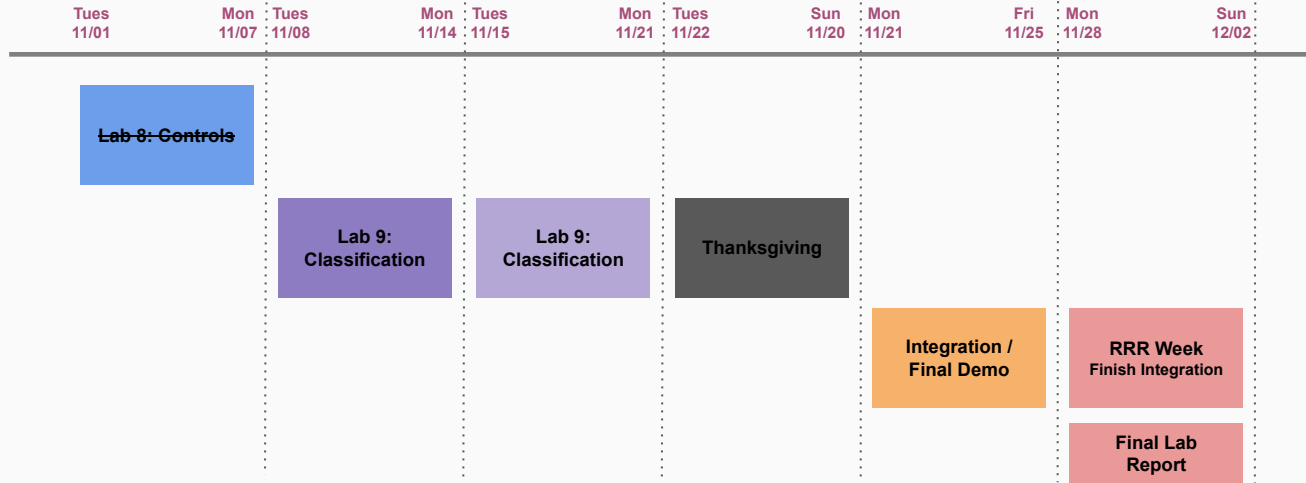


Lab 9: Classification

EECS 16B Fall 2022

<https://links.eecs16b.org/lab9-slides>

Lab Schedule



Classification Lab!

Note: PCA concepts are relatively new and was covered in Tuesday lecture this week, so please pay close attention to the presentation & ask any questions as needed!

Today's Lab

- Read [lab note](#) before starting lab!
- Long lab!
 - a. Front end verification
 - b. Voice data collection - record 6 different words!
 - c. Data preprocessing
 - d. Use SVD and PCA to find PCA basis to project data onto
 - e. Find mean clusters to distinguish between words
 - f. Implement and test classifier
 - g. Tuning Hyperparameters for Best Performance - achieve 80% accuracy on 4 chosen words
 - h. Arduino implementation of PCA Classify to classify words live

Part 1: Mic board Circuit Verification

- Check the voltage at each node of mic board:
 - VDD: 3.3 V
 - VSS: Ground
 - OUT: centered ~ 1.65 V, 2-3 Vpp
 - OS2: ~ 1.65 V
 - OS1: $\sim < 1.65$ V
- Check signal at front end audio circuit output:
 - Output of non-inverting amplifier for low-pass filter
 - Centered ~ 1.65 V
 - ~ 2.5 V Vpp when making noise at mic

Part 2: Data Collection and Speech Patterns

- SIXT33N will know four voice commands, we record 6 to be safe!
- Will go straight far, straight close, turn left, turn right

Word Guidelines

- Speech pattern recognition, not word recognition
 - Actual word doesn't matter, but the speech pattern for the word does!
- Use words with different syllables (but still within the 2 seconds)
 - e.g. Pear, apple, banana, watermelon
- Enunciate syllables well to make clear distinctions
- Try words with different endings (*hard* – back vs. *soft* – shoe)
- Remember the way you say each word, as you'll have to replicate it later!
 - Can record yourself with your phone so you have a record of how you said each word

Data Collection Guidelines

- Try to say each recording the *same way* at the *same distance* from the microphone with the *same timing* (within 2 second recording window)
- The Arduino LEDs will count up when recording, indicating when you should speak
- Timing is important: Say the word *just* after the first LED turns on
- You will only be powering your Arduino **through the USB** this week
 - Ensure that you are using the PSU for front-end power, and not your batteries

Data Collection Guidelines (continued)

- This is a long lab, don't spend too much time choosing words
 - We recommend proofing those words by a TA/ASE to make sure they are distinct enough
- Collect ~40-45 recordings for each of 6 words
- Delete outliers to have exactly 40 recordings per word
- Don't spend too much time eliminating outliers!
 - Can check for outliers when making line plots of .csv file in Excel
 - If certain recordings don't stand out, delete from the start/end until you reach 40

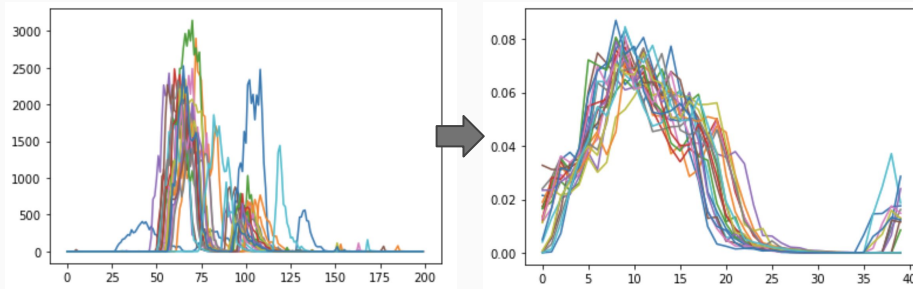
Part 3: Data Preprocessing/ Word Alignment

Threshold - percentage of the max value for the sample counts as a spoken command

Pre-length - how many timesteps before we hit this threshold did we start speaking the word

Length - how long the sample is

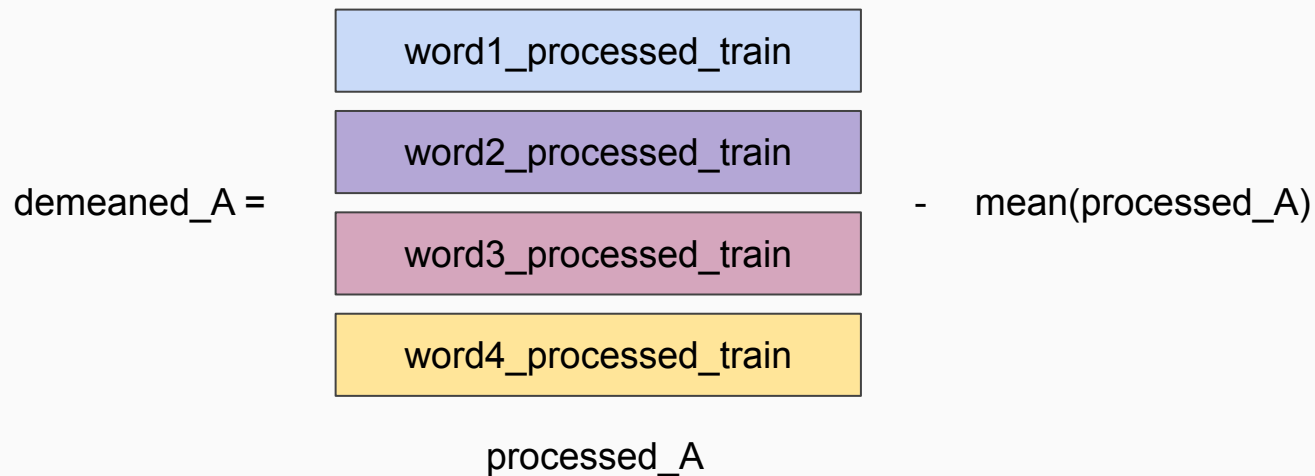
- What is length for the example to the right?



Word example: "meat"

Part 4: Computing SVD on our Data

- We split our data into train and test, with training being 70% of the samples for each word
- We pick the best alignment using the training data
- We stack the aligned words in a data matrix
- We then demean using the mean of each timestep (each “feature”) before performing SVD



SVD and PCA

PCA = Principal Component Analysis

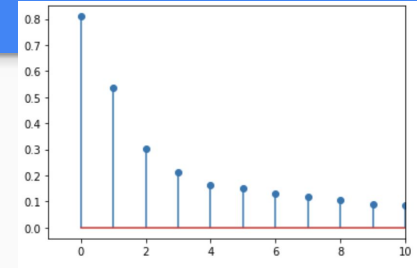
SVD = Singular Vale Decomposition

SVD gives us a way to find the principal components of a data set

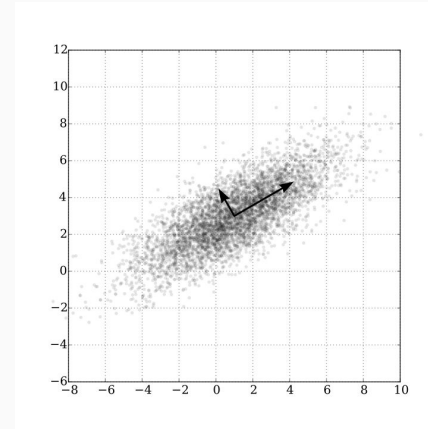
- Principal components: basis vectors that maximize variation
- The ones corresponding to the largest sigma/singular values
- Fewer dimensions = easier to work with on Arduino/Low Power computer

Numpy has useful functions for us:

- `numpy.linalg.svd()`



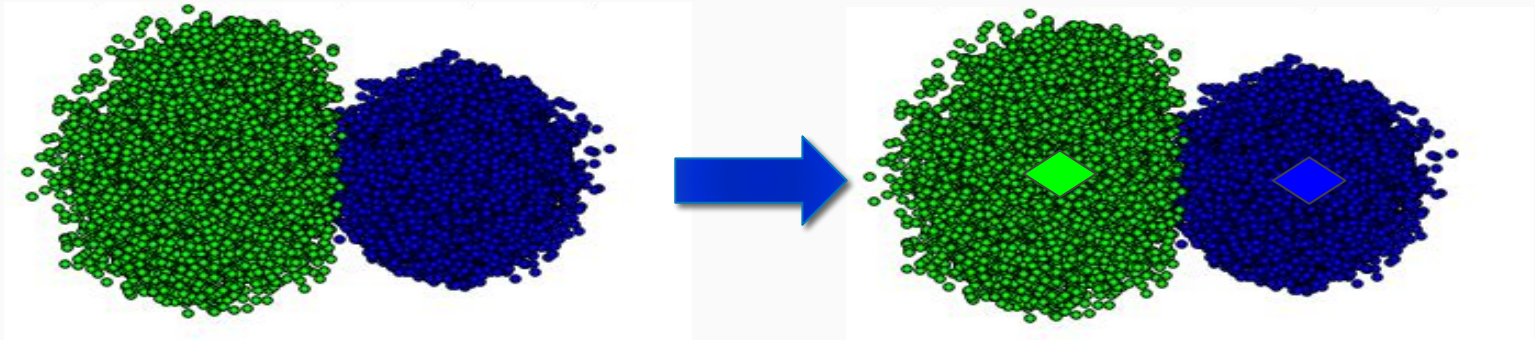
Sigma Values Example



Principal Components of Data Example

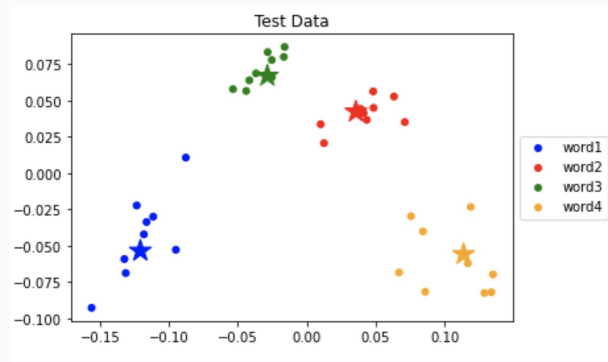
Part 5: Mean Centroid Classification

- Already have labelled data
- Project all data onto axes of most variation (PCA basis vectors)
- Take mean of each projected word to find centroid for that word
- Classify new, unlabelled data by projecting it onto same basis and finding which centroid it is closest in Euclidean distance to



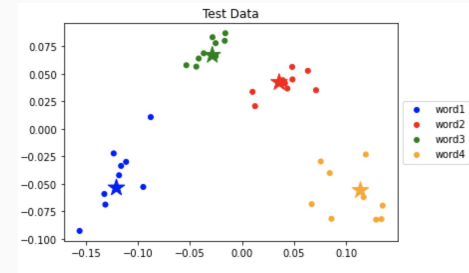
Part 6: Testing your classifier + Tuning

- We saved 30% of our data as test data
- We know the labels, so we
 - Project test data onto our PCA vectors, subtract projected mean vector
 - Assign to closest cluster in 3D space
- Aiming for 80% accuracy



Part 7: Arduino Implementation

- EUCLIDEAN_THRESHOLD
 - How far away from the centroid is considered noise?
 - Refer to centroid plots from SVD/PCA for reasonable values
- LOUDNESS_THRESHOLD (value in 10s - 100s)
 - Ensure that background noise isn't being classified
- If the word isn't classifying properly, you can add print statements to help debug what's happening in classify.ino!



Tips, Tricks, and Warnings

- Make sure the pins used in your code are the same as on your Arduino!
- Do NOT plug in your batteries in this lab
- You have free reign over choosing your words, but choose them in an educated manner according to the guidelines!
 - You don't want to have to keep recording words because they sound too similar
 - Keep your words appropriate, as you will have to say and use them in front of course staff in the final Integration lab's final demo
- Submit a help request if you encounter a **COM port issue where it is not being recognized!**

Lab 9 Checkoff

- Requirements for checkoff:
 - Have all questions answered, code, and plots completed in ipynb
 - Show min 80% classification rate on all 4 chosen words
 - `classify.ino` code and demonstrate all 4 words classify correctly live on the Arduino
 - Be prepared to answer conceptual checkoff questions!

Forms & Information

- Help request form: <https://eecs16b.org/lab-help>
- Checkoff request form: <https://eecs16b.org/lab-checkoff>
- Extension Requests: <https://eecs16b.org/extensions>
- Makeup Lab: <https://makeup.eecs16b.org>
- Slides: <https://links.eecs16b.org/lab9-slides>
- Anon Feedback: <https://eecs16b.org/lab-anon-feedback>
- Lab Grades error: <https://links.eecs16b.org/lab-checkoff-error>