Musical Instrument Identification Using Wavelets and Neural Nets

Jeff Livingston

and

Nathan Shepard

Overview

- Classification goal
- Preprocessing
  - Classification vector derived from wavelet transform coefficients
- Neural Network Building
  - Network selection & training
  - Performance analysis & optimization
- Q&A

Soloing Instrument Classification

- Goal: to identify which instrument in an ensemble is dominant (soloing) in an musical excerpt
- Practical Uses:
  - Identify logical sections in a song, to allow users to skip to sections of interest
  - Content based online music searches
    - E.g. find recordings by artist X that contain saxophone solos

Data Set

- Small ensemble jazz groups
  - Consistent characteristics: instrumentation, style
  - Alternating individual soloists
- Five instrument classification types
  - Bass, drums, piano, trumpet, tenor sax
- Two second blocks
  - Sized to insure soloing instrument is present

Audio content description for classification

- Most current techniques rely on Fourier Transform techniques (STFT)
  - Uniform temporal resolution for all frequencies, determined by window size and spacing
  - Uniformly spaced analysis frequencies
  - Doesn’t match human auditory perception model
    - Information in STFT is not distributed efficiently w.r.t the way we perceive sound

Wavelet Transform for Audio Analysis

- Variable frequency/temporal resolution across frequencies
  - High freq’s: highly localized in time, coarse frequency resolution
  - Low freq’s: less localized in time, fine frequency resolution.
- Mirrors human auditory perception model
  - Analysis freq bands logarithmically spaced
  - Information concentrated more ideally than STFT

Preprocessing: Wavelet Rank Dispersion Measure

- Obtain WT coef's
  - E.g. 3 scales 5 samples

$$\mathbf{C} = \begin{array}{ccccc|c} 1 & 2 & 3 & 4 & 5 & \\ \hline 0.43 & 0.22 & 0.14 & 0.76 & 0.33 & 1 \\ 0.10 & 0.32 & 0.11 & 0.28 & 0.90 & 2 \\ 0.54 & 0.49 & 0.34 & 0.18 & 0.91 & 3 \end{array}$$

- Assign coefs' (magnitude) rank over wavelet scales for each sample (column)

1	2	3	4	5	
0.43 (2)	0.22 (3)	0.14 (2)	0.76 (1)	0.33 (3)	1
0.10 (3)	0.32 (2)	0.11 (3)	0.28 (2)	0.90 (2)	2
0.54 (1)	0.49 (1)	0.34 (1)	0.18 (3)	0.91 (1)	3

Preprocessing: Wavelet Rank Dispersion Measure

- Retain just the rankings, discard coefs
- Sum the frequency of ranks for each scale, assemble into a histogram matrix
- This is the wavelet dispersion measure, NxN (N = # of scales)
  - Significant data reduction

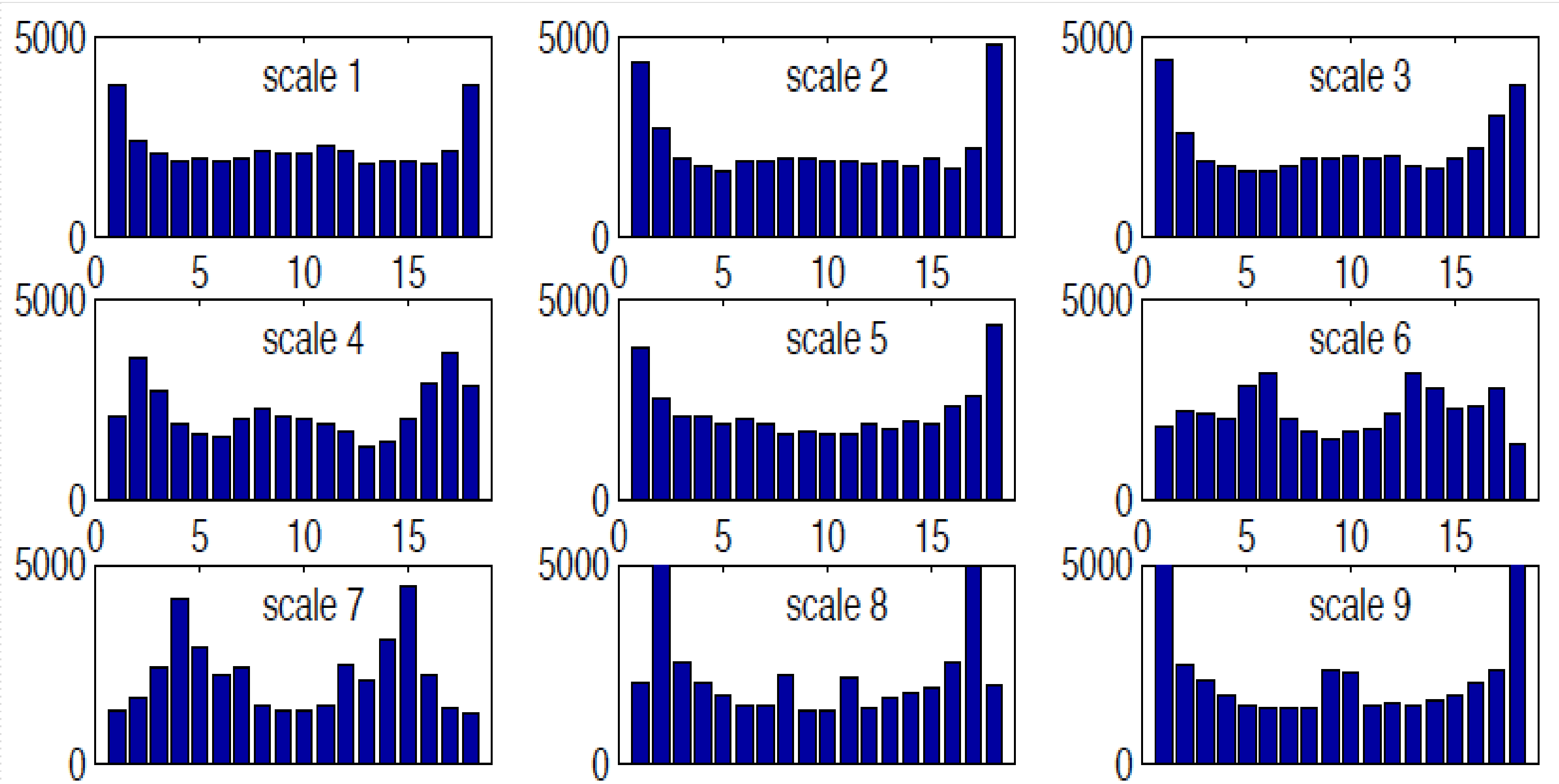
$$\begin{array}{ccccc|c} 1 & 2 & 3 & 4 & 5 & \\ \hline 2 & 3 & 2 & 1 & 3 & 1 \\ 3 & 2 & 3 & 2 & 2 & 2 \\ 1 & 1 & 1 & 3 & 1 & 3 \end{array}$$

$$C_{\mathrm{disp}} = \begin{array}{ccc|c} 1 & 2 & 3 & \\ \hline 1 & 2 & 2 & 1 \\ 0 & 3 & 2 & 2 \\ 4 & 0 & 1 & 3 \end{array}$$

Preprocessing: Wavelet Rank Dispersion Measure

- Further data reduction by discarding small % of highest and lowest scales' histogram columns (low and high magnitude outliers)
- Wavelet rank dispersion vector (WRDV) constructed by concatenating rows of histogram matrix into a single vector
  - Each audio block converted into a WRDV
- WRDV = input to neural net

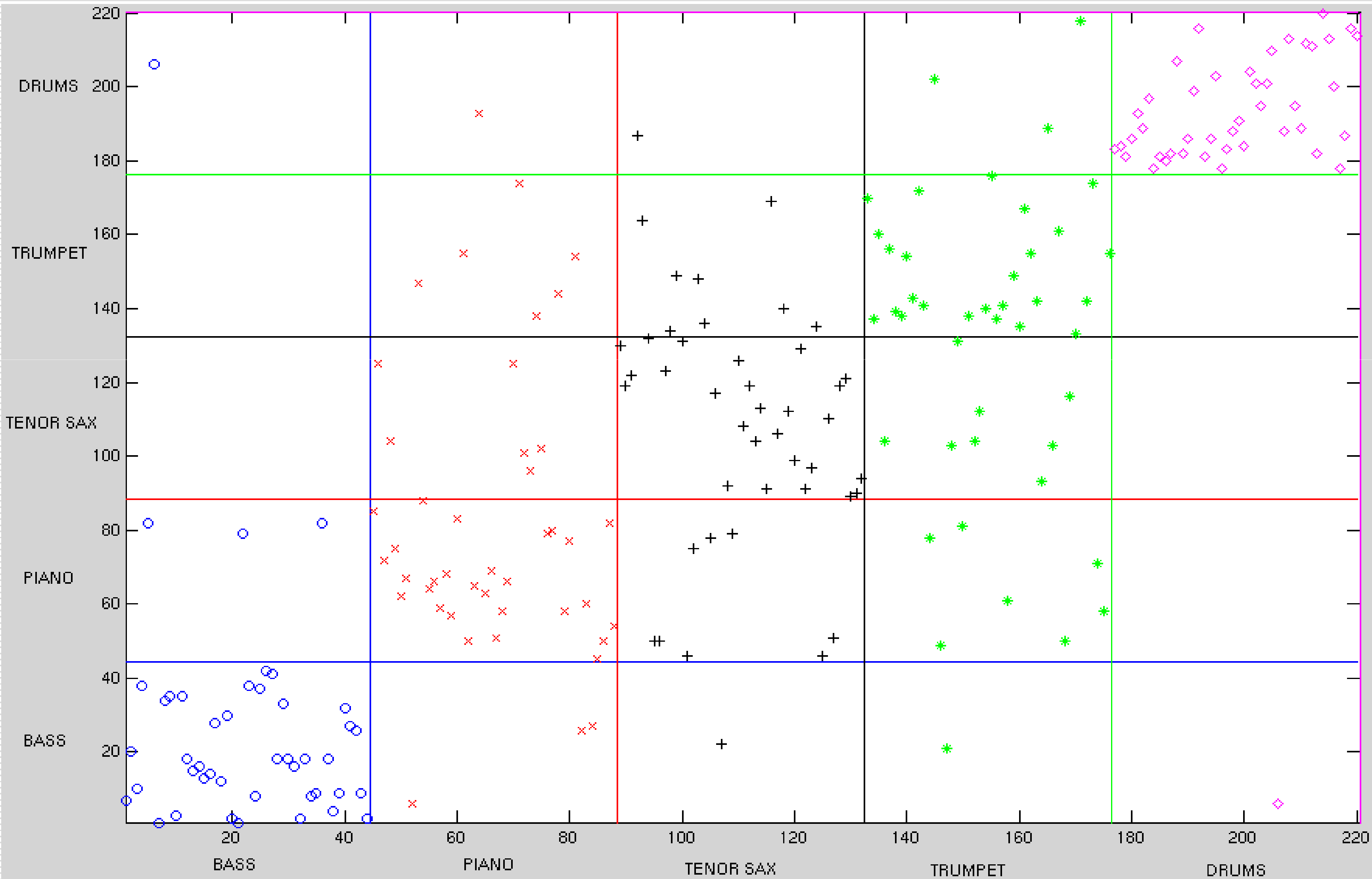
Wavelet Rank Dispersion Histograms



WRDV Classification Performance Analysis

- Use correlation coef's for WRDVs over the data set to indicate potential for good classification performance
- For good classification performance potential, we expect:
  - High correlation between like instruments
  - Low correlation between unlike instruments
- WRDV exhibits these characteristics

WRDV correlation,73%matching



Network Optimization

Nathan Shepard

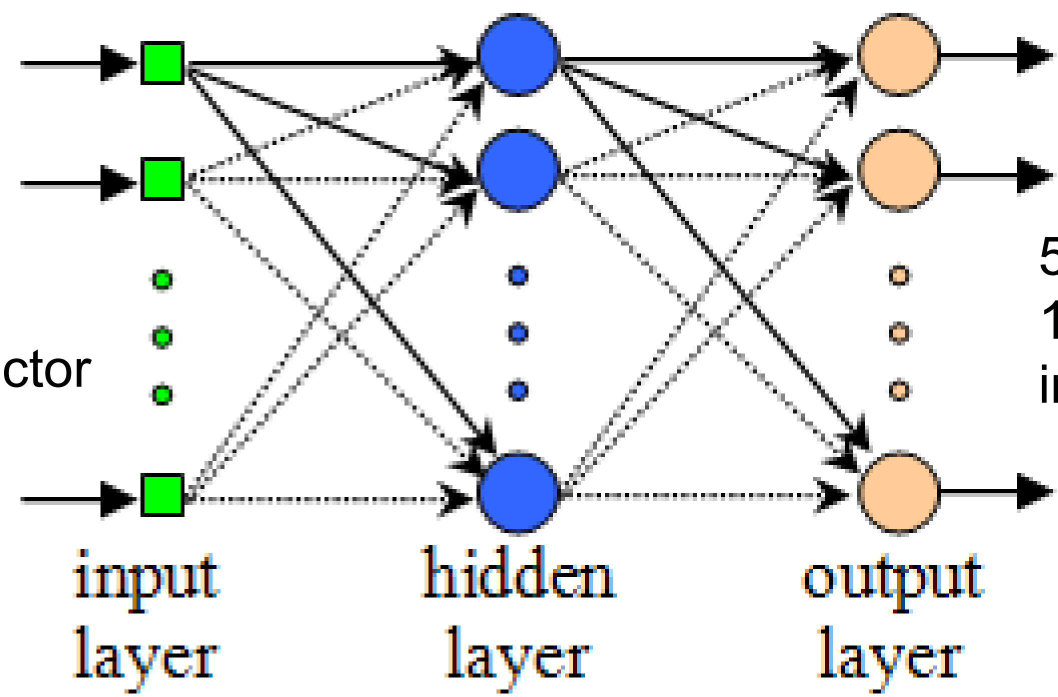
- Network structure
- Network parameters
- Network performance measure
- Empirical improvements
- Final network selection
  - Best performance, how good is it?

Network Structure

- Two appropriate networks:
  - Multi-layer perceptron network (MLP)
    - Allows for high-dimension input space
    - Invariant network structure
  - Radial basis function network (RBF)
    - Good classifier for wavelet transform input

MLP Linear Network

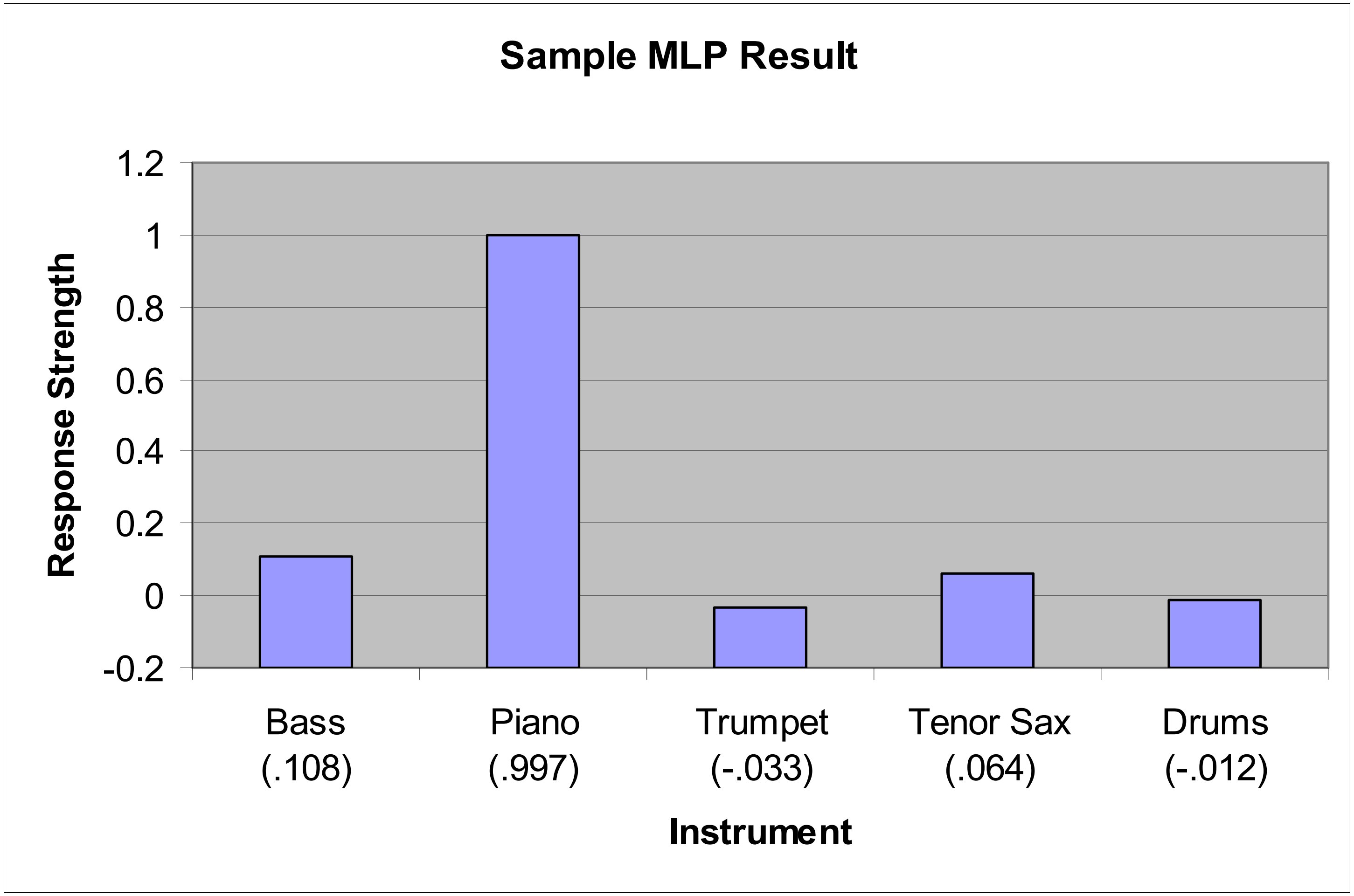
480 inputs
from wavelet
dispersion vector



5 outputs,
1 for each
instrument

- Variables:
  - Number of hidden units
  - Number of epochs for weight updates

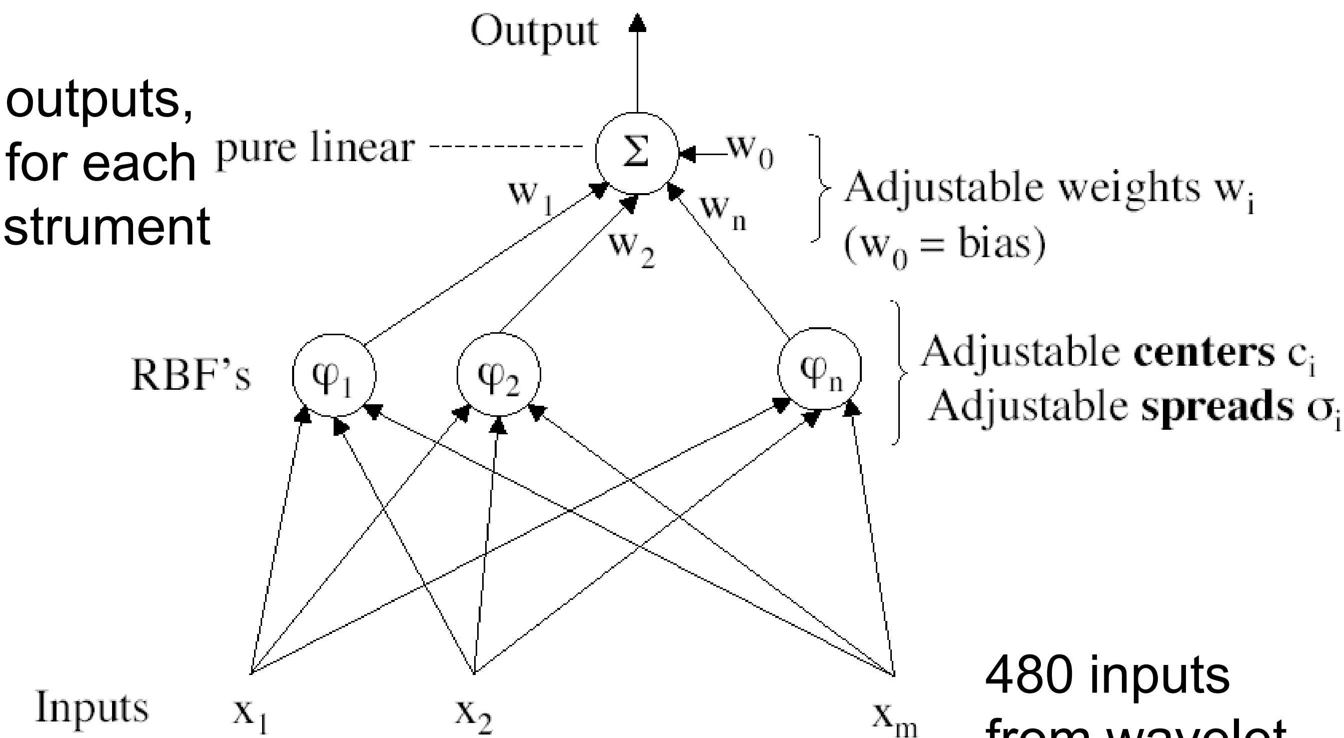
MLP Output



Mean Squared Error = 32.4%

RBF Linear Network

5 outputs,
1 for each
instrument



480 inputs
from wavelet
dispersion vector

- Variables:
  - Number of centers
  - Spread width

Network Performance Measure

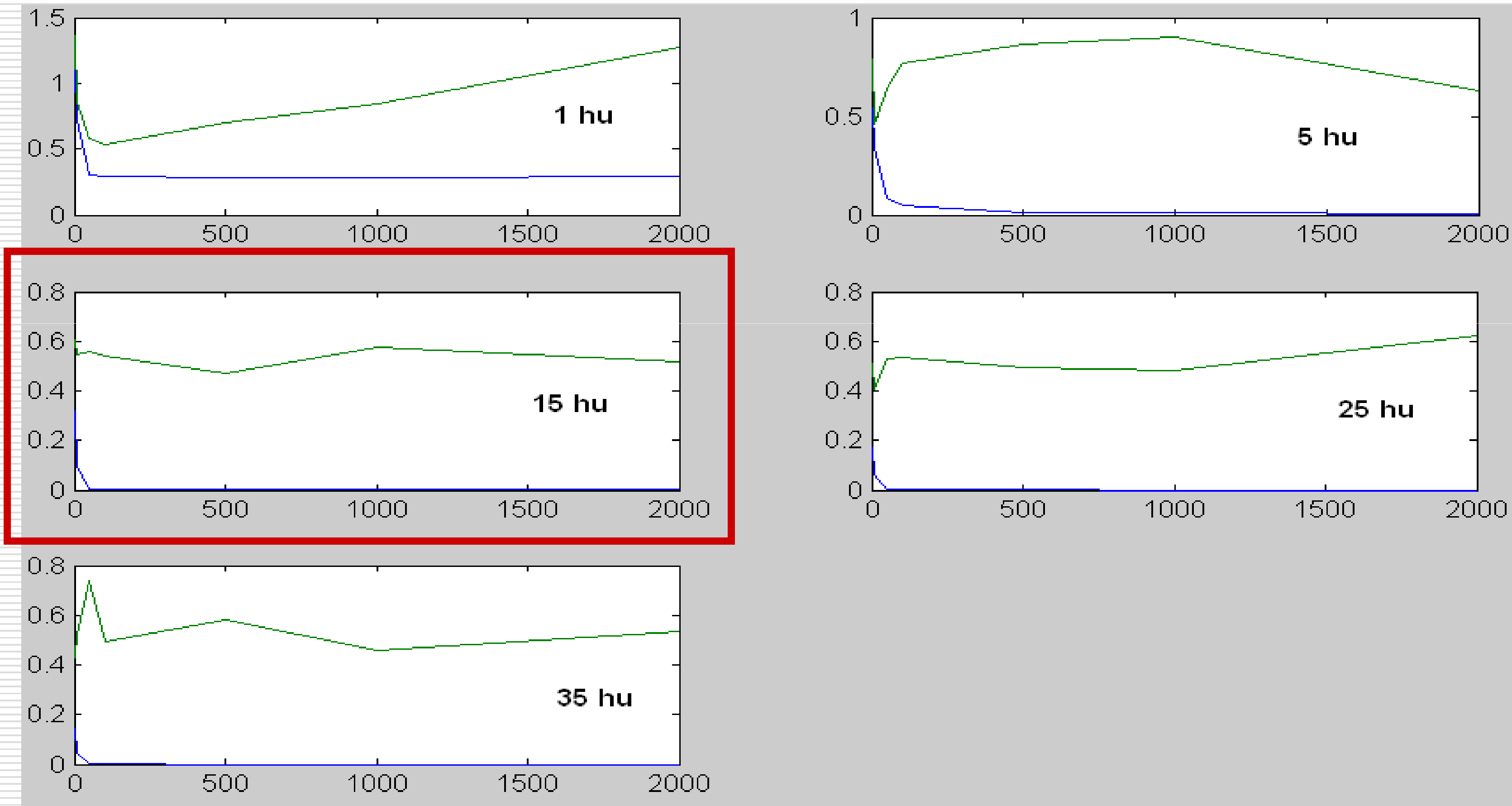
- Input is a wavelet dispersion vector
  - 480 dimensions
  - 220 samples (44 from each of 5 instruments)
- Leave-one-out training/validation method
  - High-dimensional input requires many training samples to cover the input space
  - RBF needs many training samples for accurate K-means clustering.

Simulation Results

- Multi-layer perceptron (MLP)
  - For each number of hidden units tested,
    - Graph training error vs. number of epochs
    - Graph validation error vs. number of epochs

- Radial basis function (RBF)
  - For each value of Gaussian spread,
    - Graph training error vs. number of centers
    - Graph validation error vs. number of centers

MLP Results



RBF Results

MLP vs. RBF

Final Network Performance

- Network parameters:
  - #hus, epochs, center, spread

Questions?