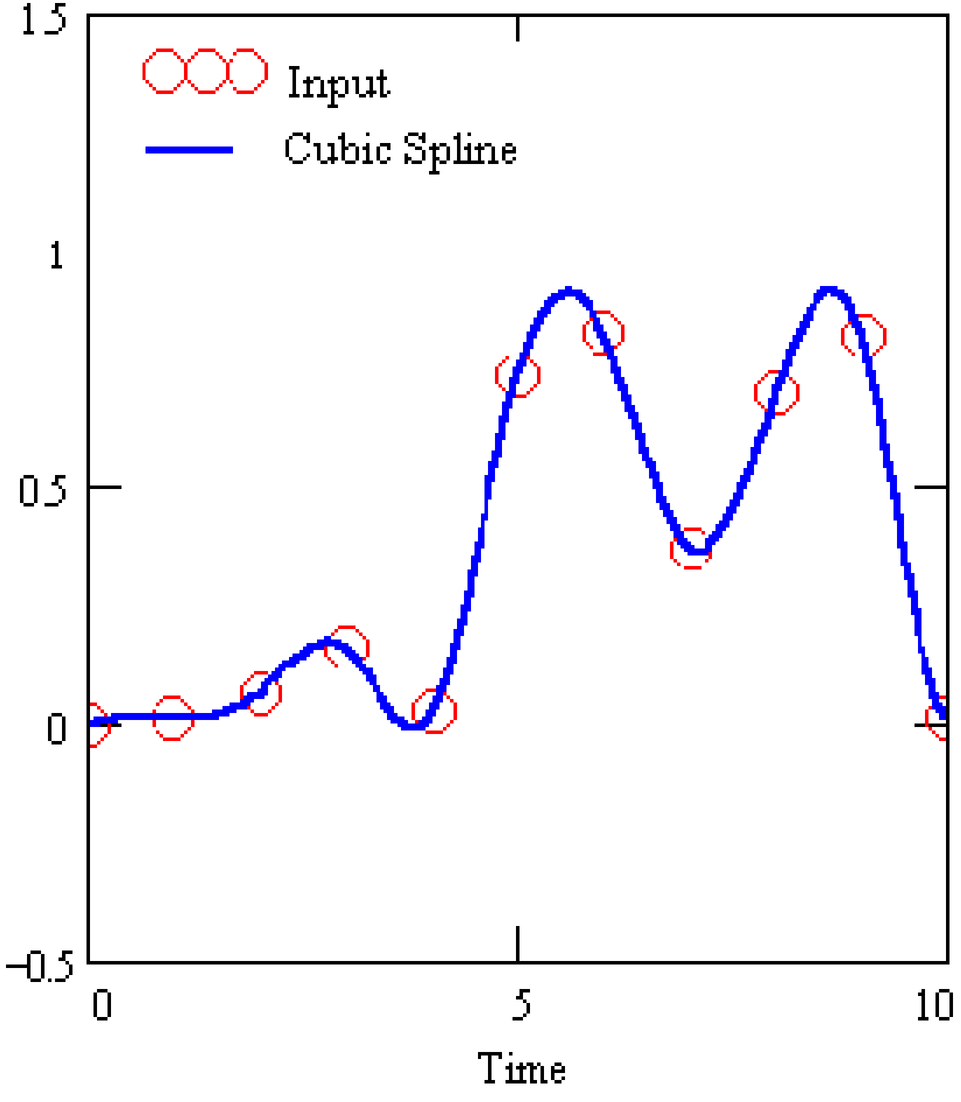
with the following properties:

Property 1: $S_i(x_i) = y_i$ and $S_i(x_{i+1}) = y_{i+1}$ for $i = 1, \ldots, n - 1$

Property 2: $S_{i-1}'(x_i) = S_i'(x_i)$ for $i = 2, \ldots, n - 1$

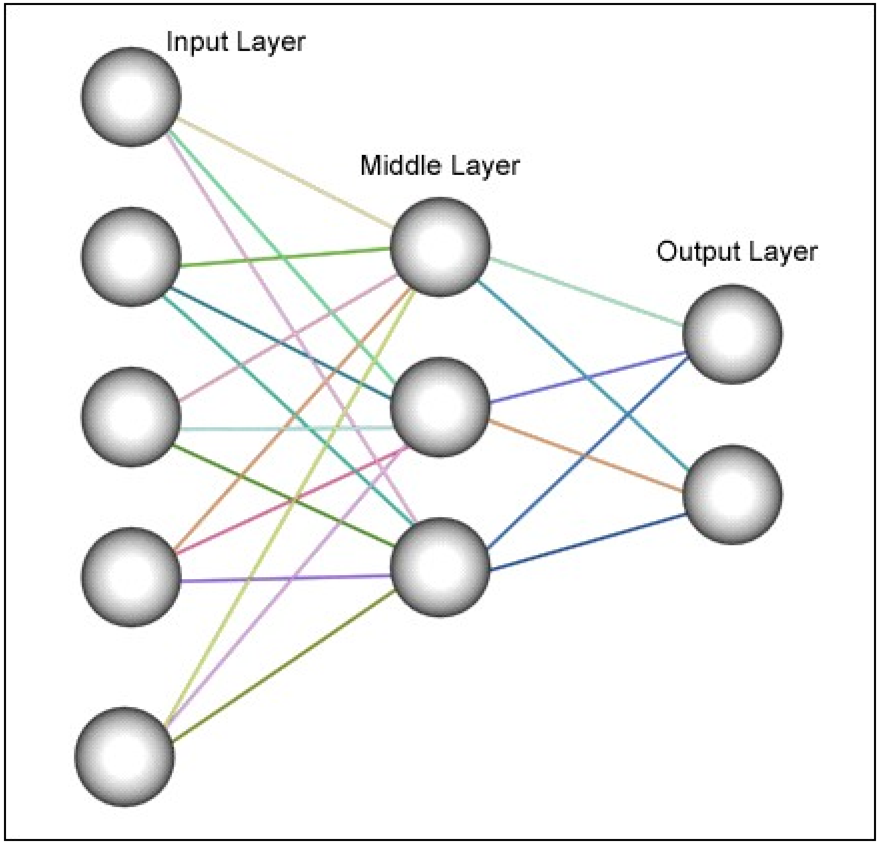
Property 3: $S_{i-1}''(x_i) = S_i''(x_i)$ for $i = 2, \ldots, n - 1$

Cubic Splines



Machine Learning

- Neural networks
- Feed-Forward



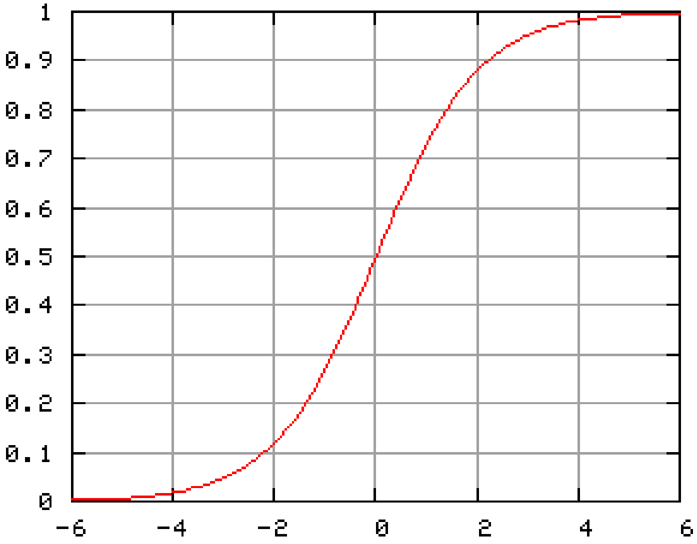
Neural Network Data Structures

```
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**
- **Solution(?): Alternative input (fractal dimension, Fourier transform data) rather than audio data**

Neural Network Challenges: Training

- Training Algorithms
- Training Data
- Backwards Propagation