# Automatic Speech Recognition: Classification of Phonemes

Robert Schwartzberg & Eric Mauro

# Background

- Automatic Speech Recognition (ASR) is the process of turning speech into text
- Real world applications:
  - Voice-controlled machinery and electronics
  - Automatic closed-captioning of videos
  - Increased accessibility for people with disabilities
- Phonemes are unique phonetic sounds that make up words
- There are 61 different phonemes in the TIMIT dataset
- Phoneme recognition is the first step in ASR

# Methods

- Phoneme classification using the TIMIT dataset
  - 630 American English speakers of 8 different major dialect regions
  - Training:  177080 phonemes, Test: 64145 phonemes
- Obtain MFCCs from phoneme audio
  - 12 coefficients - MFCCs, ΔMFCCs,  ΔΔMFCCs
  - Mean and standard deviation of coefficients from windowed phoneme samples
  - 72 total input parameters per phoneme sample

# Phoneme Groups

**(Halberstadt, 1998)**

6 groups:

- Vowels / Semivowels
- Weak Fricatives
- Nasals / Flaps
- Stops
- Strong Fricatives
- Closures

3 broader groups:

- Sonorant (Vowels and Nasals)
- Obstruent (Strong Fricatives, Weak Fricatives, and Stops)
- Silence (Closures)

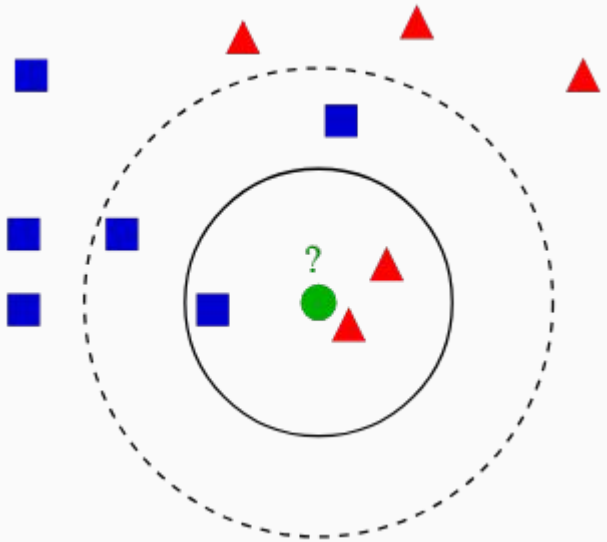
**(Scanlon et al., 2007)**

5 groups:

- Vowels
- Nasals
- Stops
- Silences
- Fricatives

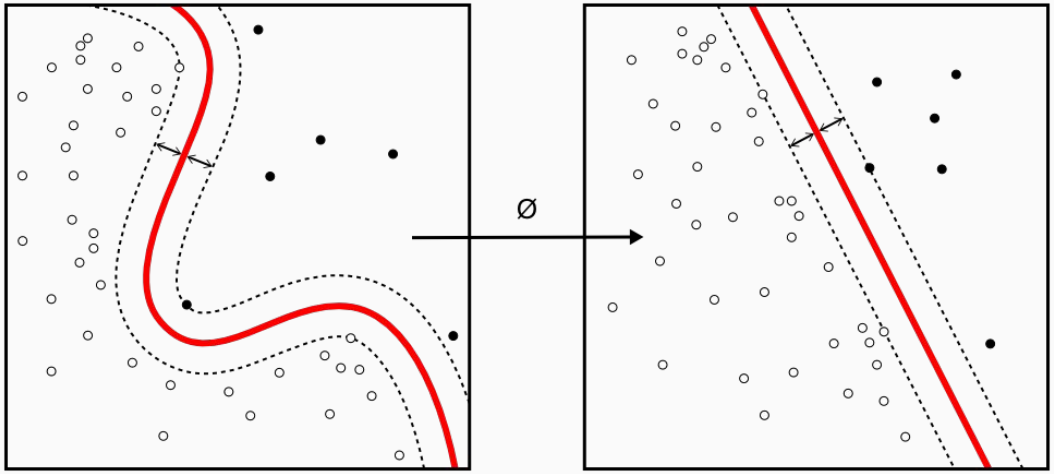
# K-Nearest Neighbor (KNN)

- Class assigned to object based on the most common class of the K neighbors closest to it
- “Lazy learning”



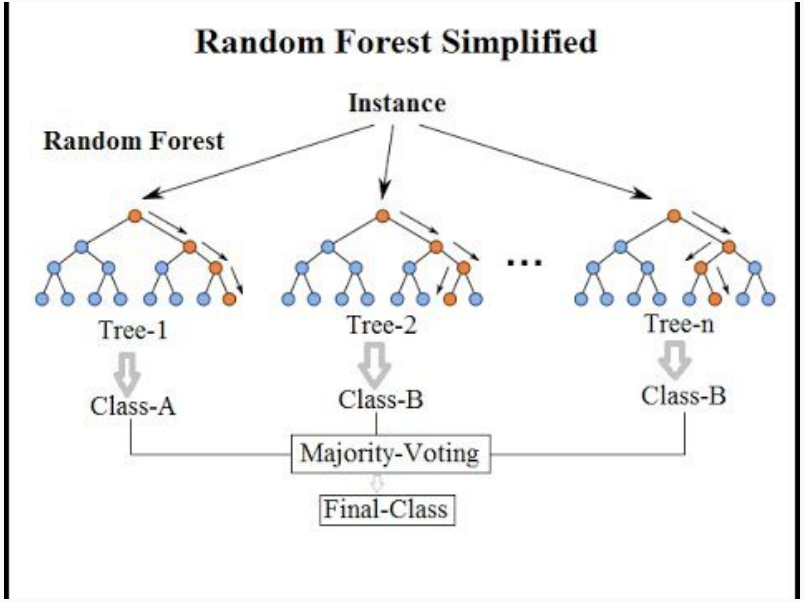
# Support Vector Machine (SVM)

- Supervised Learning
- Linear binary classifier
- Hyperplane separates the data
  - Greater margin, lower generalization error
  - Points farther from the margin known with more confidence
- Multiclass using “one-vs-one” or “one-vs-all”



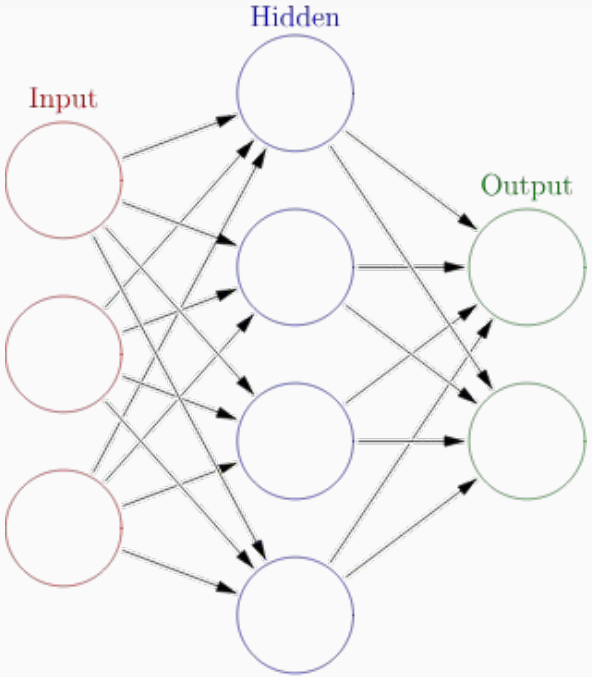
# Random Forest

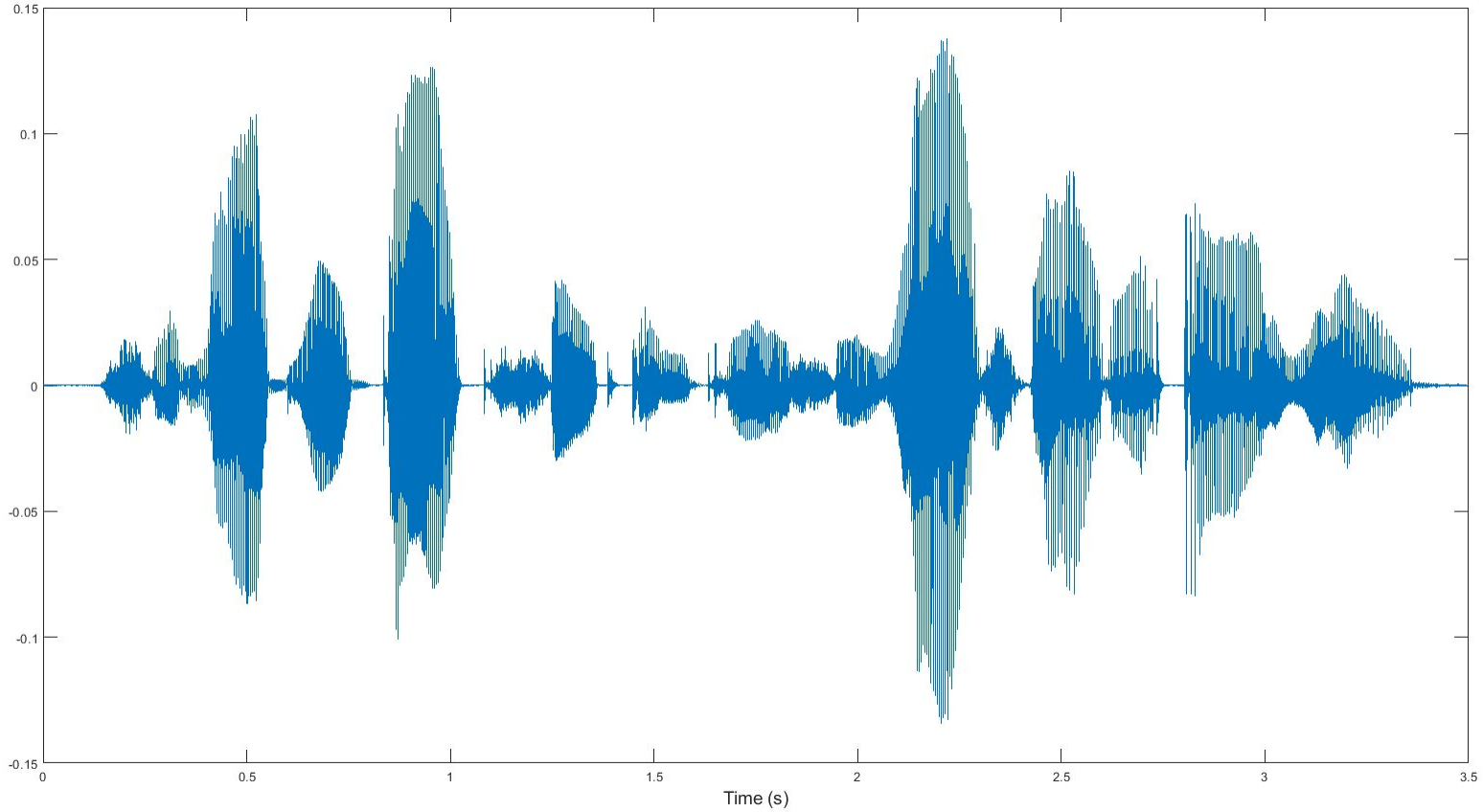
- Supervised Learning
- Creates N decision trees
- Test samples are run once on each tree
- The class is determined by a majority vote
- Provides confidence levels



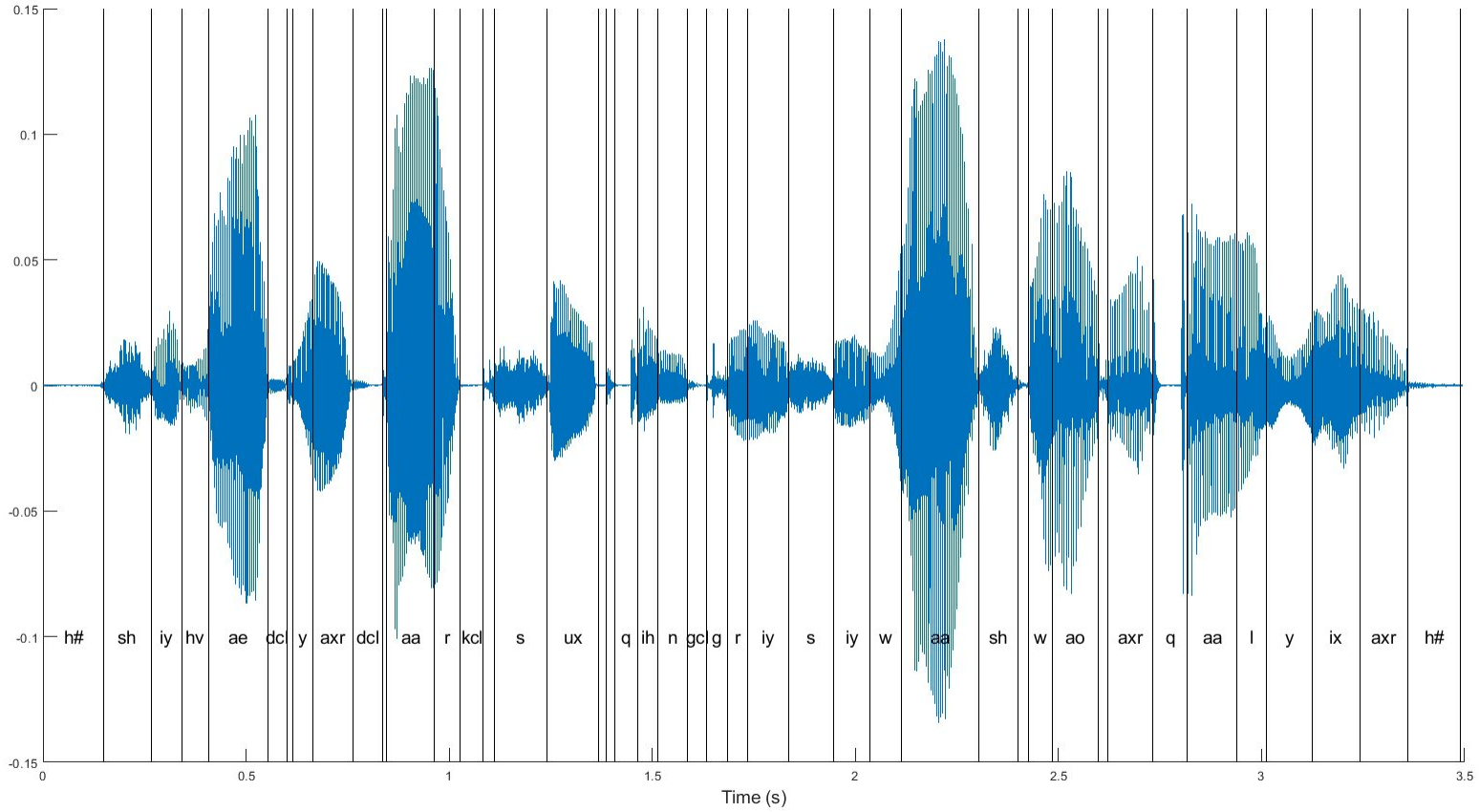
# Artificial Neural Network

- Inspired by biological neural networks
- Collection of connected nodes that transmit signals to each other
- Learned weights at each node affect the strength of transmitted signals

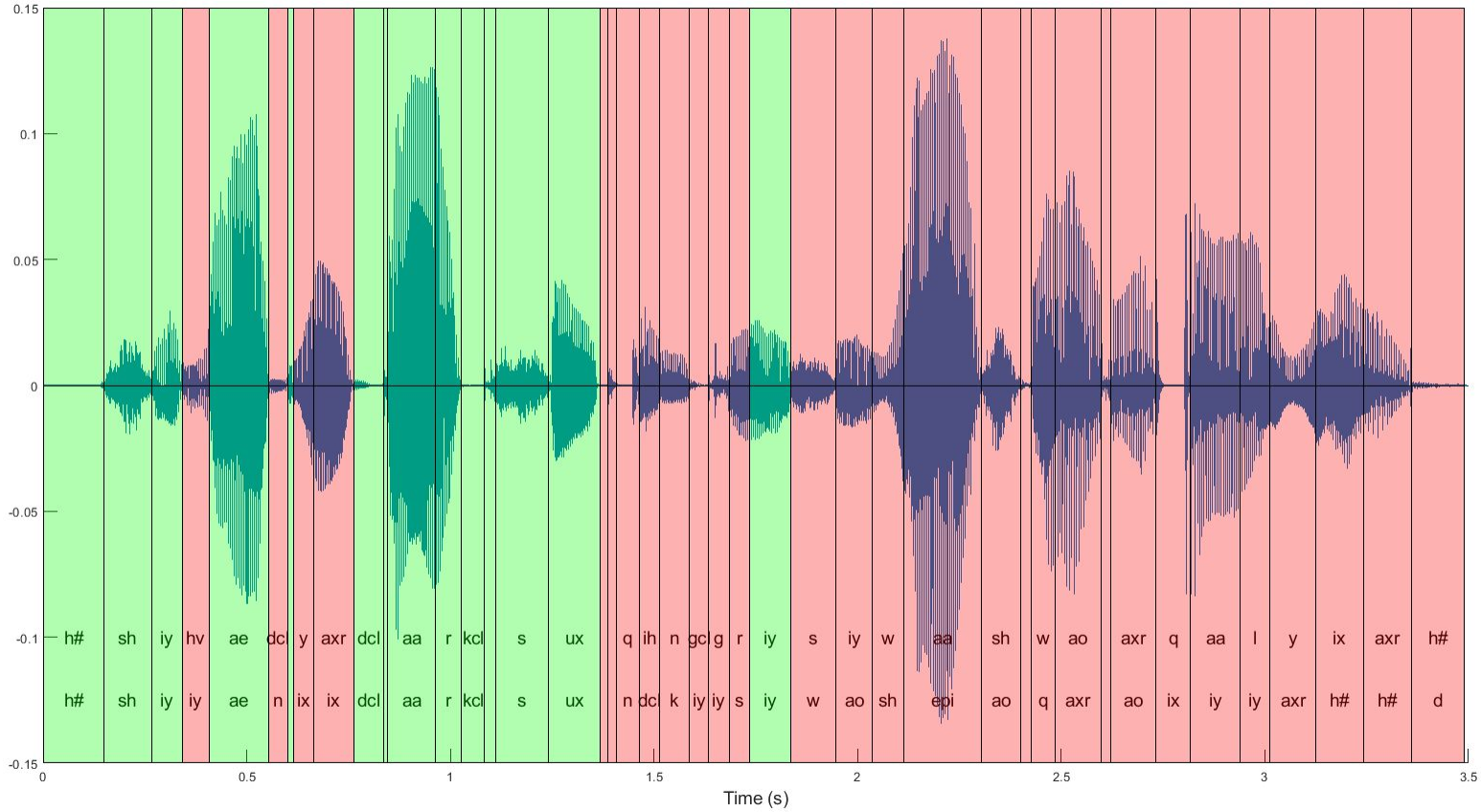





Example - speech audio signal



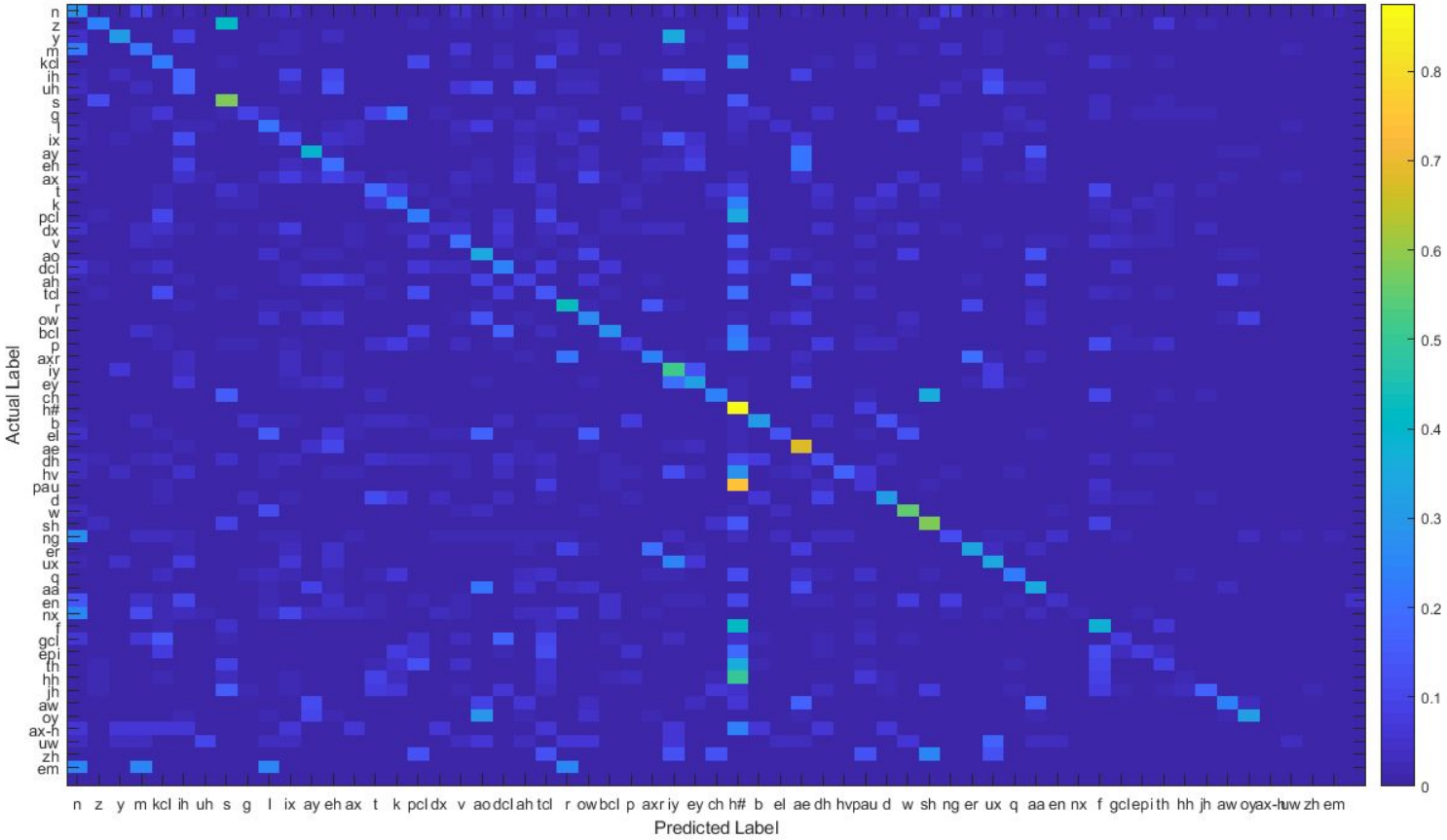

Example - phoneme divisions of speech signal



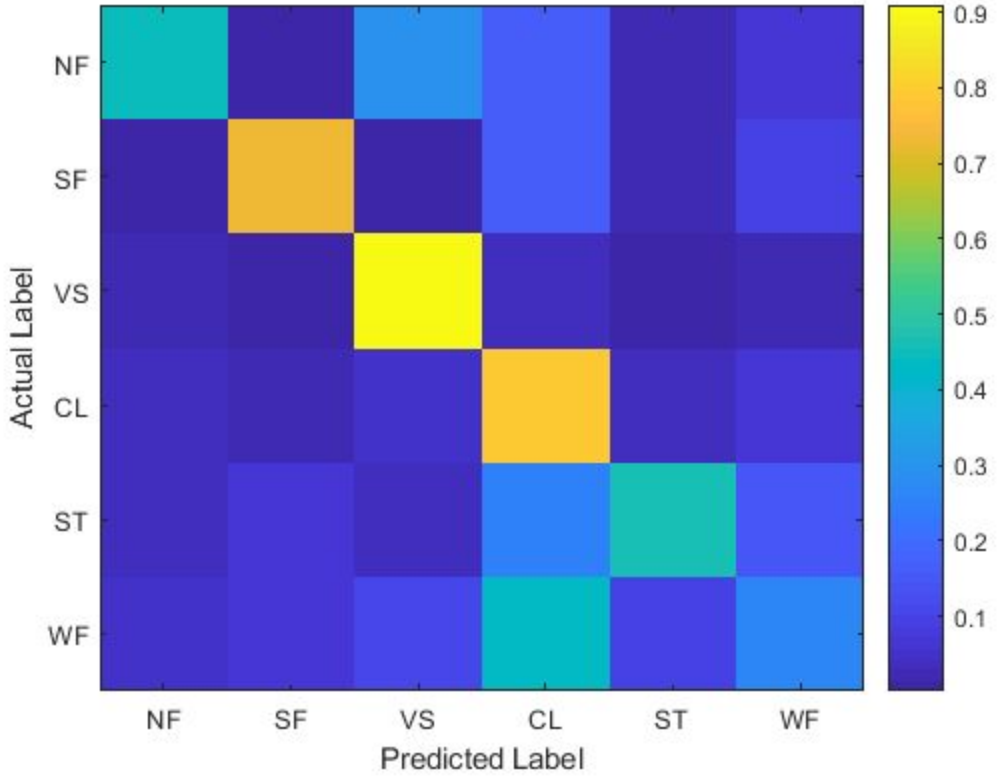

Example - phoneme classification

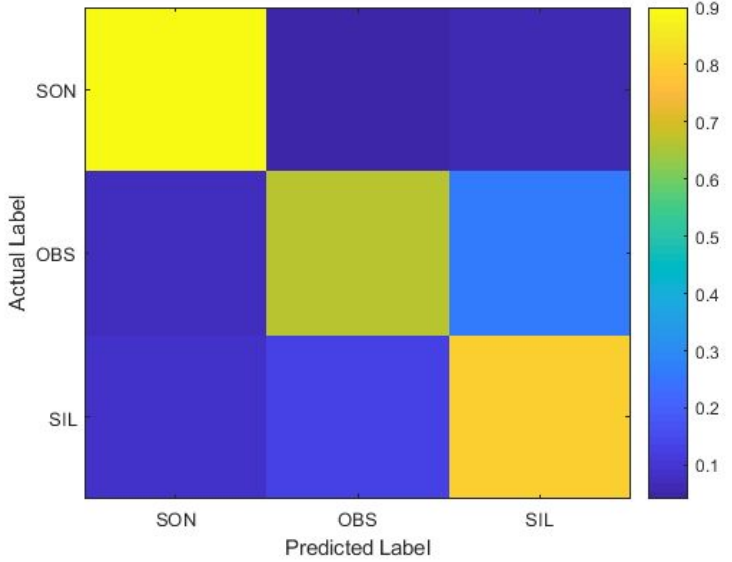
# Results - KNN

- Overall accuracy:
  - All 61 phonemes: 29.9%
  - Halberstadt 6: 73.5%
  - Halberstadt 3: 81.7%
  - Scanlon 74.6%
- Many mistakes with all phonemes, but okay with groups

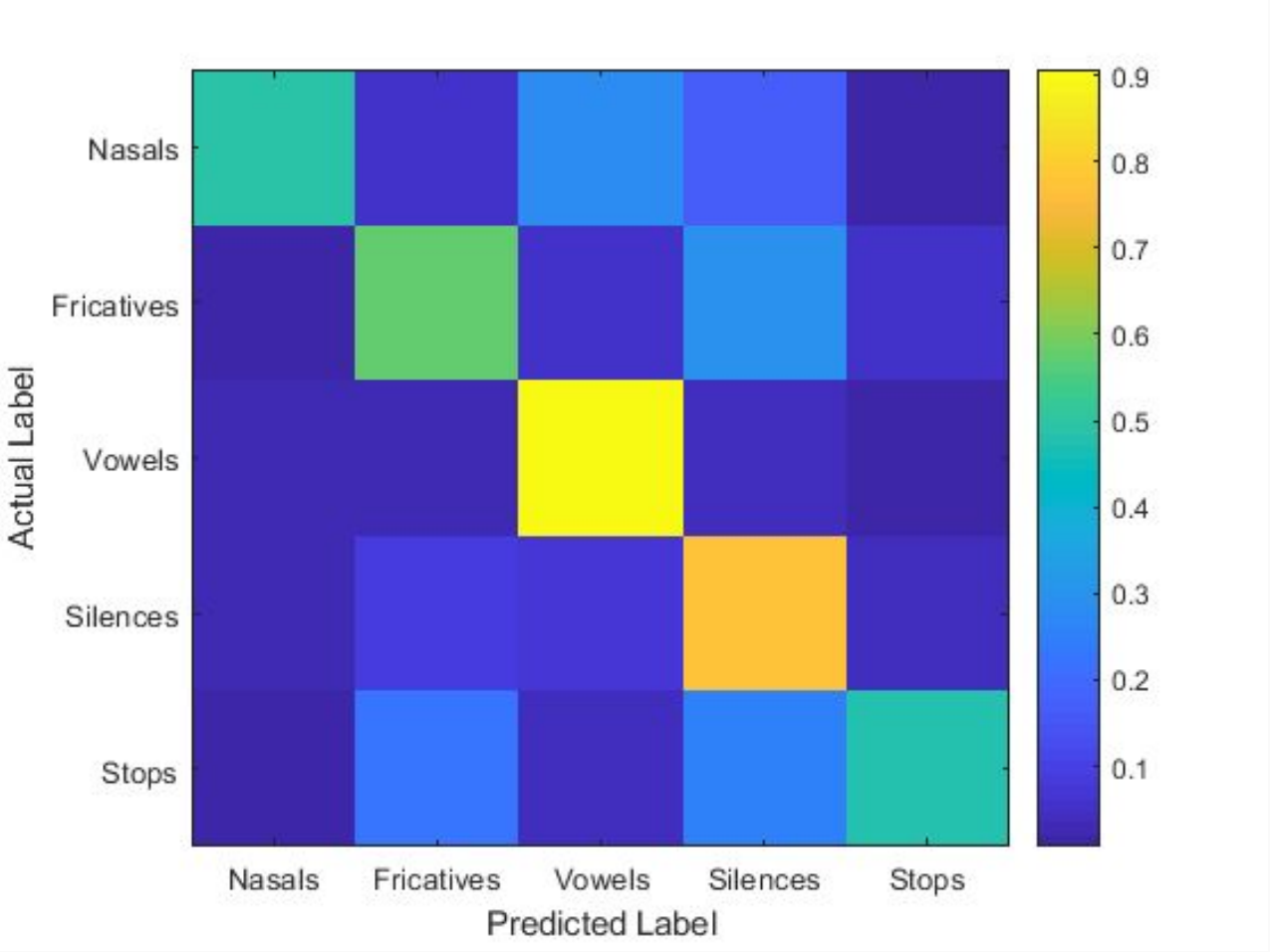



KNN classification - all phonemes



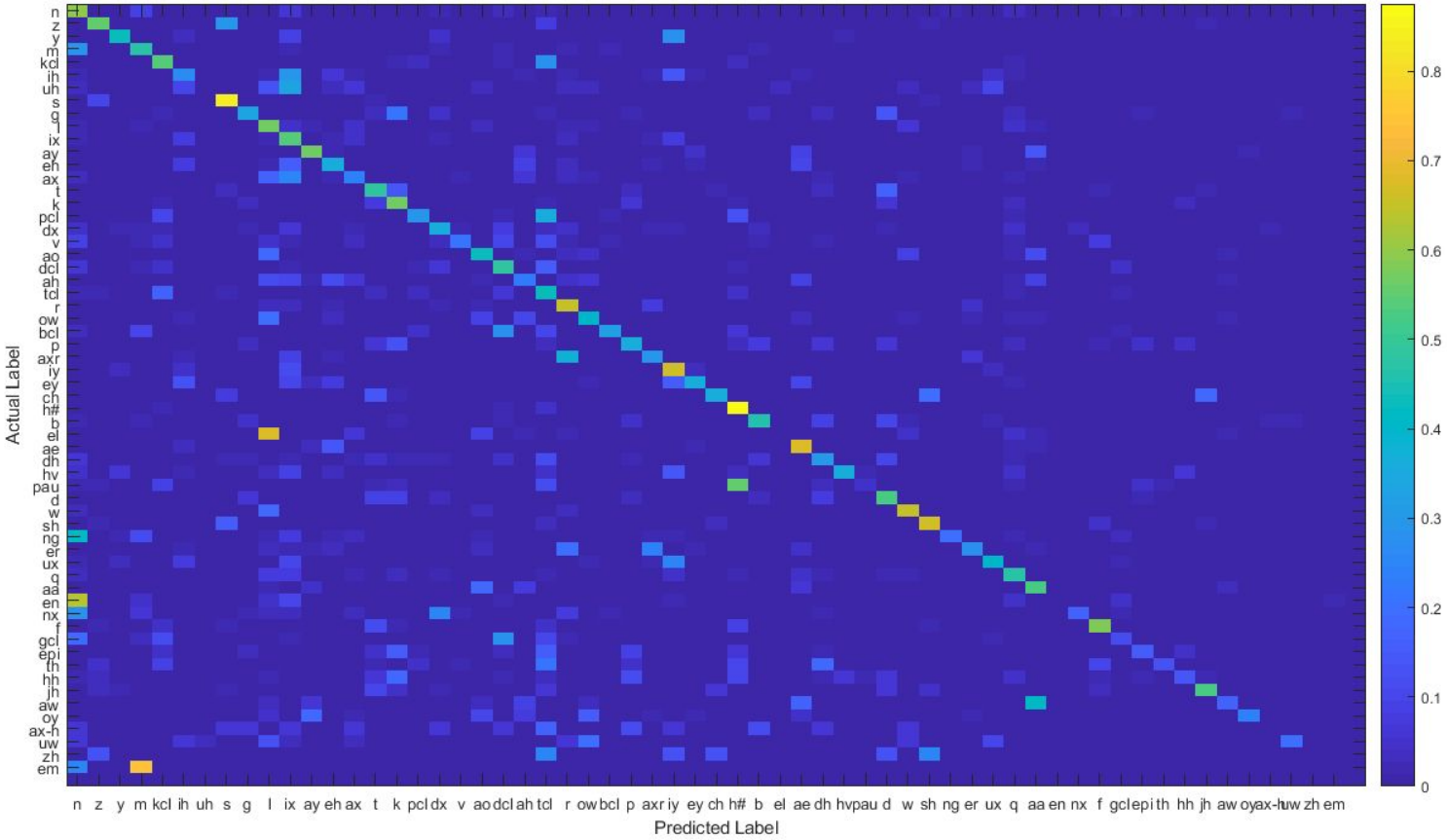

KNN classification - Halberstadt, 1998 groups



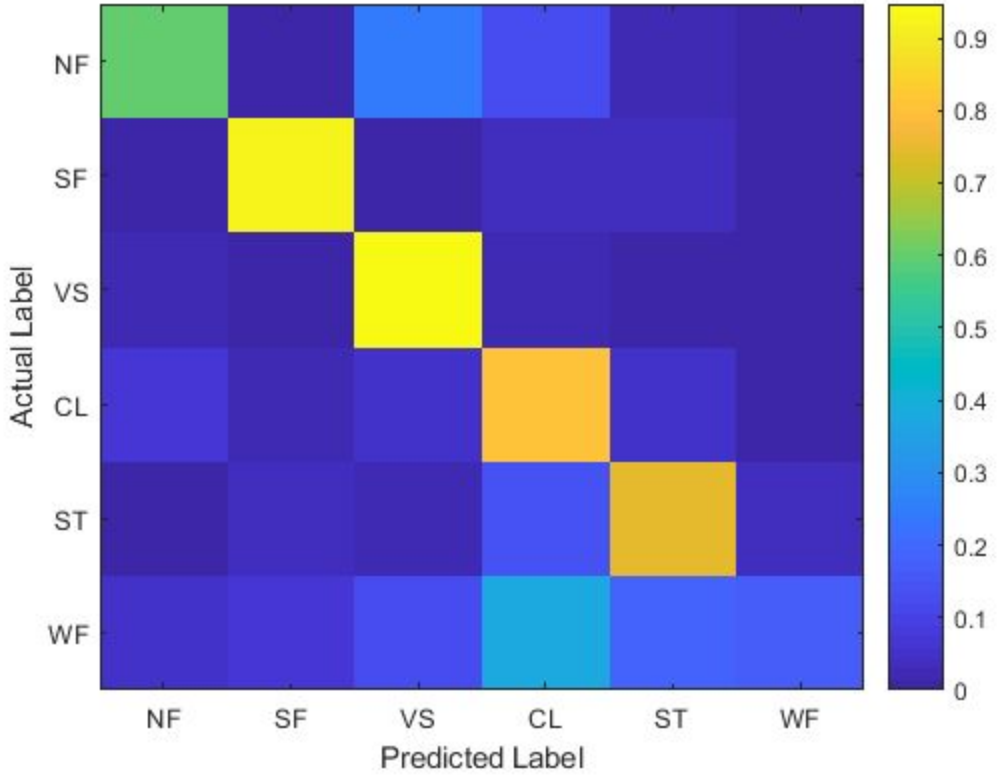

KNN classification - Scanlon, 2007 groups

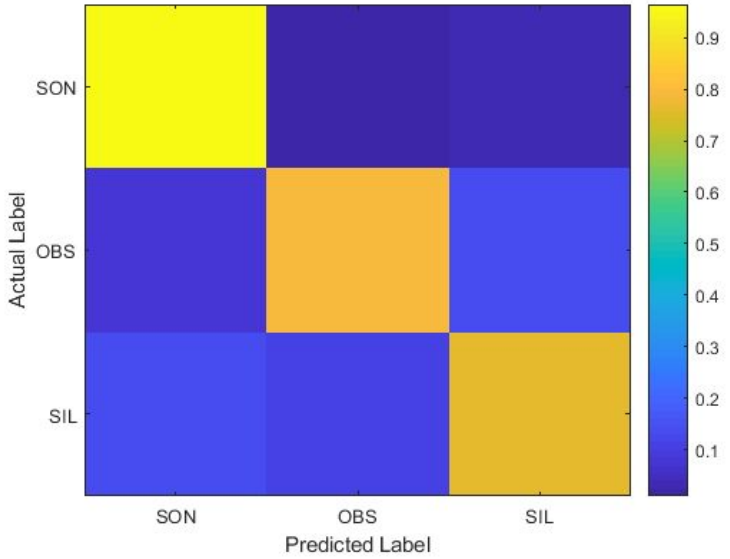
# Results - SVM

- Overall Accuracy:
  - All 61 phonemes: 48.8%
  - Halberstadt 6: 81.5%
  - Halberstadt 3: 87.7%
  - Scanlon: 81.5%
- Better than KNN but still not great for all phonemes

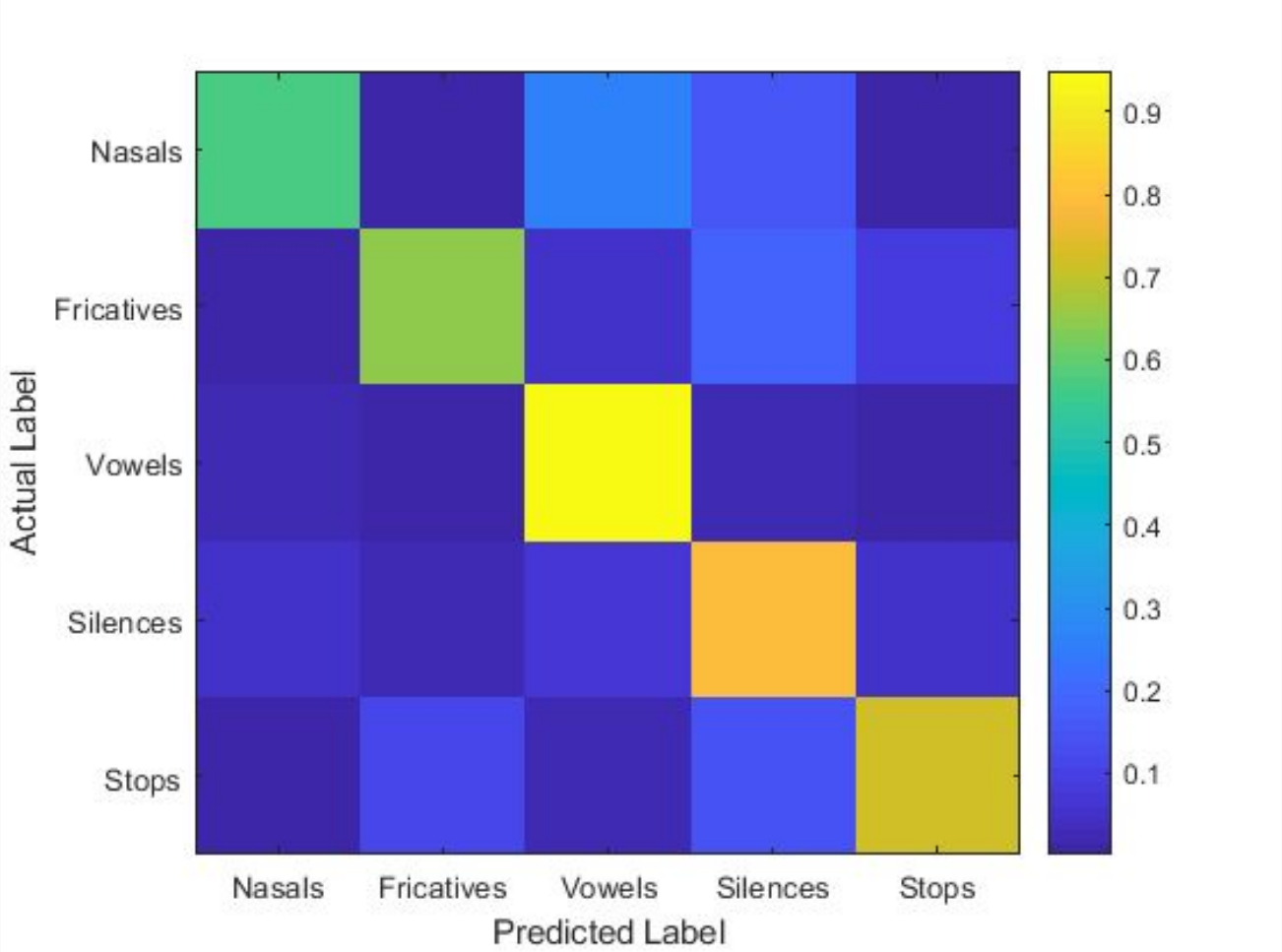



SVM classification - all phonemes



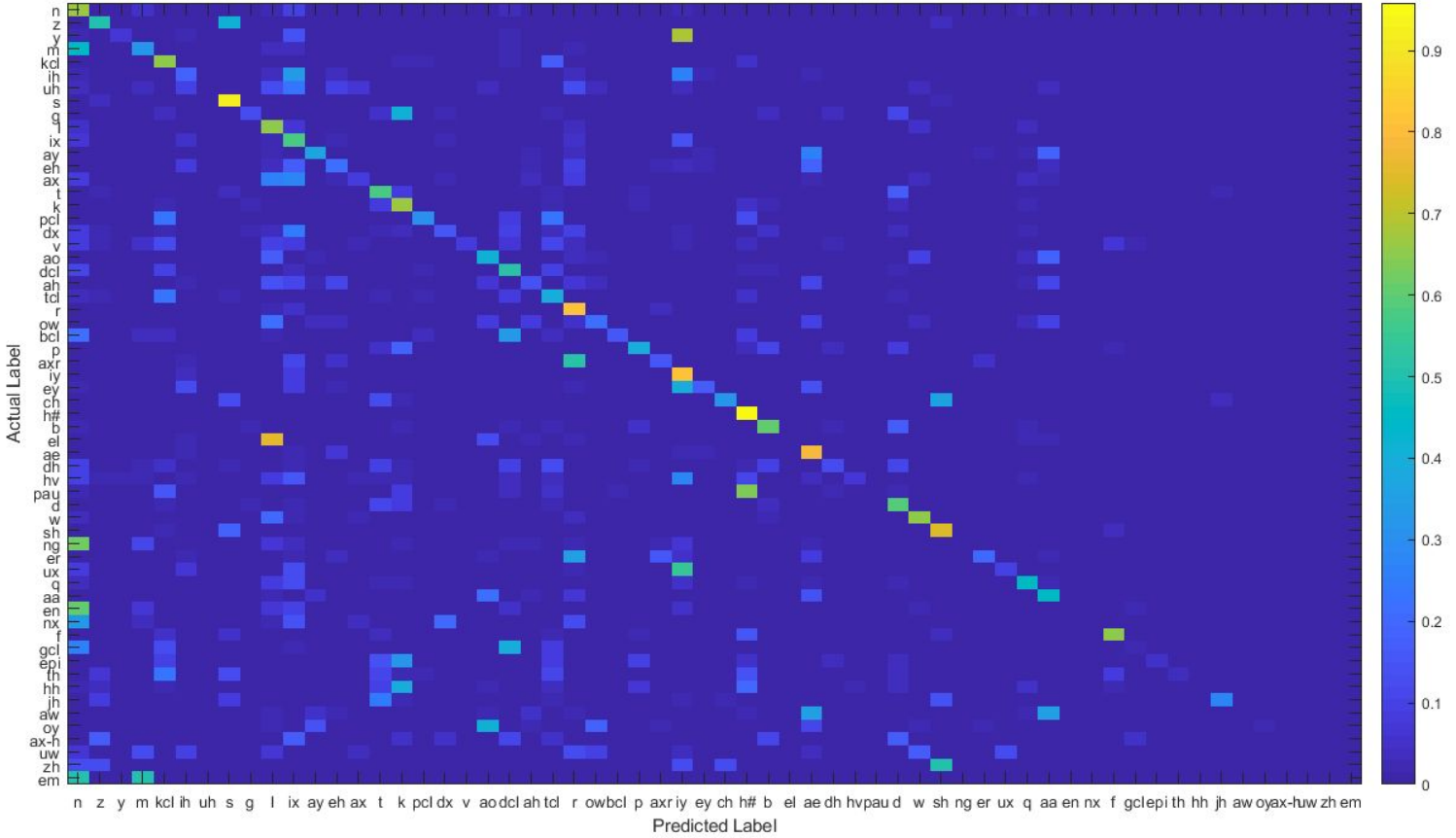

SVM classification - Halberstadt, 1998 groups



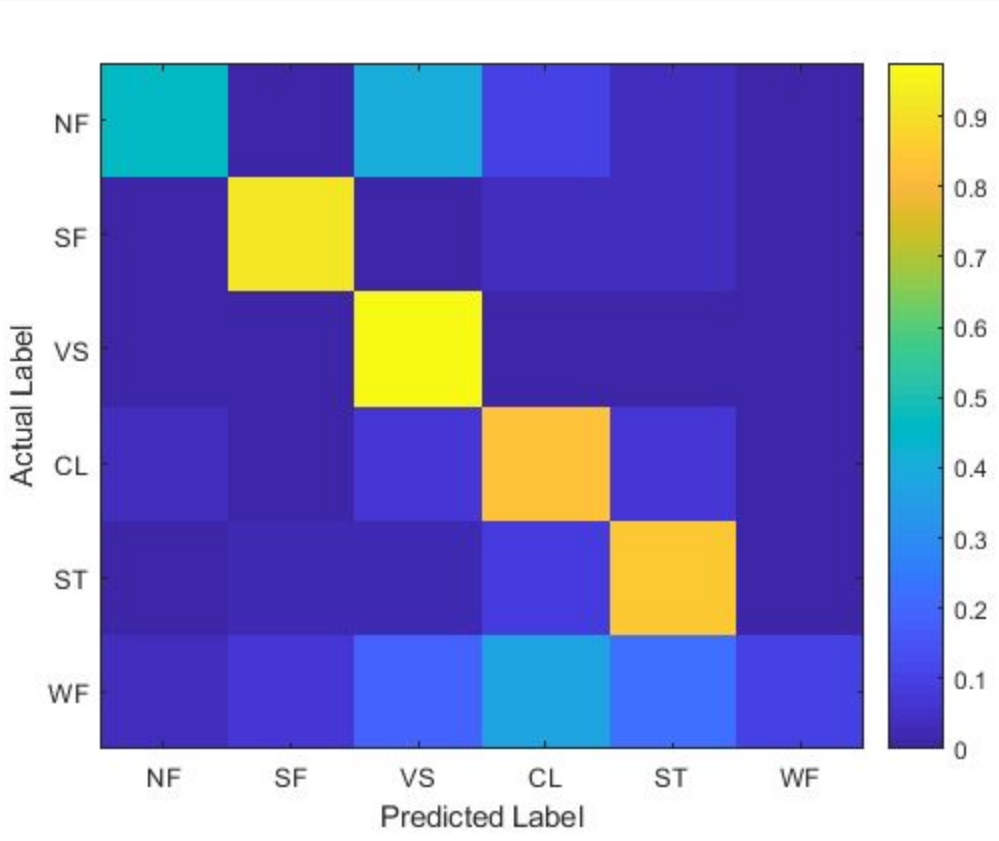

SVM classification - Scanlon, 2007 groups

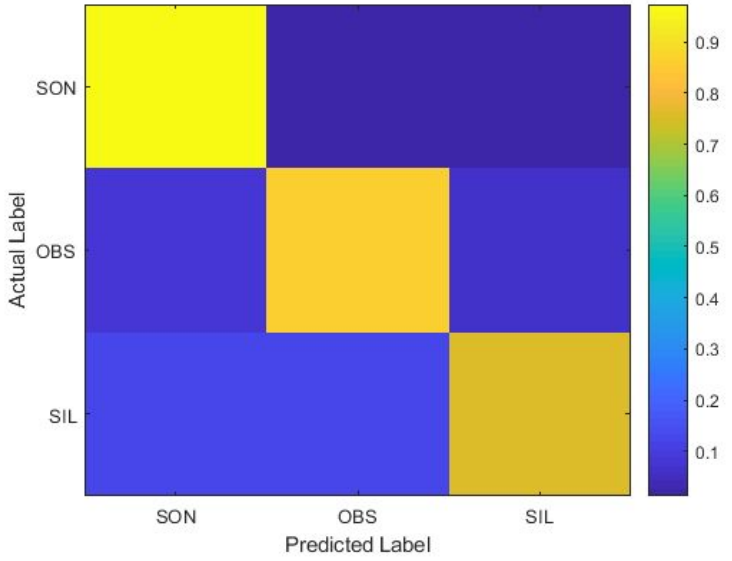
# Results - Random Forest

- Overall Accuracy:
  - All 61 phonemes: 47.3%
  - Halberstadt 6: 82.4%
  - Halberstadt 3: 89.6%
  - Scanlon: 83.5%
- Similar results to SVM
- Number of trees could be changed

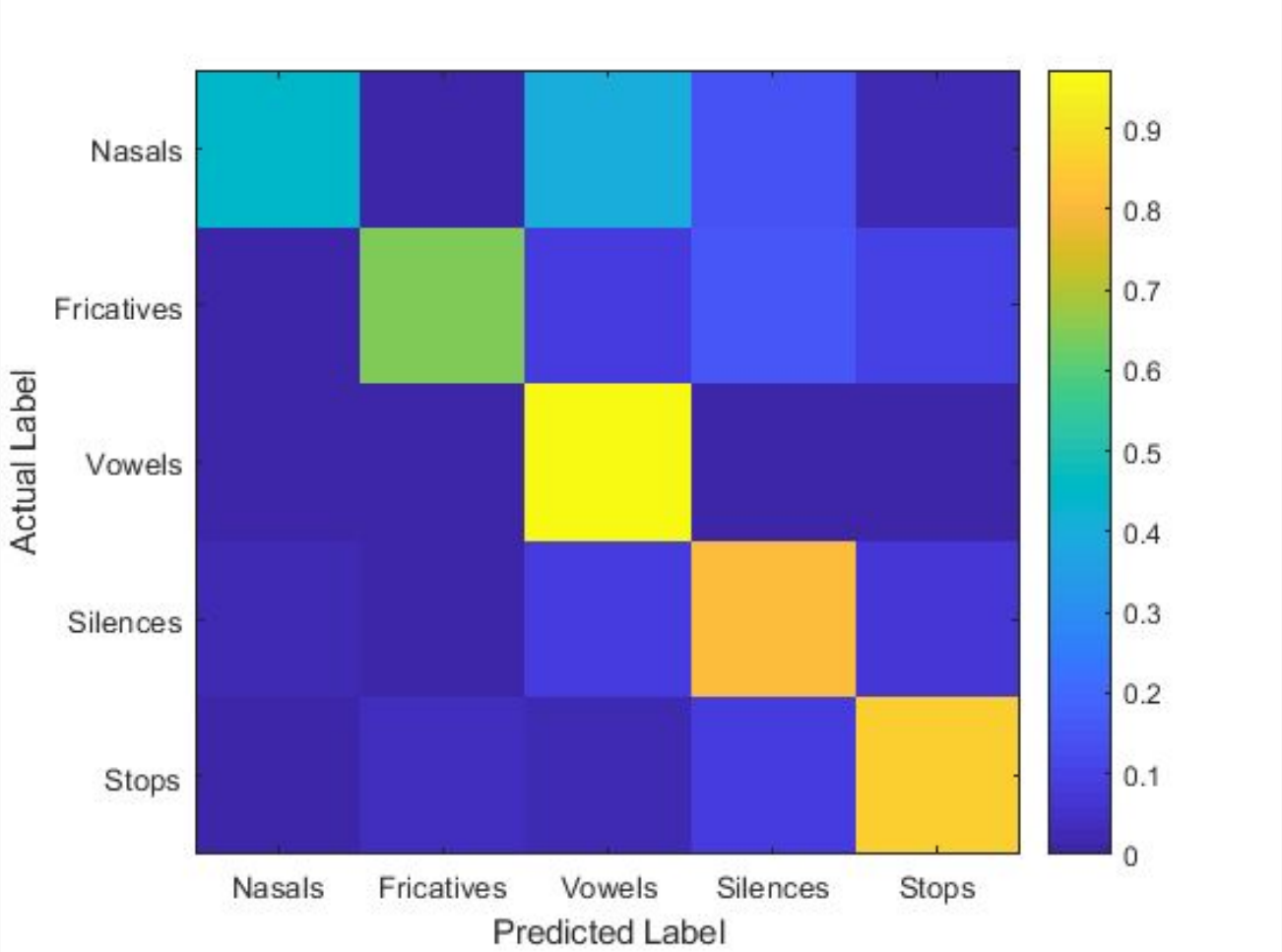



RF classification - all phonemes



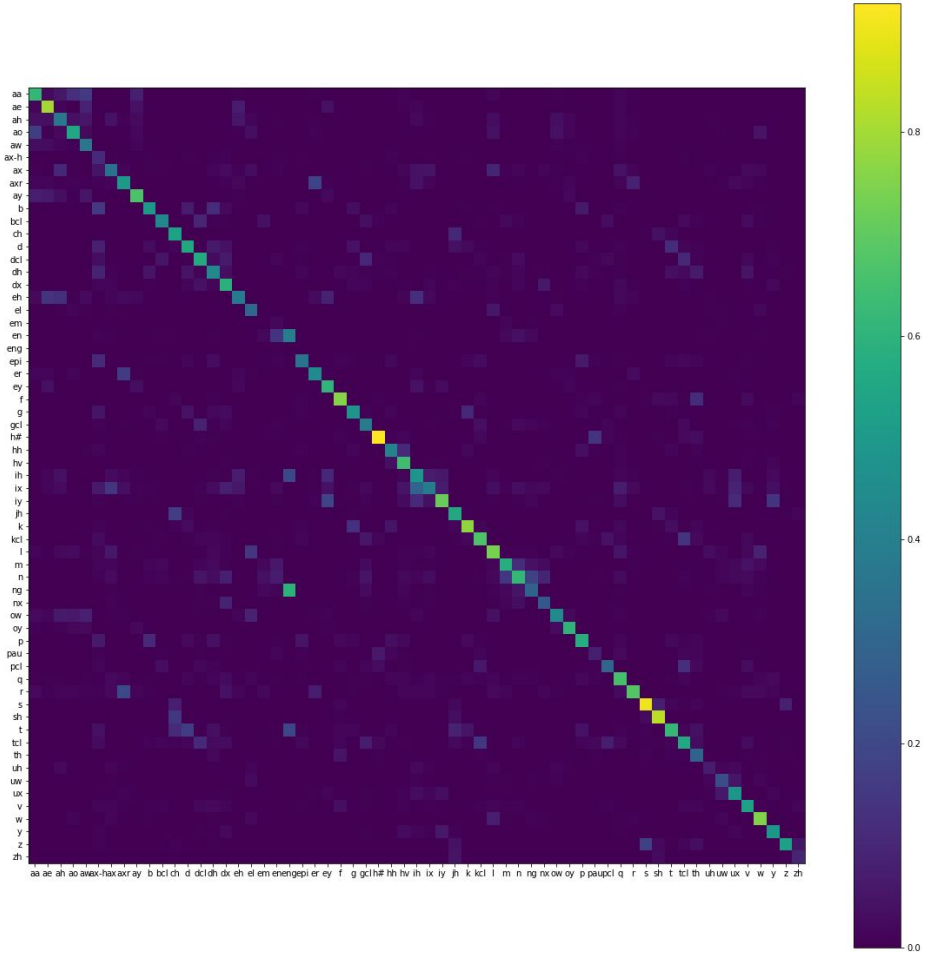

RF classification - Halberstadt, 1998 groups



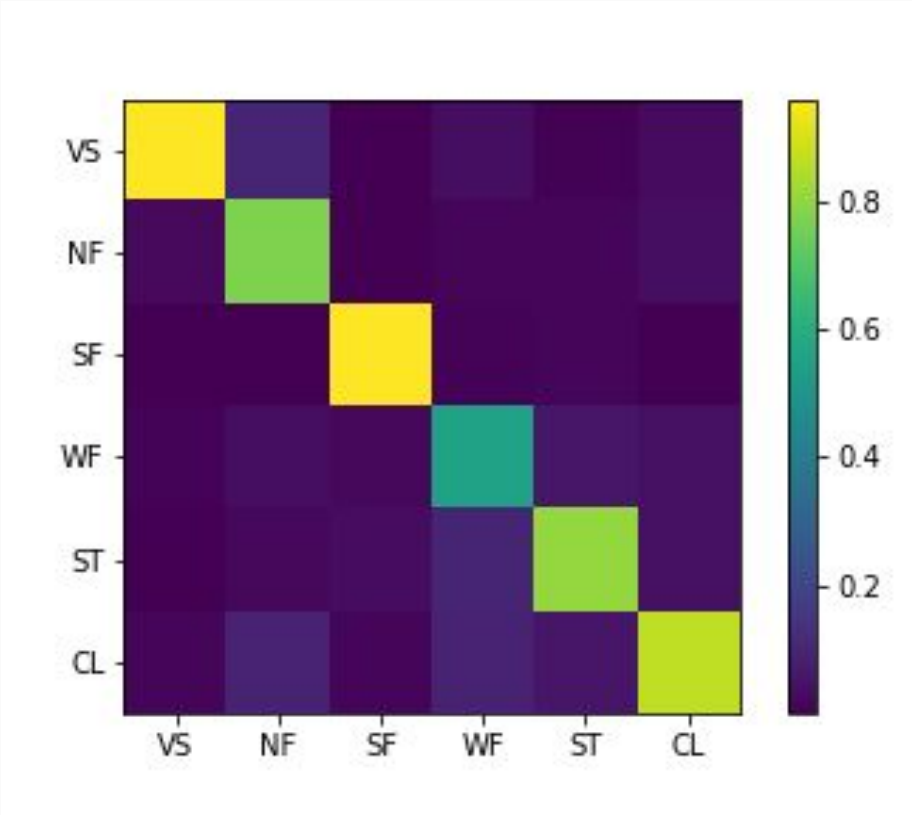

RF classification - Scanlon, 2007 groups

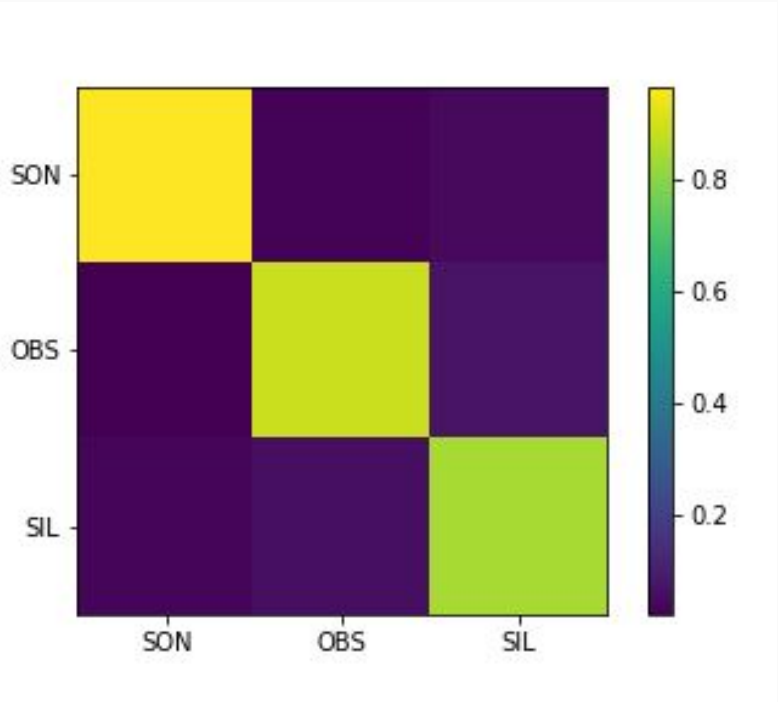
# Results - Neural Network

- Input: number of features, 72
- Output: number of classes (ie: 61 phonemes)
- Hidden: 512, sigmoid activation
- Overall Accuracy:
  - All 61 phonemes: ~59%
  - Halberstadt 6: ~88%
  - Halberstadt 3: ~92%
  - Scanlon: ~88%
- Best out of the tested methods

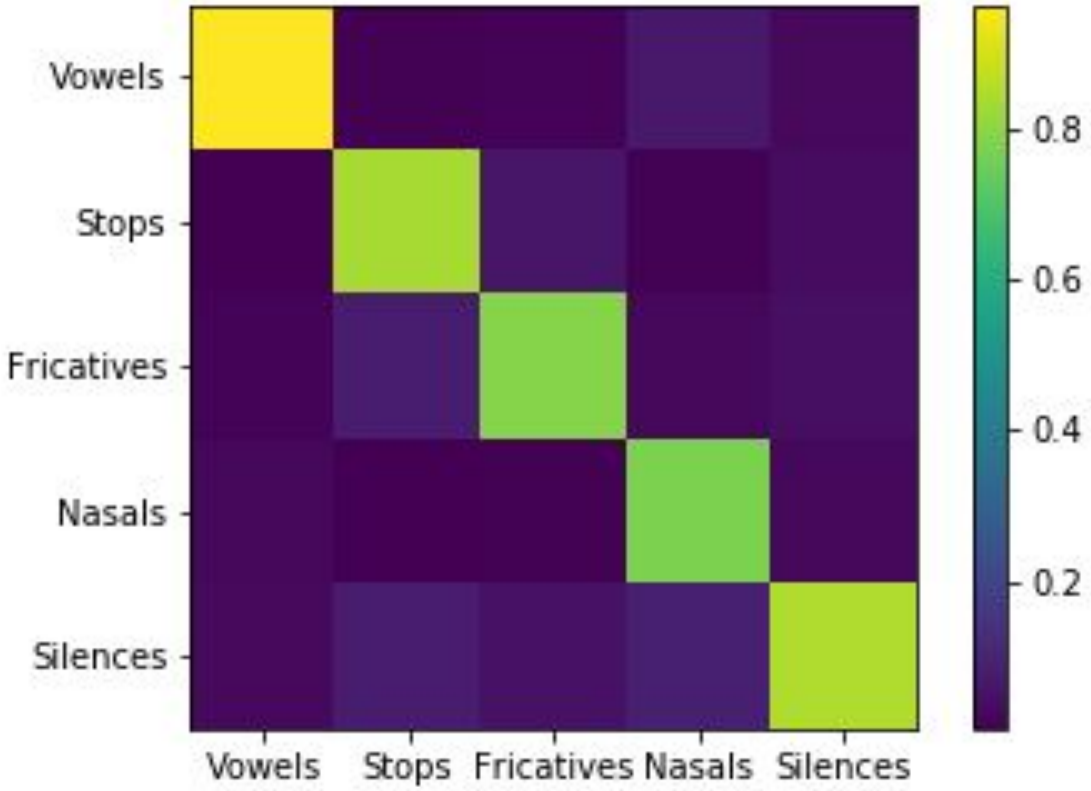



NN classification - all phonemes




NN classification - Halberstadt, 1998 groups




NN classification - Scanlon, 2007 groups

# Results - Comparison

- KNN has poorest overall accuracy
- SVM and RF have similar results, though RF parameters could be altered
- The simple neural network is not perfect yields the best results of these tested methods

# Conclusions

Possible improvements:

- Different MFCC parameters
- Different Delta and Delta Delta functions
- Different groupings
- Different Classifiers (Convolutional Neural Networks)

# References


[1] C. Lopes and F. Perdigao, “Phoneme recognition on the timit database,” 2011.

[2] K.-F. Lee and H.-W. Hon, “Speaker-independent phone recognition using hidden markov models,” 1989.

[3] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, “Phoneme recognition using time-delay neural networks,” 1987.

[4] A. Robinson and F. Fallside, Phoneme recognition from the TIMIT database using recurrent error propagation networks. University of Cambridge, Department of Engineering, 1990.

[5] A. A. Ali, J. van der Spiegel, and P. Mueller, “An acoustic-phonetic feature-based system for automatic phoneme recognition in continuous speech,” 1999.

[6] A. Graves, A. rahman Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” 2013.

[7] A. K. Halberstadt, “Heterogeneous acoustic measurements and multiple classifiers for speech recognition,” Ph.D. dissertation, Massachusetts Institute of Technology, 1999.

[8] P. Scanlon, D. P. Ellis, and R. B. Reilly, “Using broad phonetic group experts for improved speech recognition,” IEEE transactions on audio, speech, and language processing, vol. 15, no. 3, pp. 803–812, 2007.