ACOUSTIC MODELING PART I

Many illustrations provided by courtesy of James Glass

What we learned so far

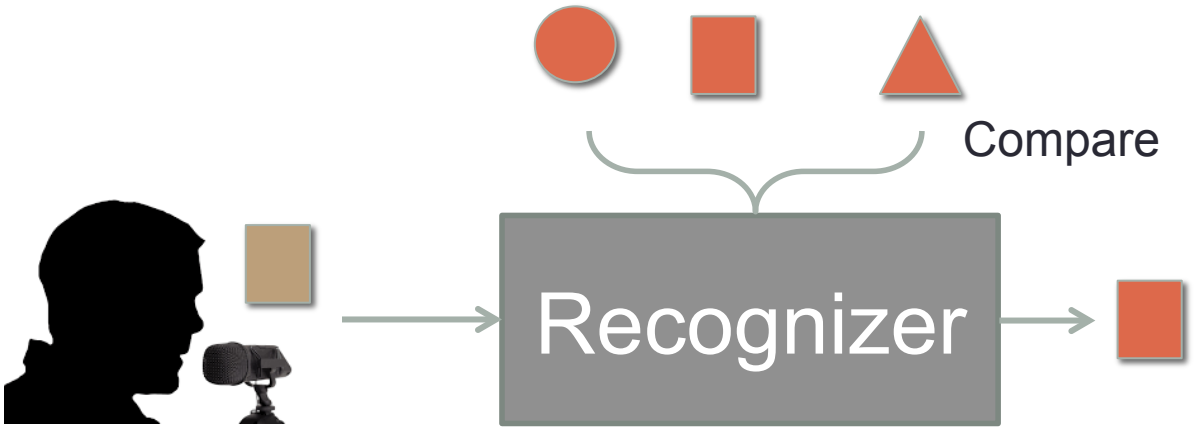
- How speech is produced?
  - Source-Filter Model
- The concept of Short-Time Fourier Analysis
  - Small slice that changes over time
- How to differentiate between speech sounds?
  - Formants
- Speech is continuous
  - Effect of one sound on sounds around it
  - The same sound can be very different

A simple recognizer/classifier

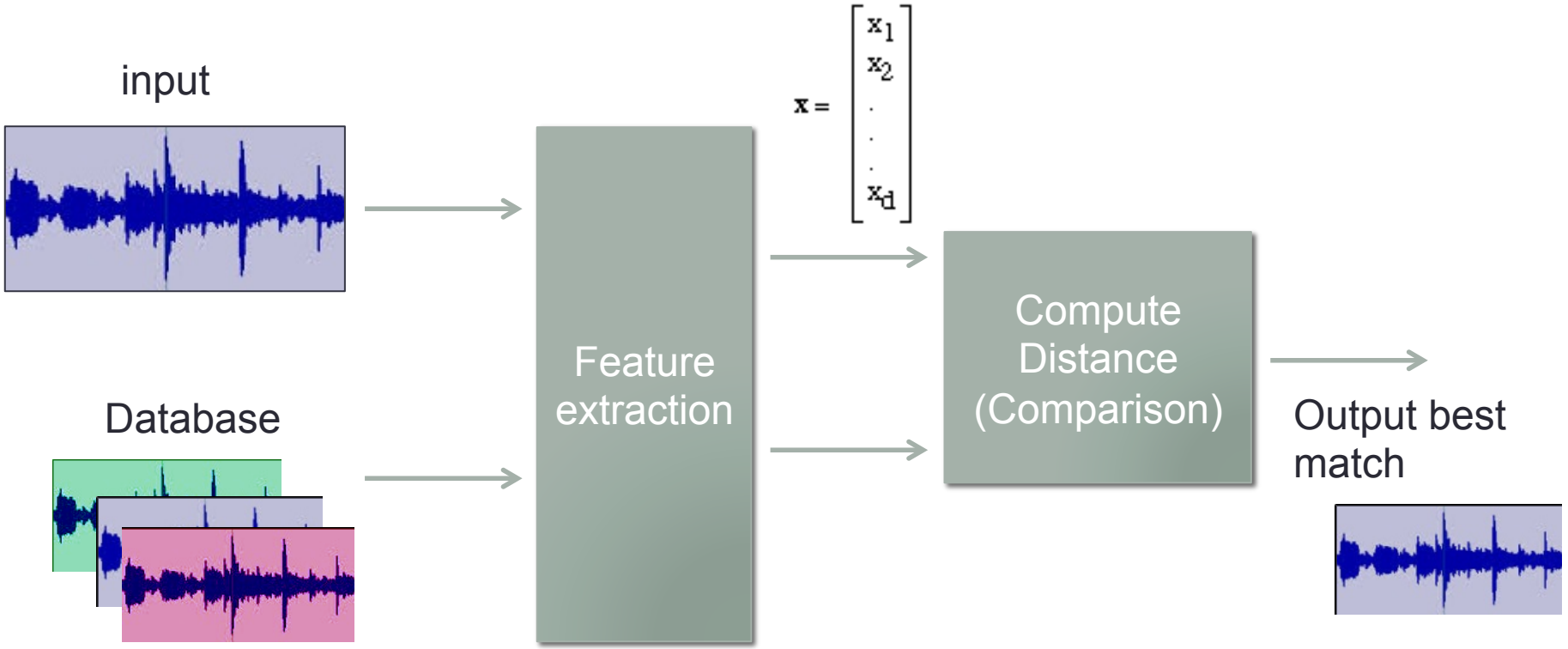
- Template based recognition
  - Consult database
  - What's "closest" is the answer

$$W^* = argmin_W D(W|X)$$

X - waveform, W - words



A classification framework



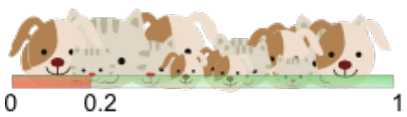
Called Nearest Neighbor method

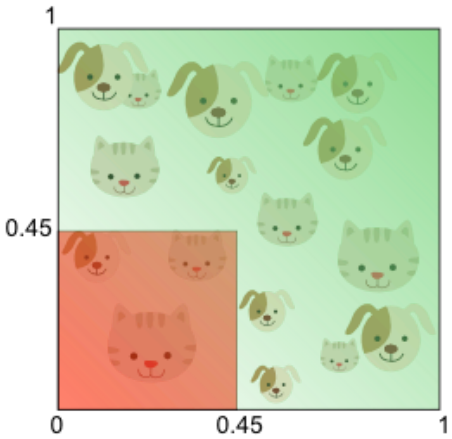
Feature extraction

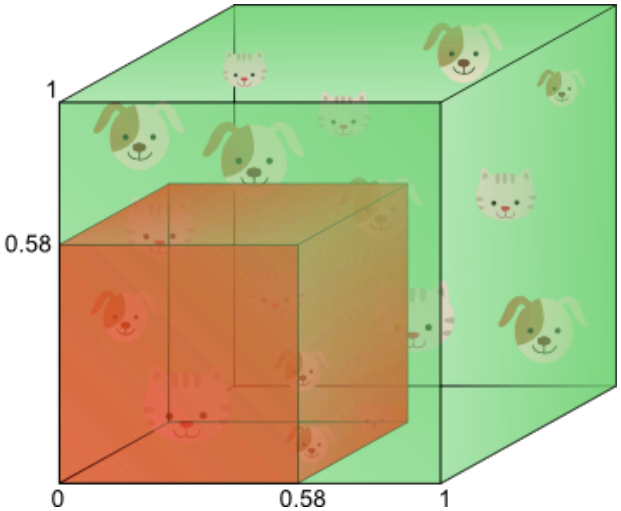
- Goals
  - Remove unwanted information
  - Reduce dimensionality
- Good features should
  - Have all the necessary information
  - Invariant to other effects
  - Match with the assumption of the machine learning model

Curse of dimensionality

- Harder to visualize or see structure of data – More training data required
  - Verifying that data come from a straight line/plane needs n+1 data points
- Hard to search in high dimension – More runtime
  - Search for a dot in
    - 1 m line
    - 1 m^2 square
    - 1 m^3 cube







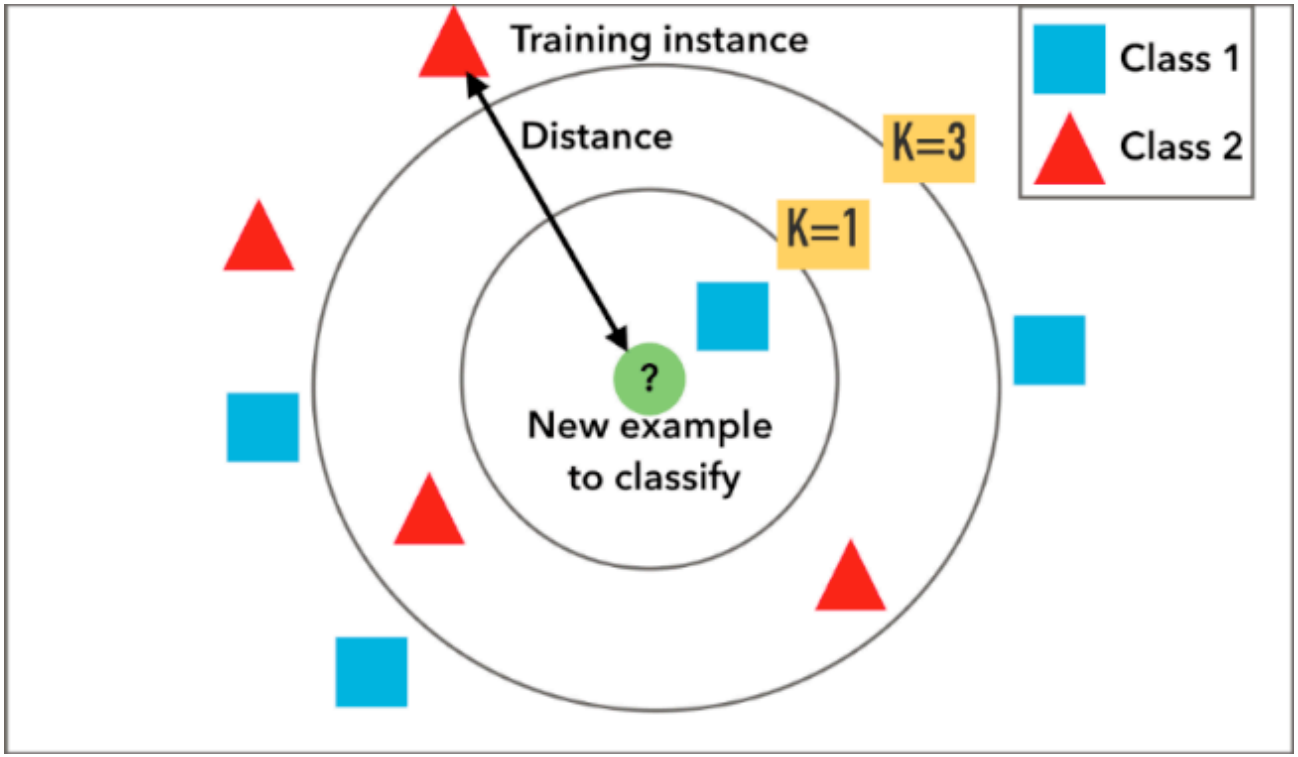
http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/

Nearest Neighbor Classifier

- The thing most similar to the test data must be of the same class

Find the nearest training data, and use that label

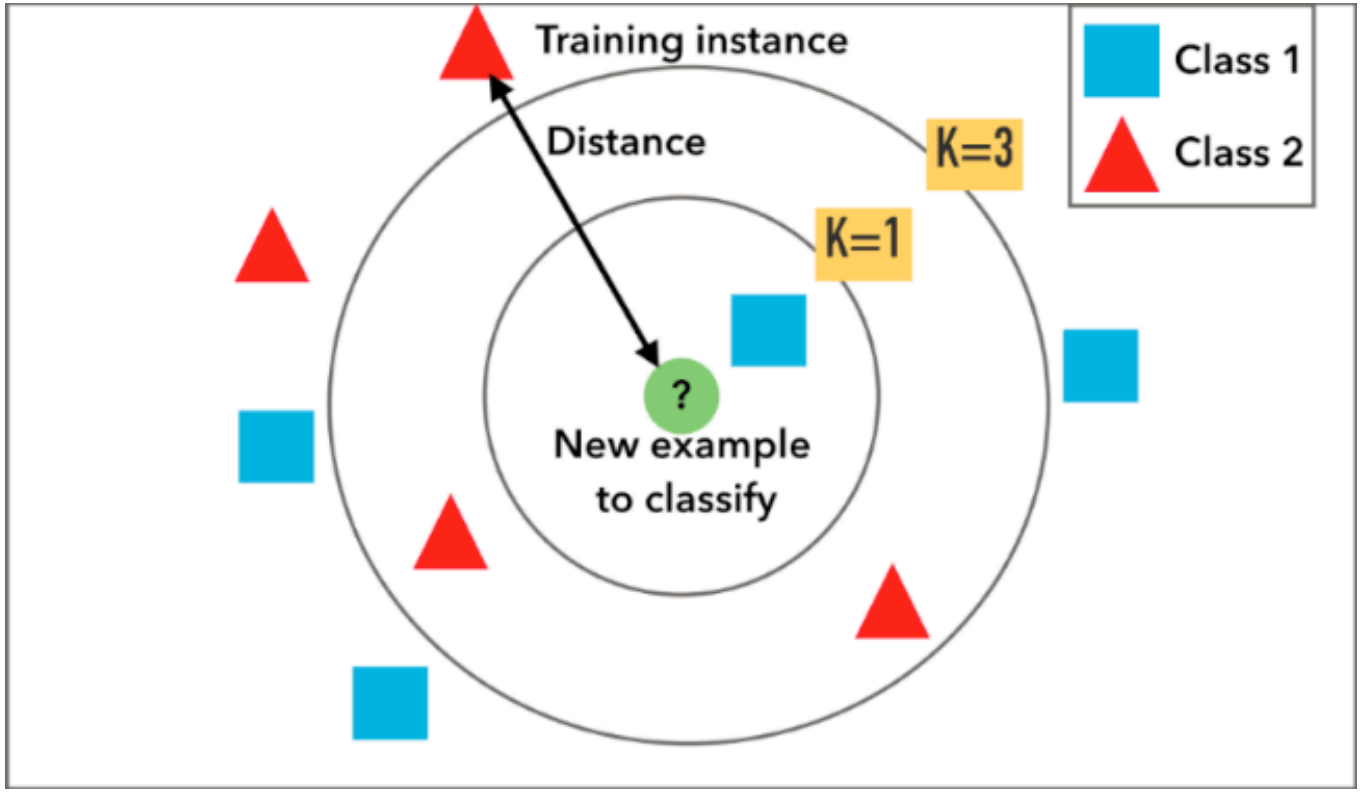
- Use “distance” as a measure of closeness.



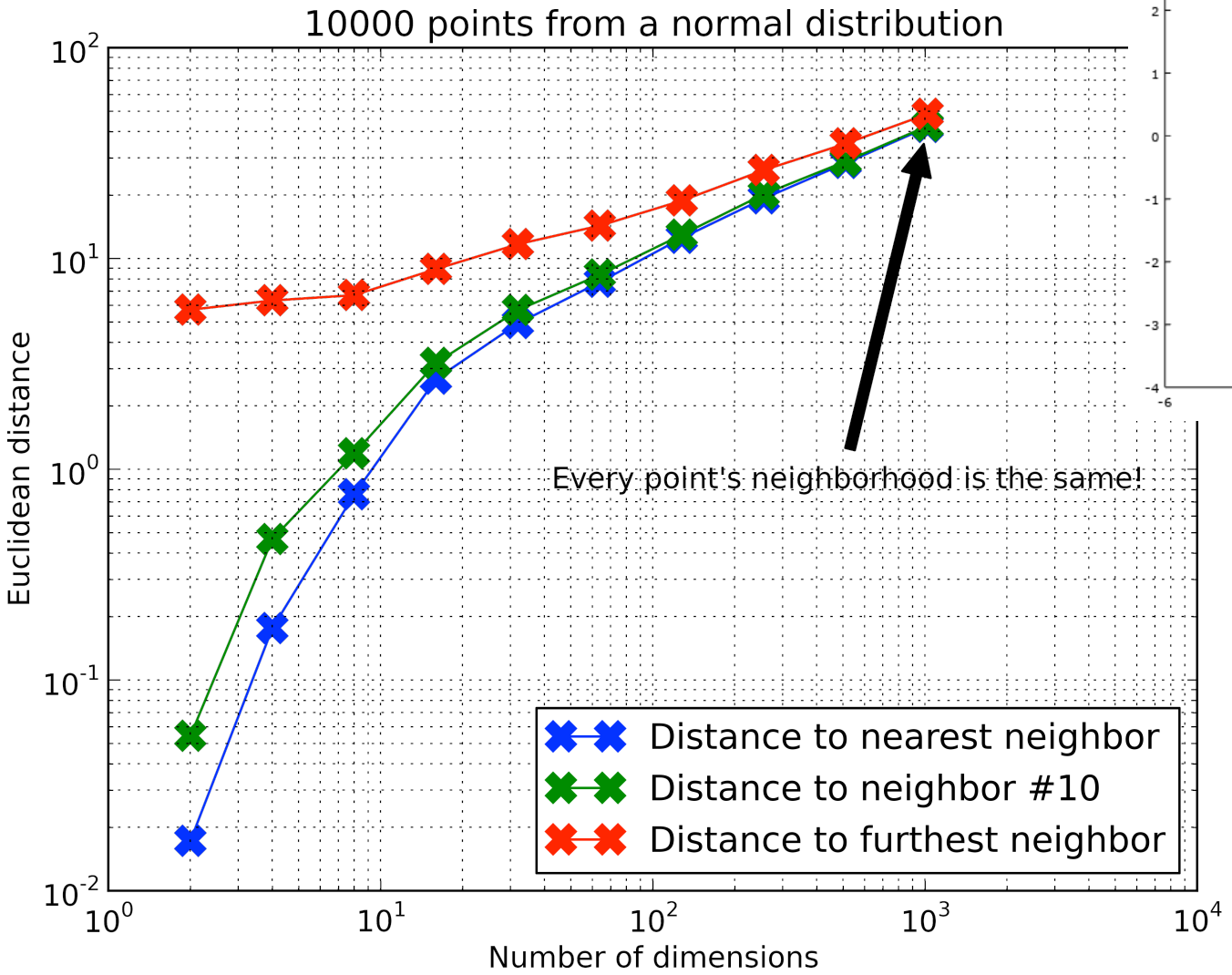
https://arifuzzamanfaisal.com/k-nearest-neighbor-regression/

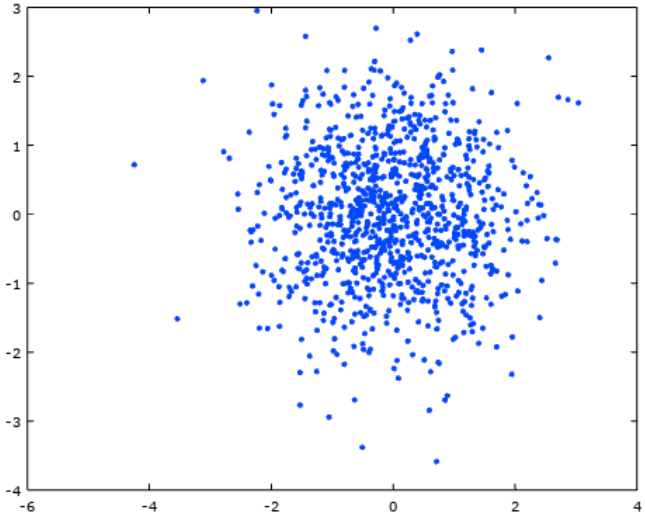
k-Nearest Neighbor Classifier

- Nearest neighbor is susceptible to label noise
- Use the k-nearest neighbors as the classification decision
  - Use majority vote



What's wrong with k-NN in high dimension?





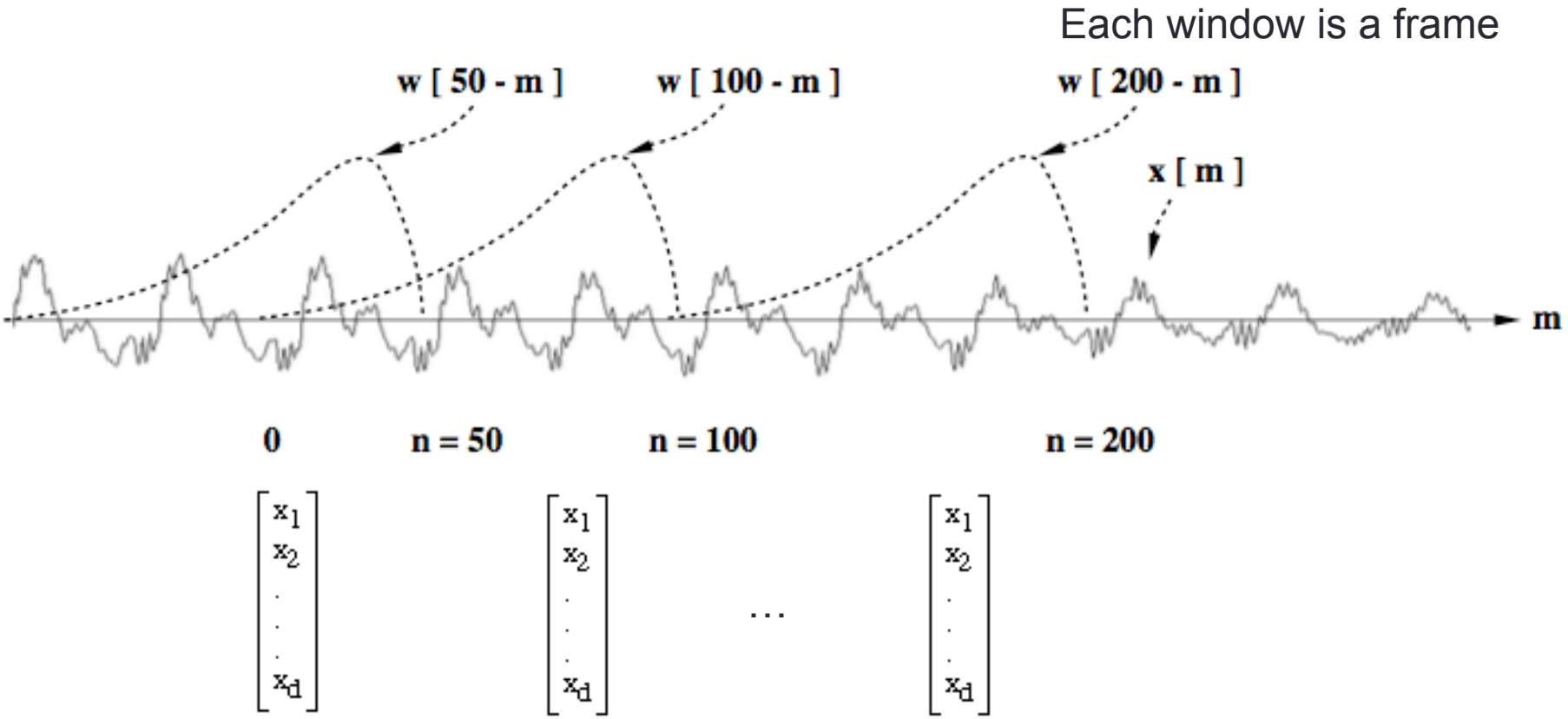
https://erikbern.com/2015/10/20/nearest-neighbors-and-vector-models-epilogue-curse-of-dimensionality.html

Speech Features

- Spectrum
- Filter bank features
- Cepstrum
- MFCC, PLP
- Energy, Pitch, Speaking rate, Lips movement…

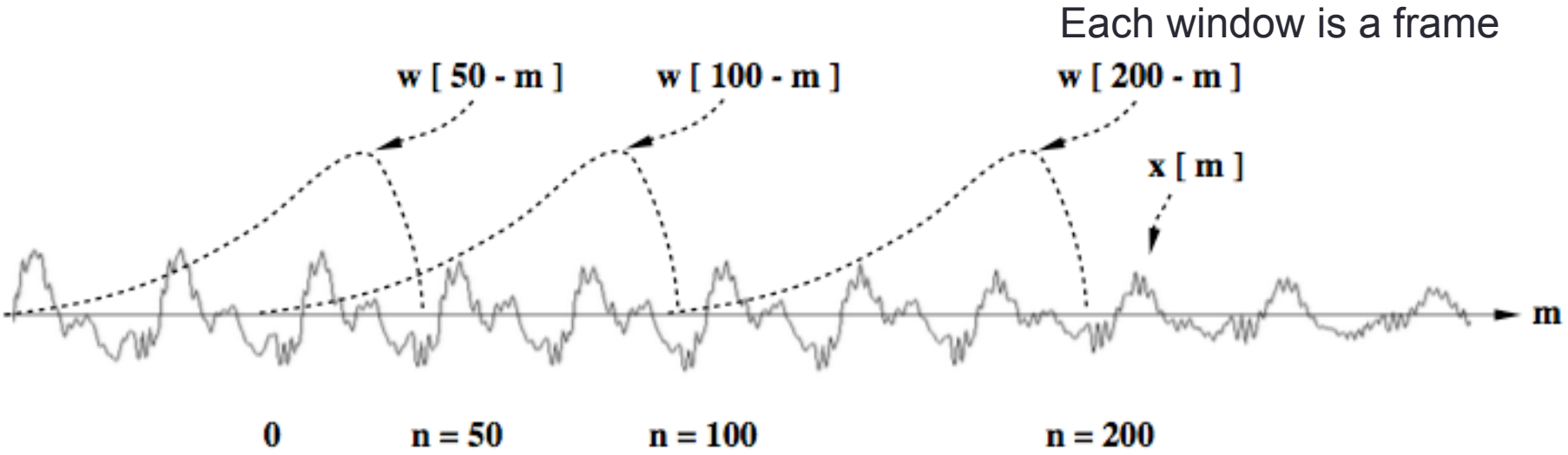
Speech Frame

- An utterance is separated into frames



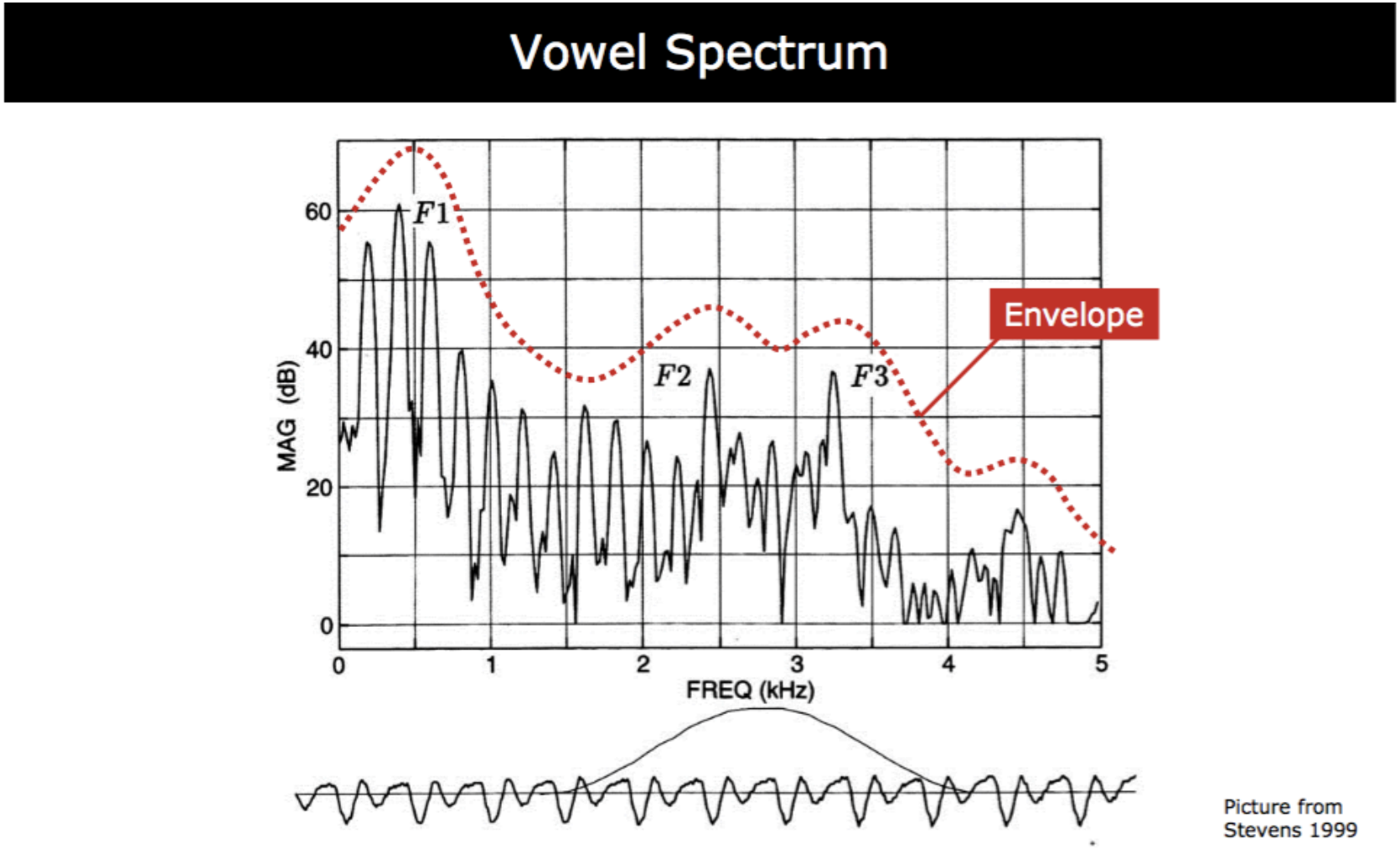
Spectrum

- Short Time Fourier Analysis (STFT)



We usually deal only with the magnitude portion of the spectrum.

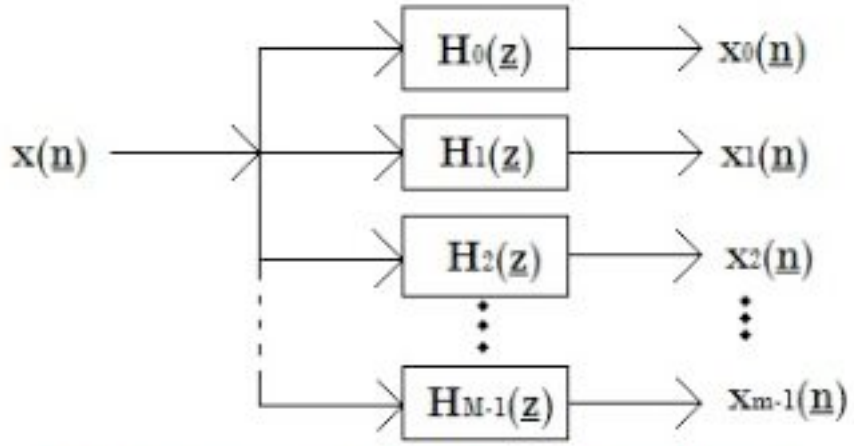
Spectrum

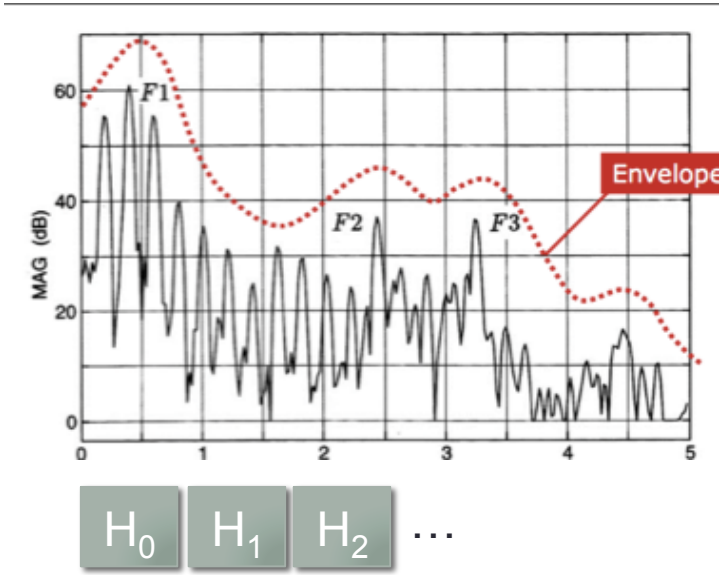


Picture from
Stevens 1999

Filterbank features

- Multiple band pass filters. Each extracts different band of frequencies.
- Can be computed in the time domain, or in the frequency domain (spectrogram)
- Convolution in time = Multiplication in frequency

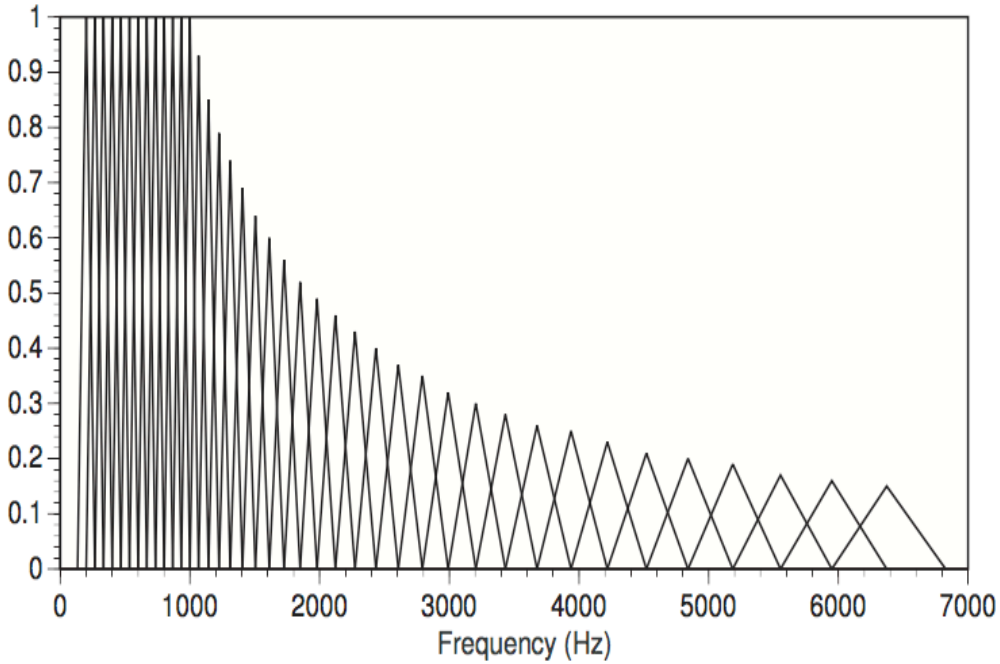




The dimension is reduced but still keep the necessary information to classify speech

Filterbank features

- The frequencies of the filters can be motivated by the human auditory system
- Mel scale, Bark scale – frequency mappings between perceived pitch and actual frequency

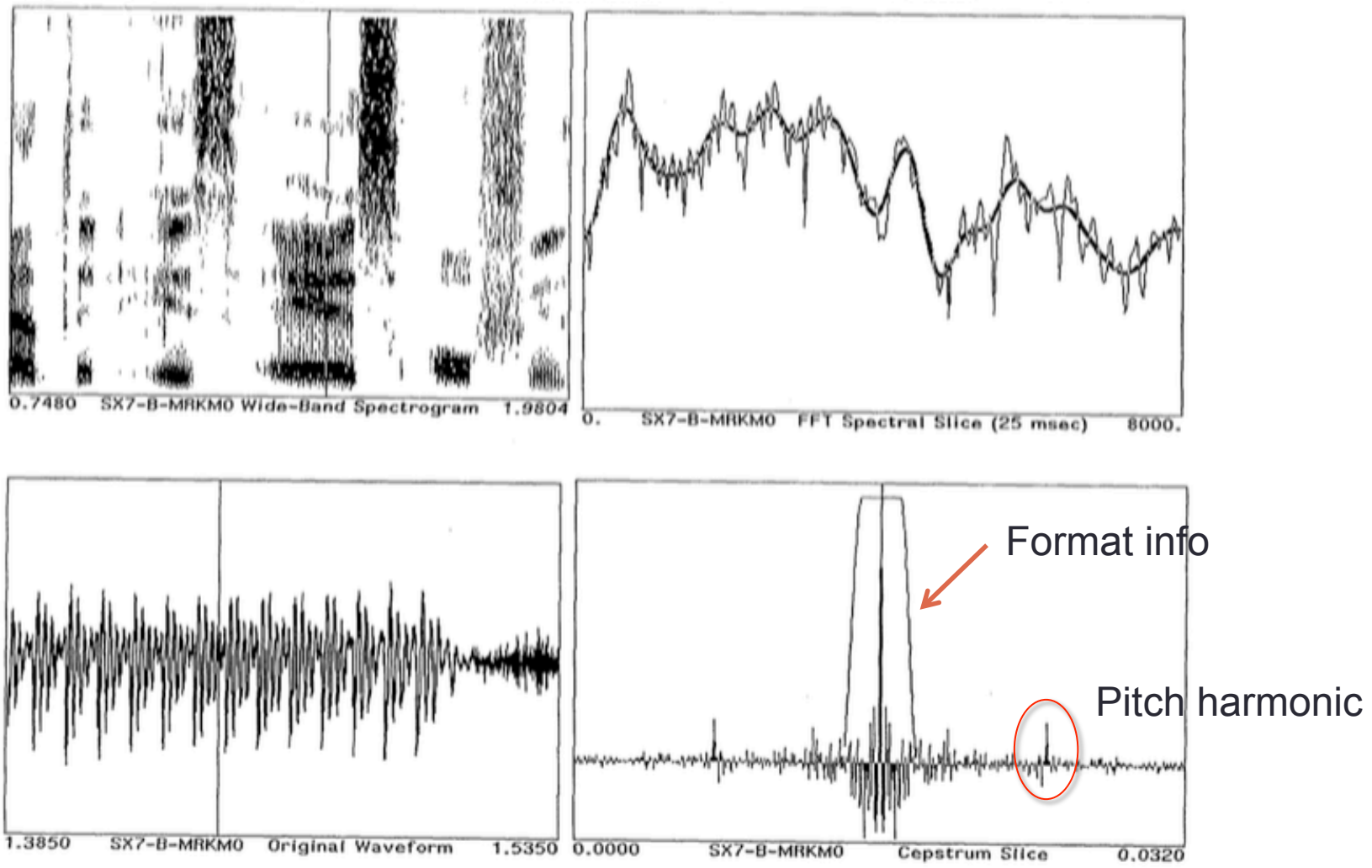


Mel filterbank
Each bank represents the frequencies that activates each neuron in the ear. Center frequencies are linear below 1000 Hz, and logarithmic at higher frequencies

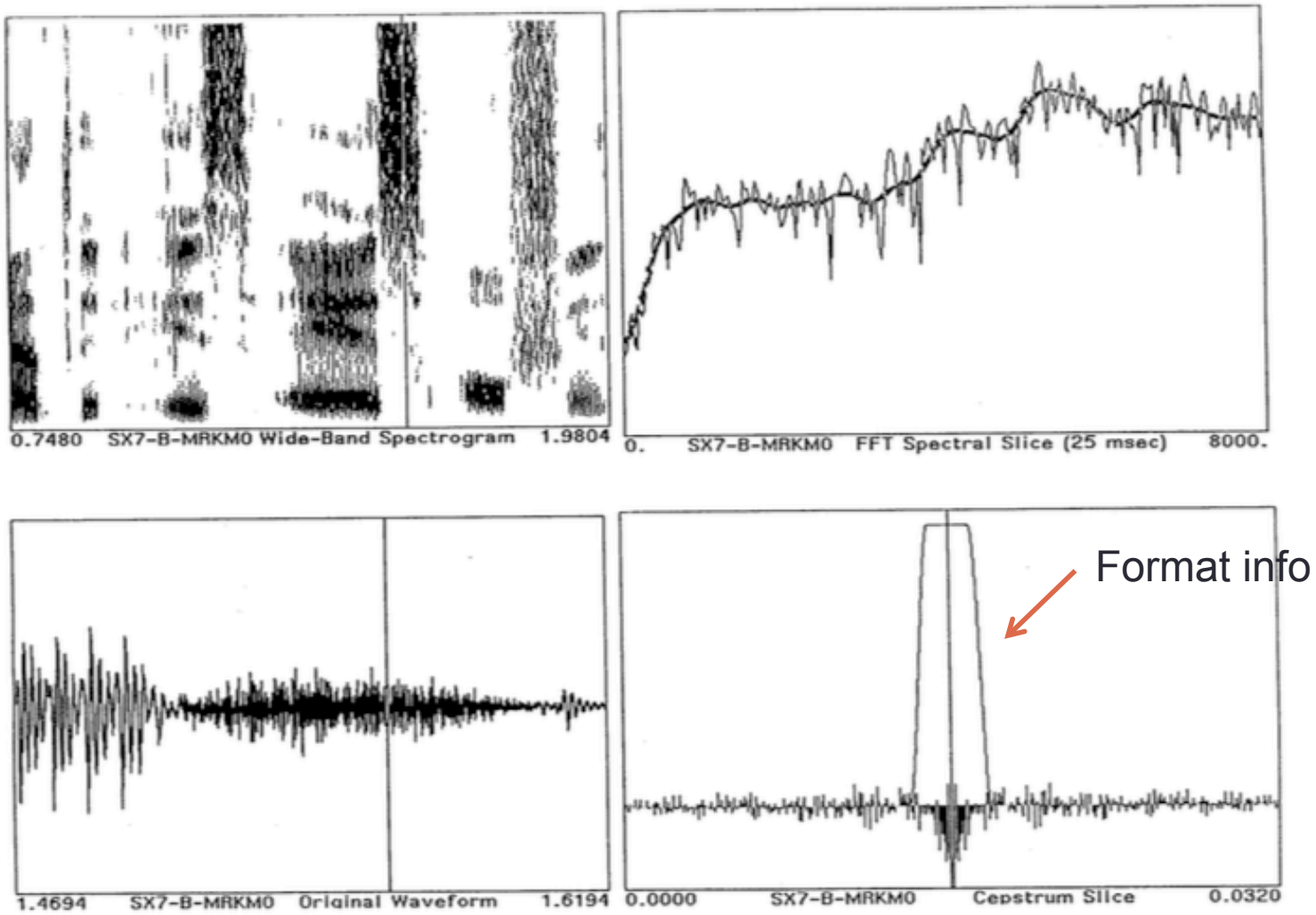
Cepstrum

- Source-filter model
  - We are interested in the filter
- Motivation : Easy to extract additive things than convolutive or multiplication
- s[n] * h[n] ⇔ S(e^jw) H(e^jw)
- Taking the log : log(S(e^jw)) + log(H(e^jw))
- If s[n] and h[n] are real valued, it can be showed that
- s[n] * h[n] ⇔ s'[n] + h'[n]
- where $s'[n] = F^{-1}\{ \log(|F\{s[n]\}|) \}$
- s'[n] is the Cepstrum

Cepstrum of a vowel

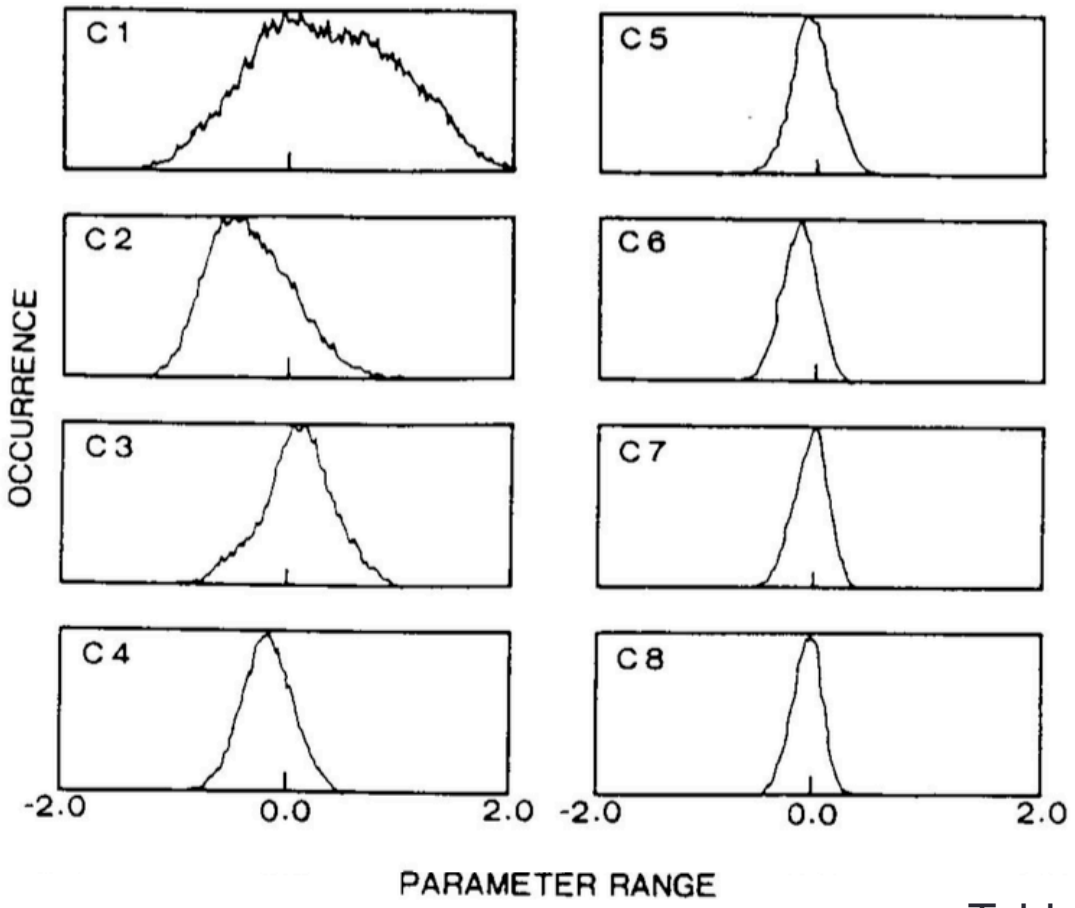


Cepstrum of unvoiced fricative



Statistical property of Cepstral coefficients

From a digit database (100 speakers) over dial-up telephone lines.

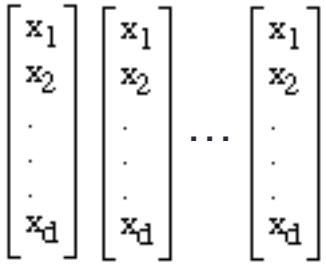


Tohkura, 1987

Cepstrum notes

- Anagrams
  - Spectrum - Cepstrum
  - Filtering – Liftering
  - Frequency - Quefrency
- Cepstrum can be considered as “Spectrum of Spectrum” – extract frequency information of the spectrum (view as a time signal)
- In ASR, we use Discrete Cosine Transform (DCT) instead of Inverse Fourier Transform.
  - DCT gives real value (saves computation)
  - For natural occurring signals, DCT tends to compact more energy in the lower frequencies (less dimensions required)

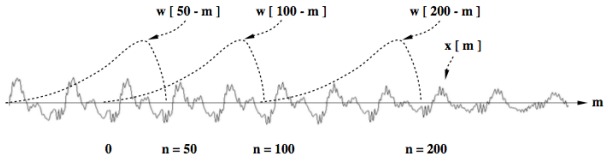
MFCC (Mel Frequency Cepstral Coefficient)



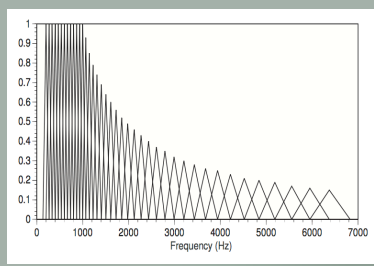
Select lower coefficients

Goal: reduce dimension while keeping all relevant information

DCT



|STFT|

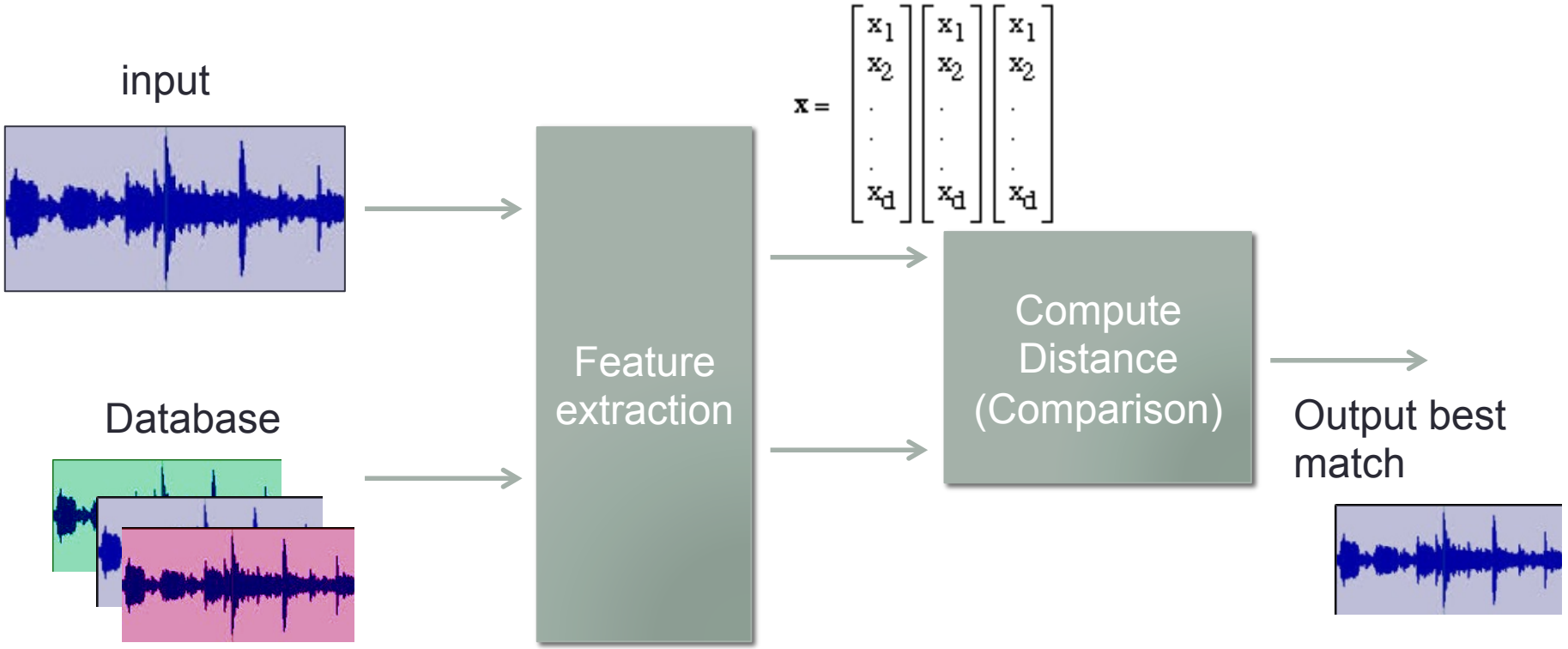


log

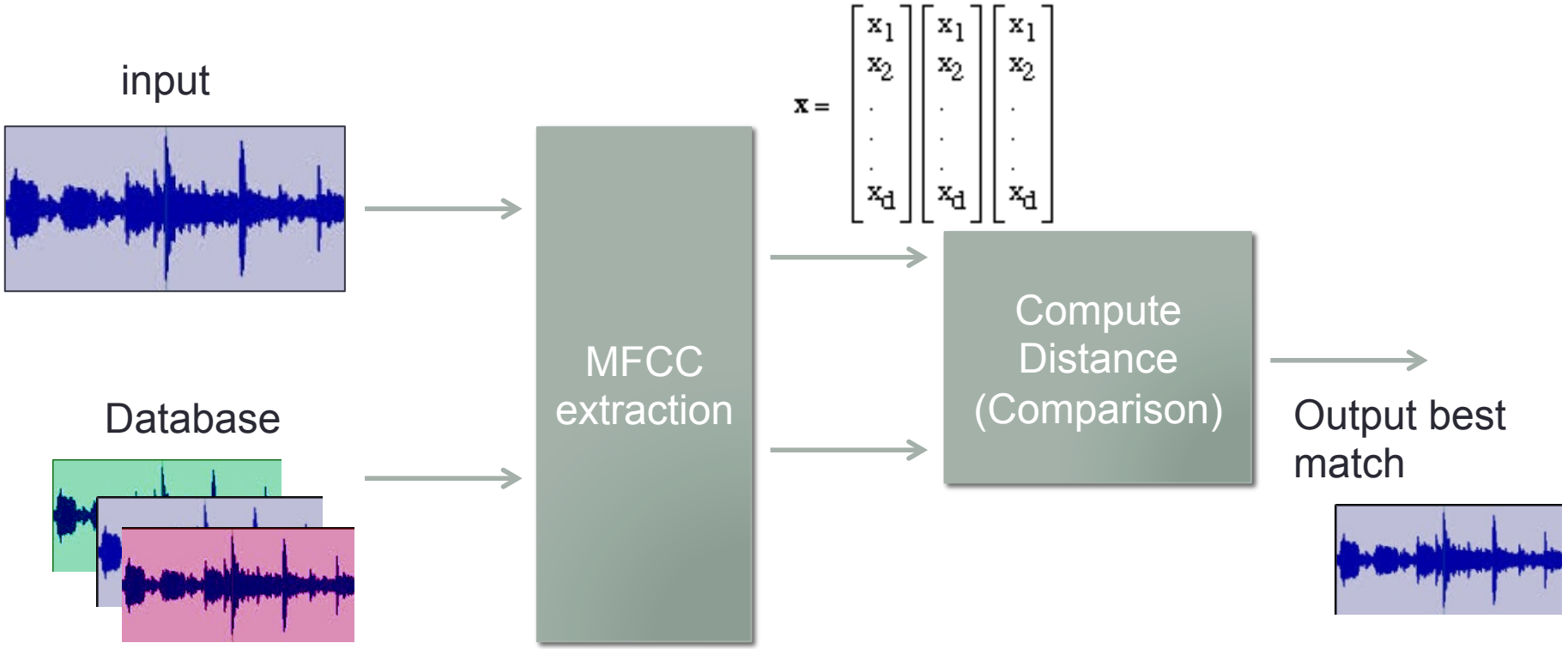
Is MFCC invariance to

- Speaking rate?
- Pitch?
- Gender?
- Noise?
- Loudness?
- Microphone?

A classification framework

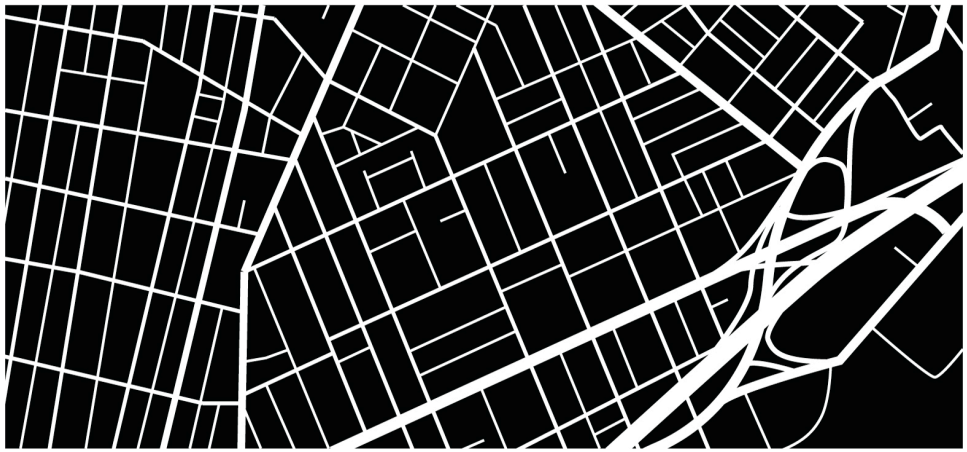


A classification framework



Distance function

- Property of distance function
  - d(x,y) >= 0 non-negativity
  - d(x,y) = 0 ⇔ x = y identity
  - d(x,y) = d(y,x) symmetry
  - d(x,z) <= d(x,y) + d(y,z) triangle inequality
- h(x,y) represents shortest path to drive a car from x to y
  - Does h(x,y) satisfy
    - Non-negativity?
    - Identity?
    - Symmetry?
    - Triangle inequality?

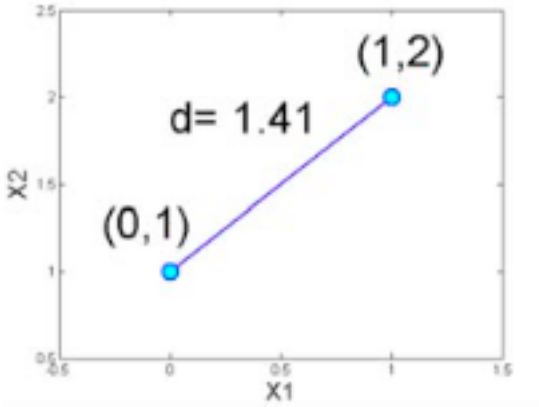


Euclidean Distance

- $X = (x_1, x_2, x_3, \ldots, x_n)$, $Y = (y_1, y_2, y_3, \ldots, y_n)$

$$d(X, Y) = \sqrt{\sum_i (x_i - y_i)^2}$$

Good for when the data are equally spread in each dimensions (variance in each dimension is the same)

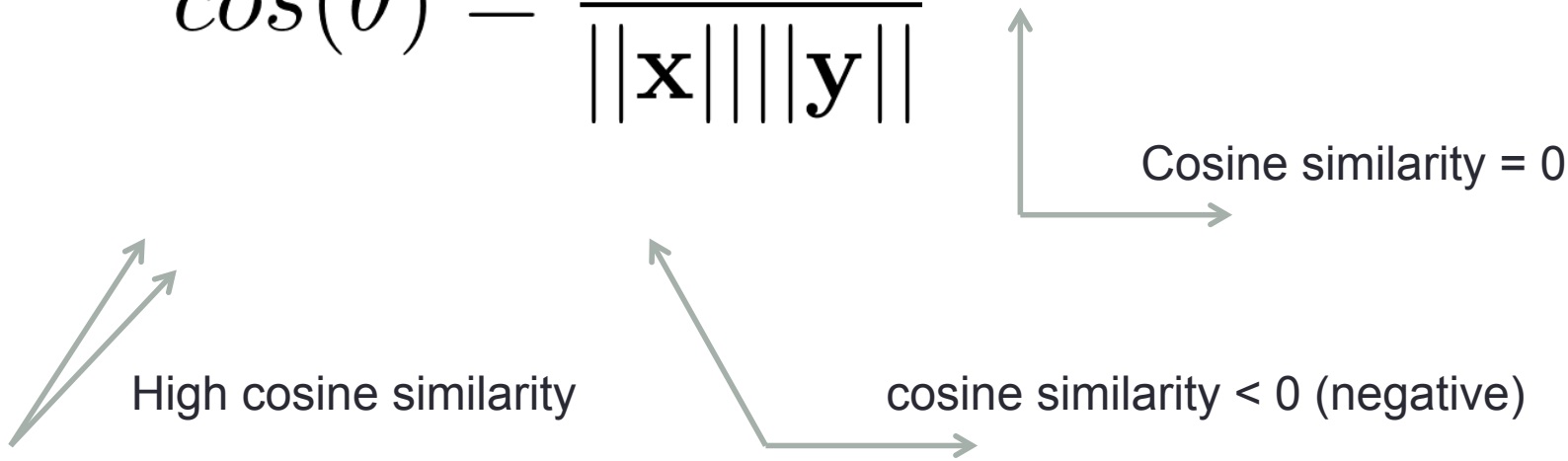


Cosine similarity

- X = (x_1,x_2,x_3,…,x_n), Y = (y_1,y_2,y_3,…,y_n)
- Angle between two vectors

$$\mathbf{x} \cdot \mathbf{y} = ||\mathbf{x}||||\mathbf{y}||cos(\theta)$$

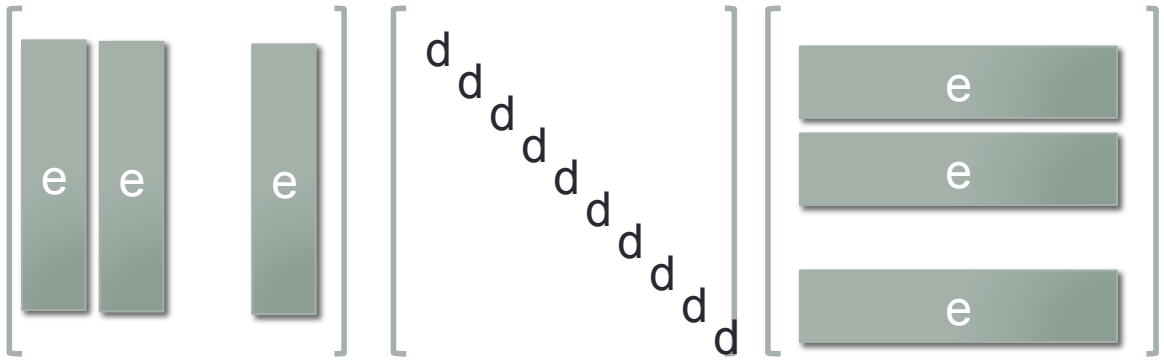
$$cos(\theta) = \frac{\mathbf{x} \cdot \mathbf{y}}{||\mathbf{x}||||\mathbf{y}||}$$



Whitening (PCA)

- Principal Component Analysis
- Find the project along the dimensions that has the highest variance in the data
- Let $\sum$ be the covariance matrix. E is the matrix of eigenvectors, and D has eigenvalues along the diagonal. With eigenvalue decomposition:

$$\Sigma = EDE'$$



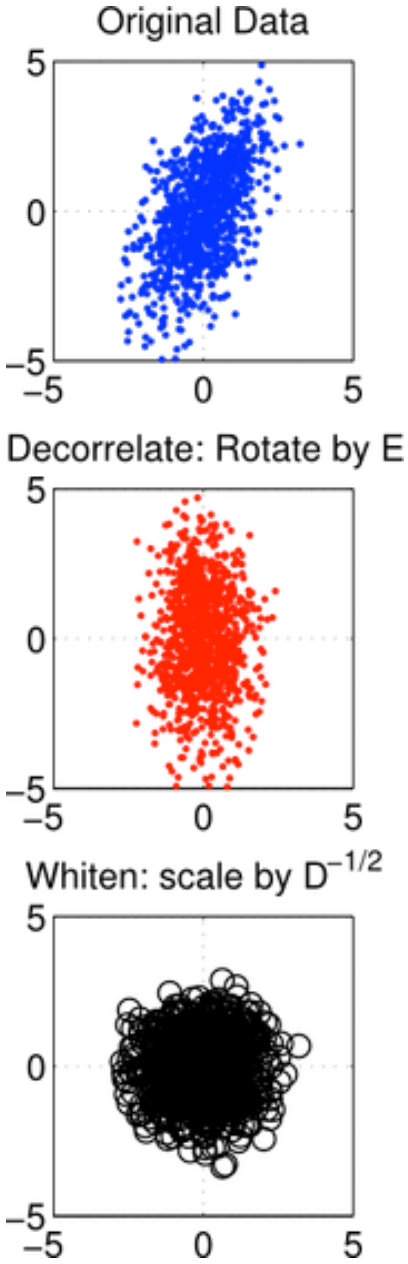
Whitening (PCA)

- Whitening decorrelates and scale

$$Y = D^{-1/2}E'X$$

Where X is the original data [d x n], d is the feature dimension, n is the number of input features

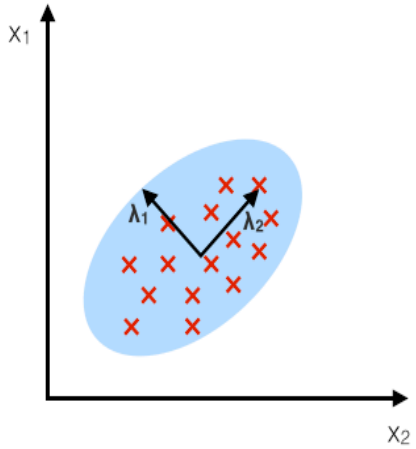
- Can be used to reduce dimension by keeping only the eigenvectors associated with the highest eigenvalues



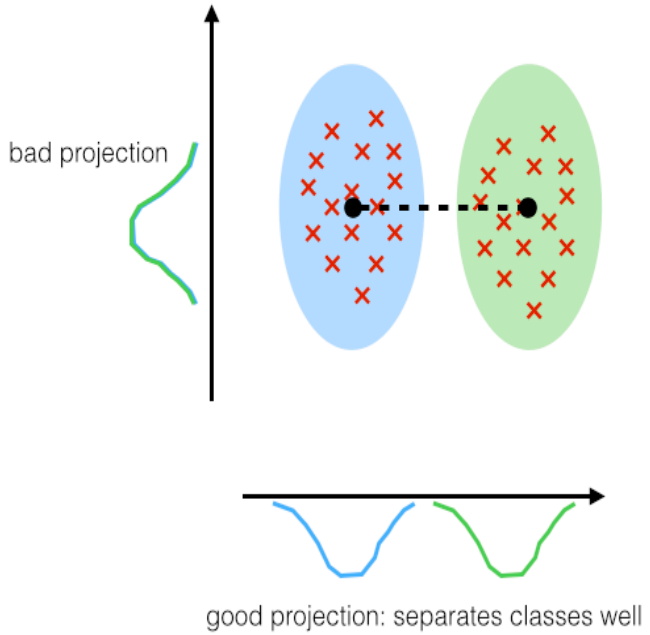
https://theclevermachine.wordpress.com/2013/03/30/the-statistical-whitening-transform/

LDA (Linear Discriminant Analysis)

PCA:
component axes that maximize the variance



LDA:
maximizing the component axes for class-separation



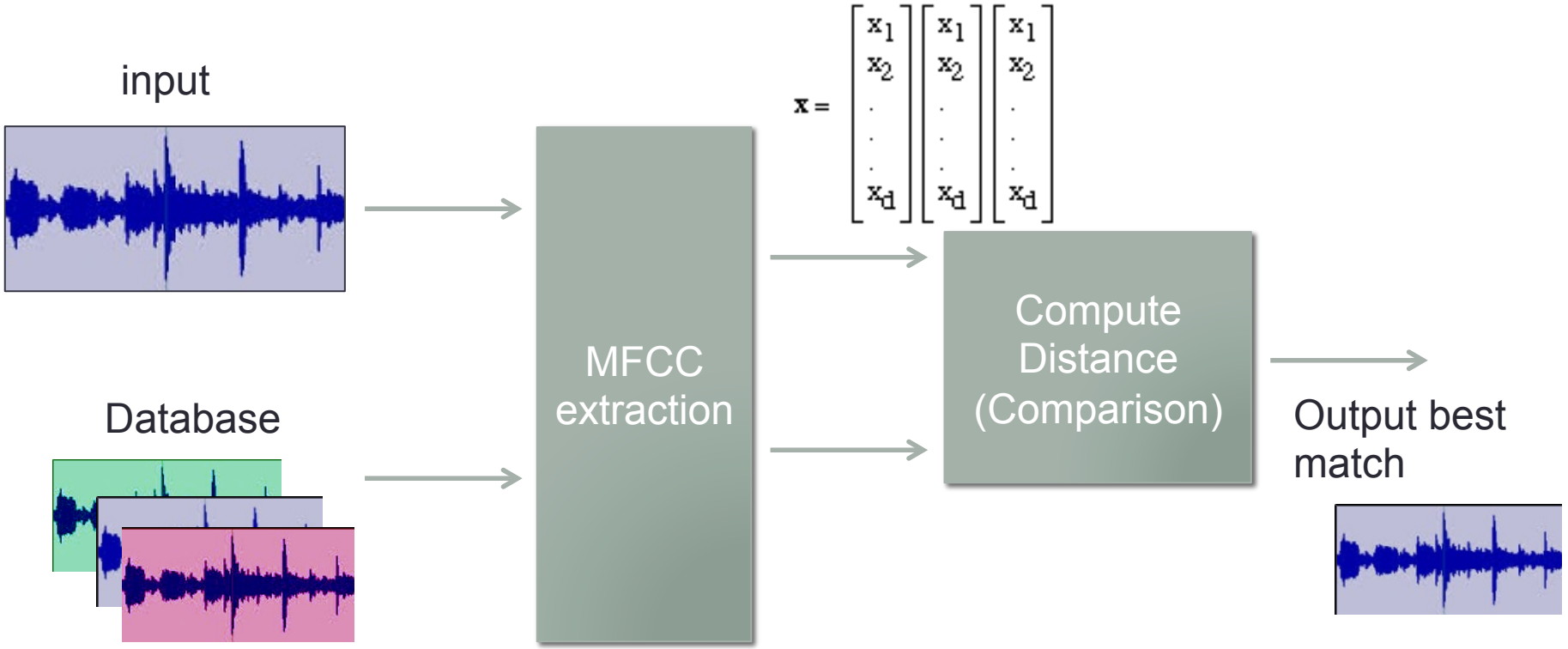
http://sebastianraschka.com/Articles/2014_python_lda.html

LDA vs PCA

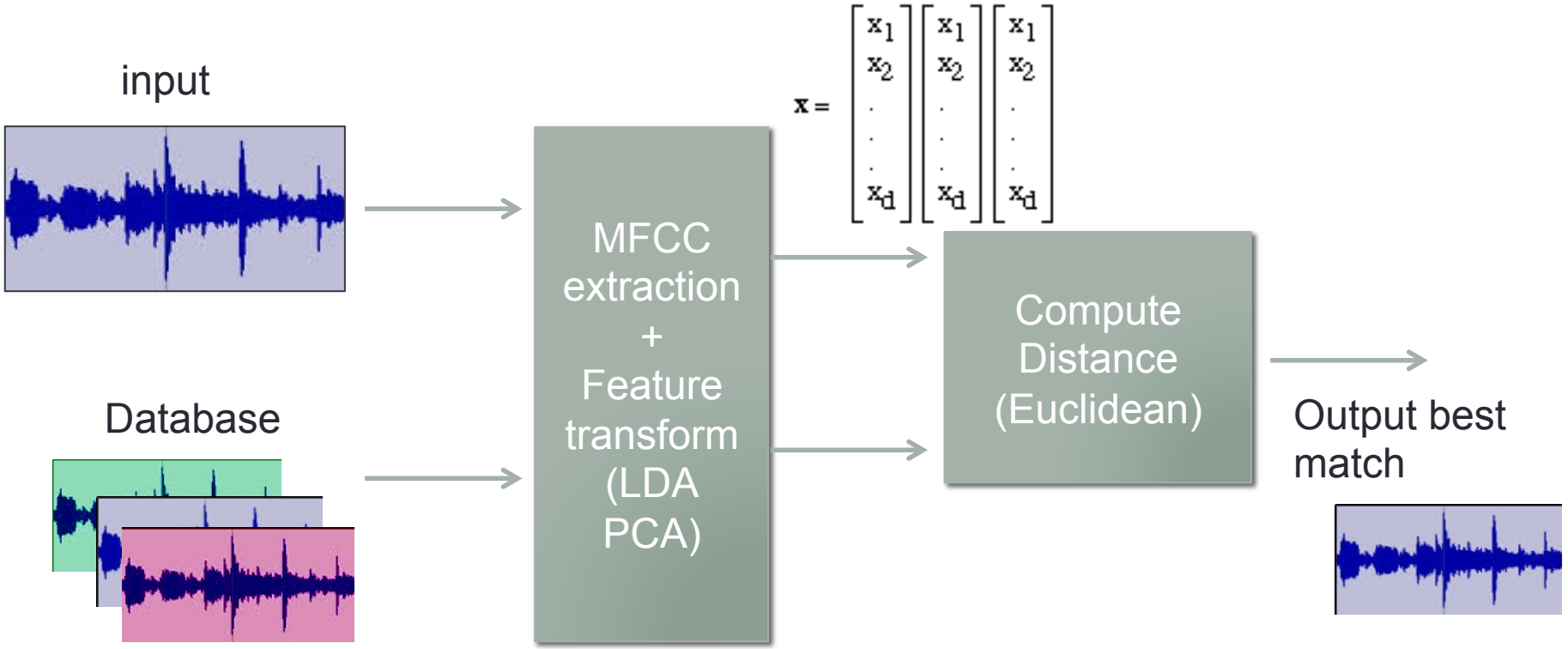
- PCA
  - Just need data without labels
  - Good for whitening and dimensionality reduction
  - Good for data visualization
- LDA
  - Needs class labels
  - Good for classification

Machine learning applications usually use both. First use PCA to reduce feature dimension. Then LDA to increase discrimination between classes

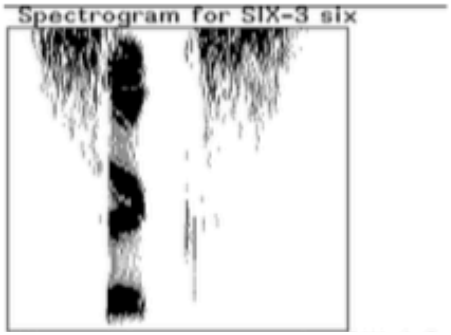
A classification framework

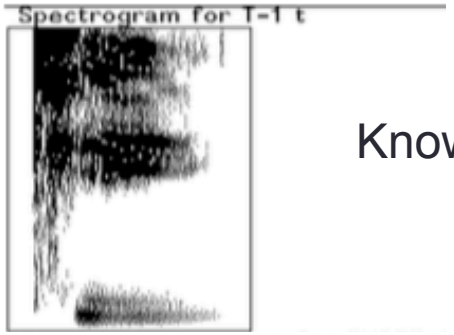


A classification framework

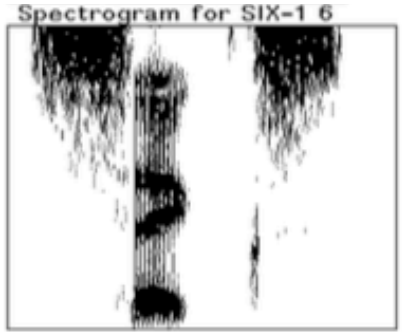


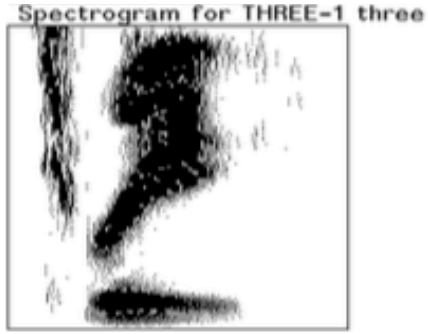
Mismatch in length





Known templates

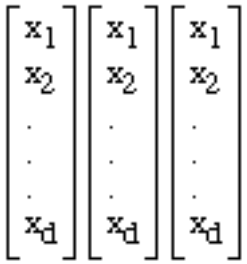


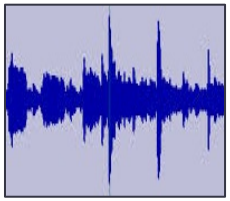


Test sample

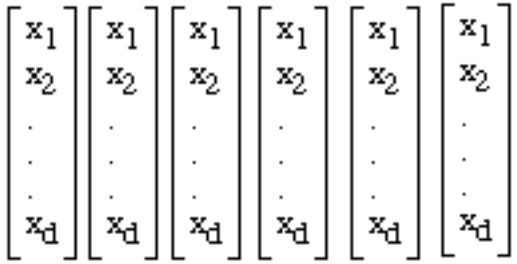
Speaking speed

- How do you compare two utterances with different length?





Template

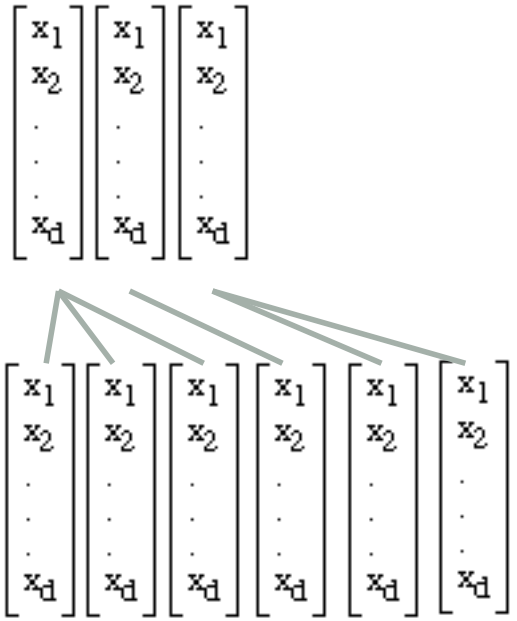


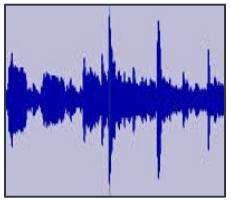
Test

Find a way to align the two sequences

Alignment

- How do you compare two utterances with different length?





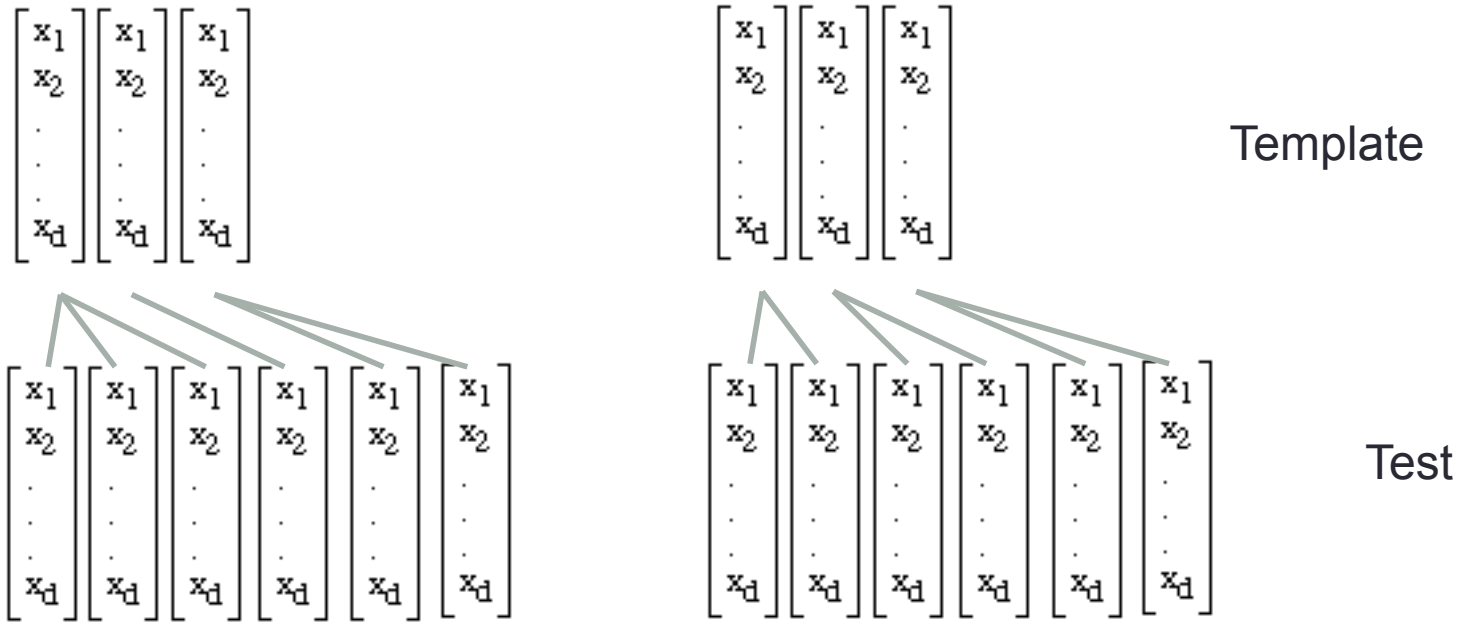
Template

Test

Find a way to align the two sequences

Alignment

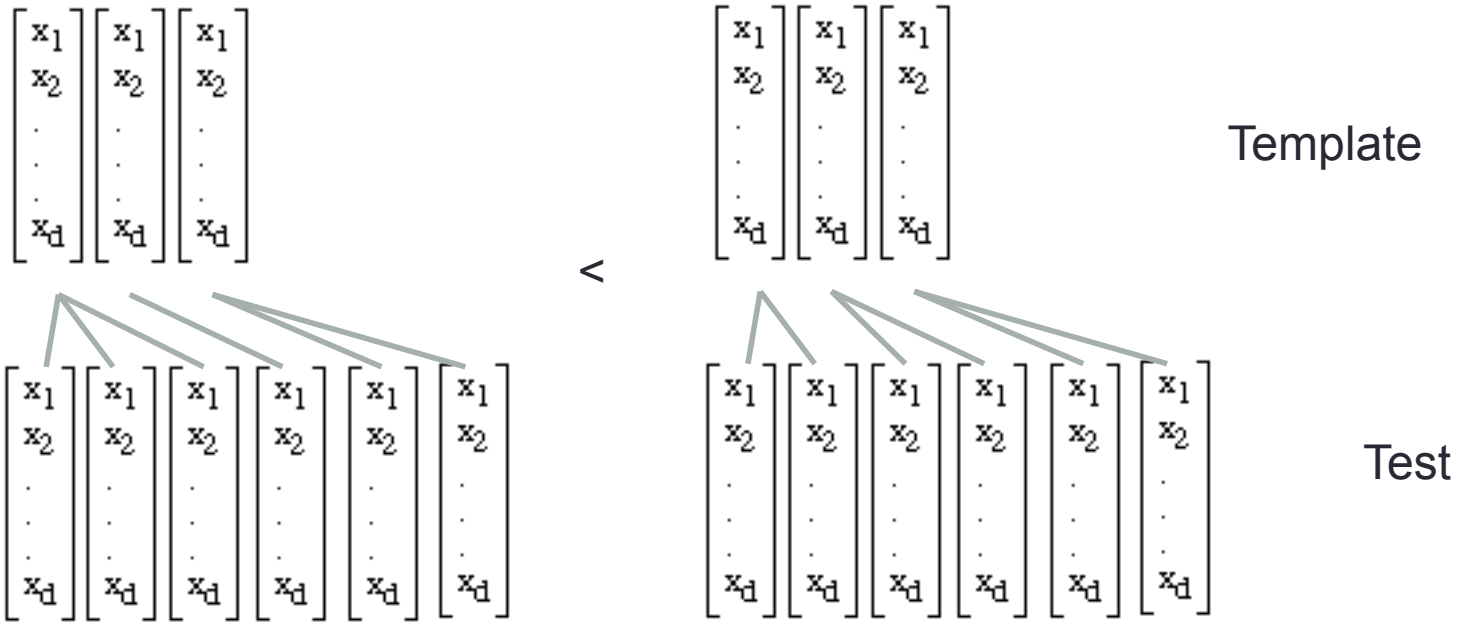
- How do you compare two utterances with different length?



Which one is better?
The one that have smaller “total distance”
d1 + d2 + d3 + d4 + d5 + d6

Alignment

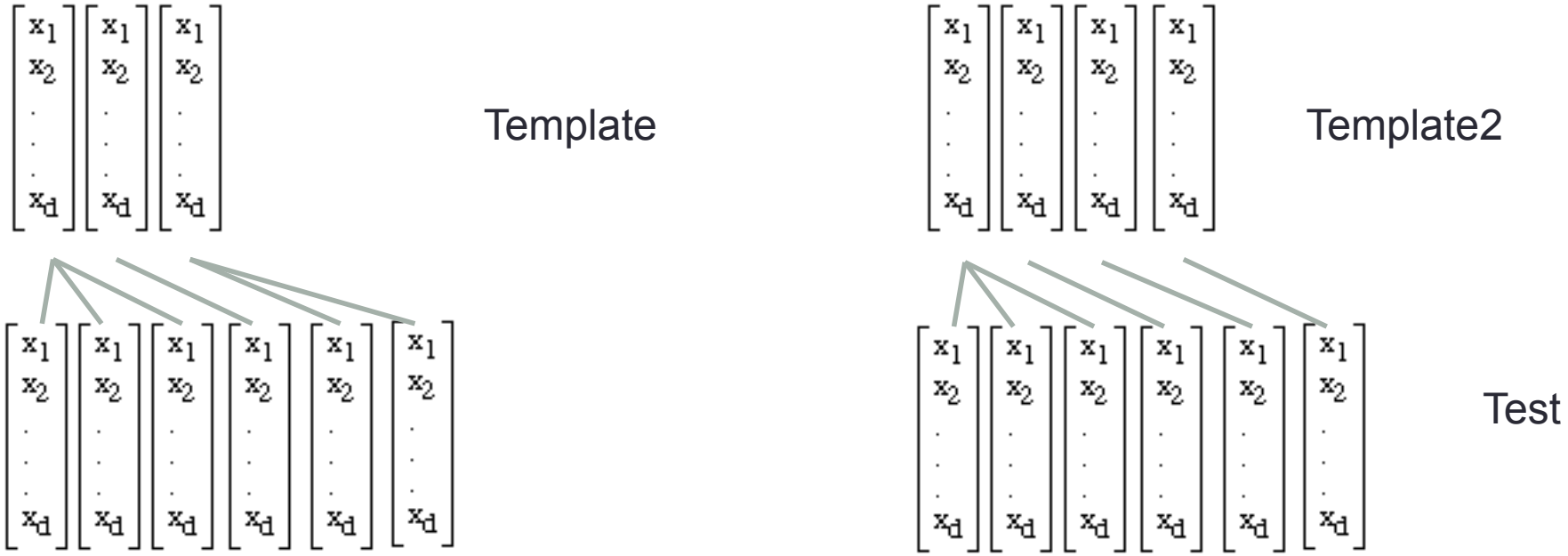
- How do you compare two utterances with different length?



Which one is better?
The one that have smaller “total distance”
d1 + d2 + d3 + d4 + d5 + d6

Alignment

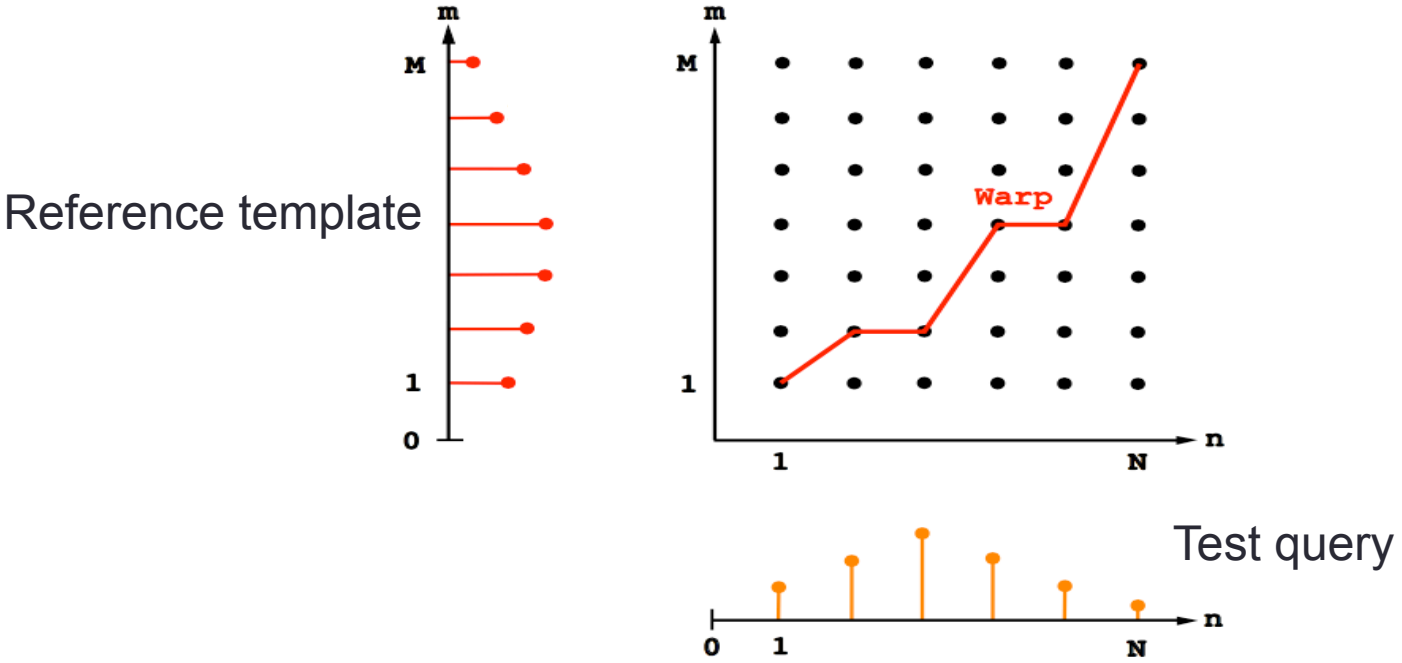
- How do you compare two utterances with different length?



Assign label using the closest template

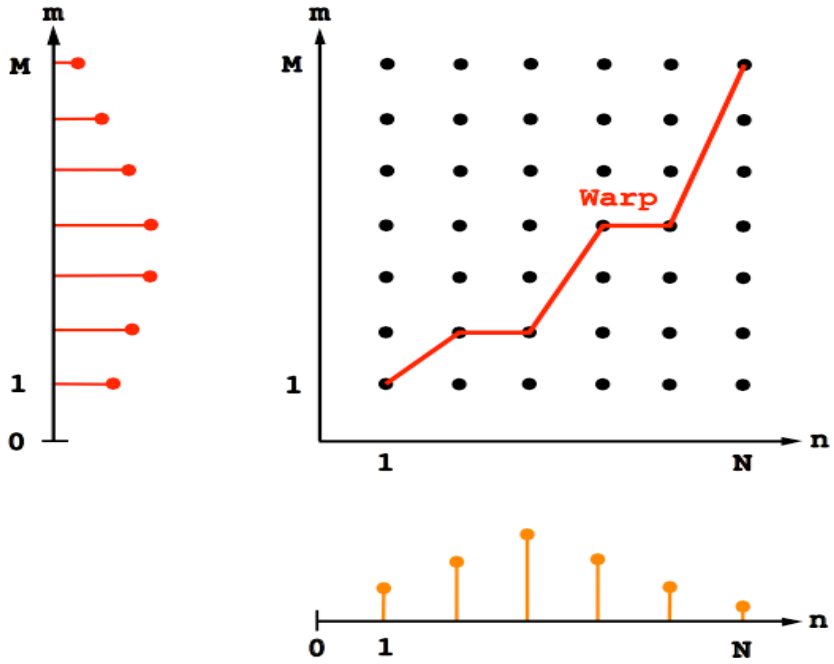
Dynamic Time Warping

- Find the best warping path to align the two inputs
- Can be represented as a 2-D plot of size M,N
- If the warp goes through point (x,y), the frames x and y are aligned with each other



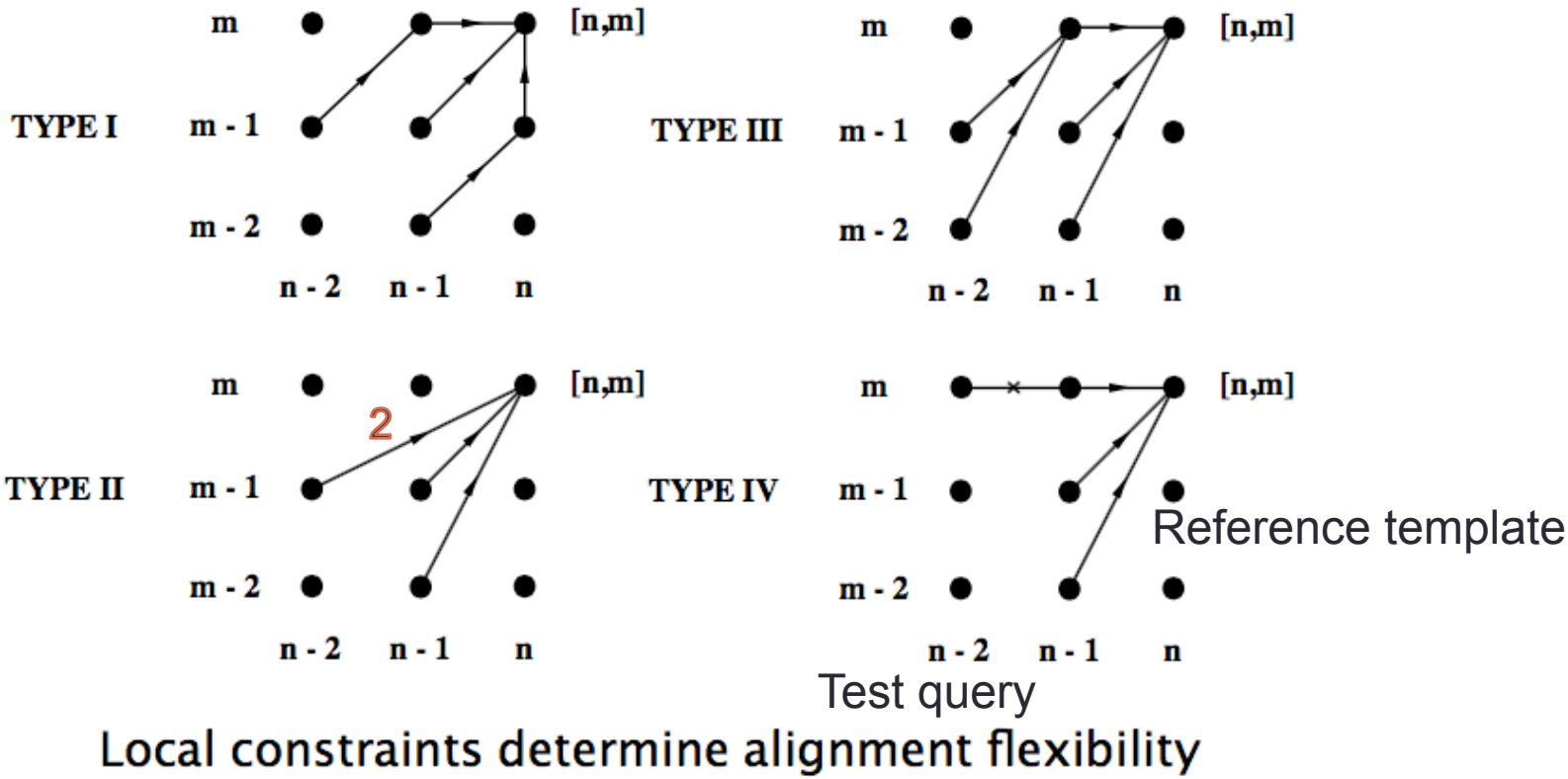
Warp constraint

- End points : start at (1,1) ends at (N,M)
- Monotonic : non-decreasing (cannot go back)
- This reduce possible search paths and makes for efficient algorithms



Local constraint

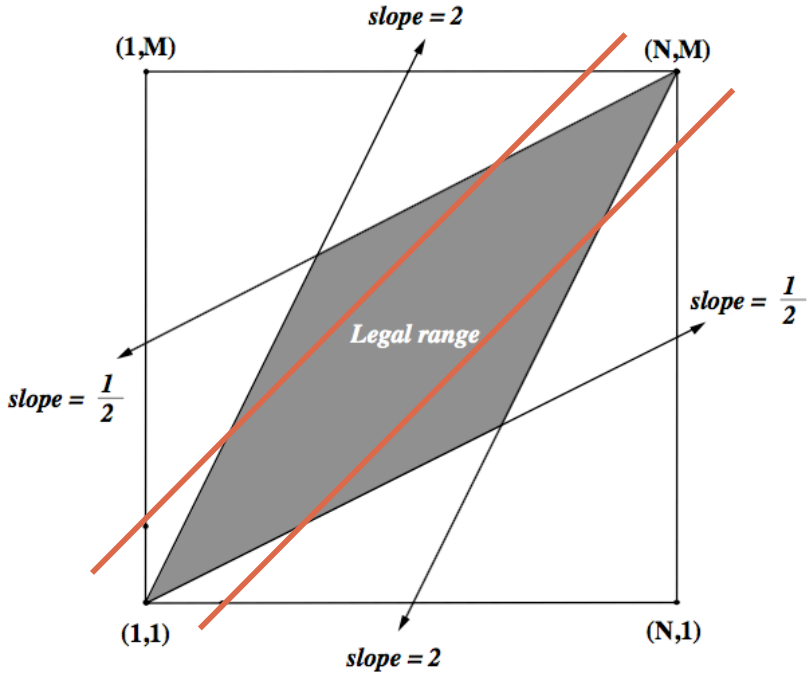
- Not all moves should be valid



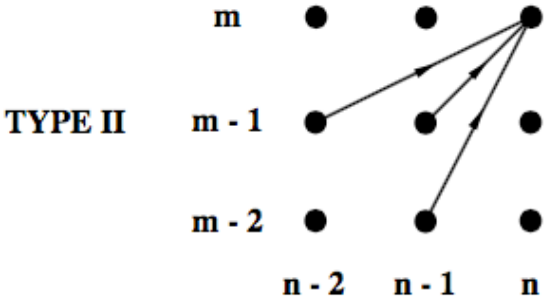
Local constraints determine alignment flexibility

Global constraint

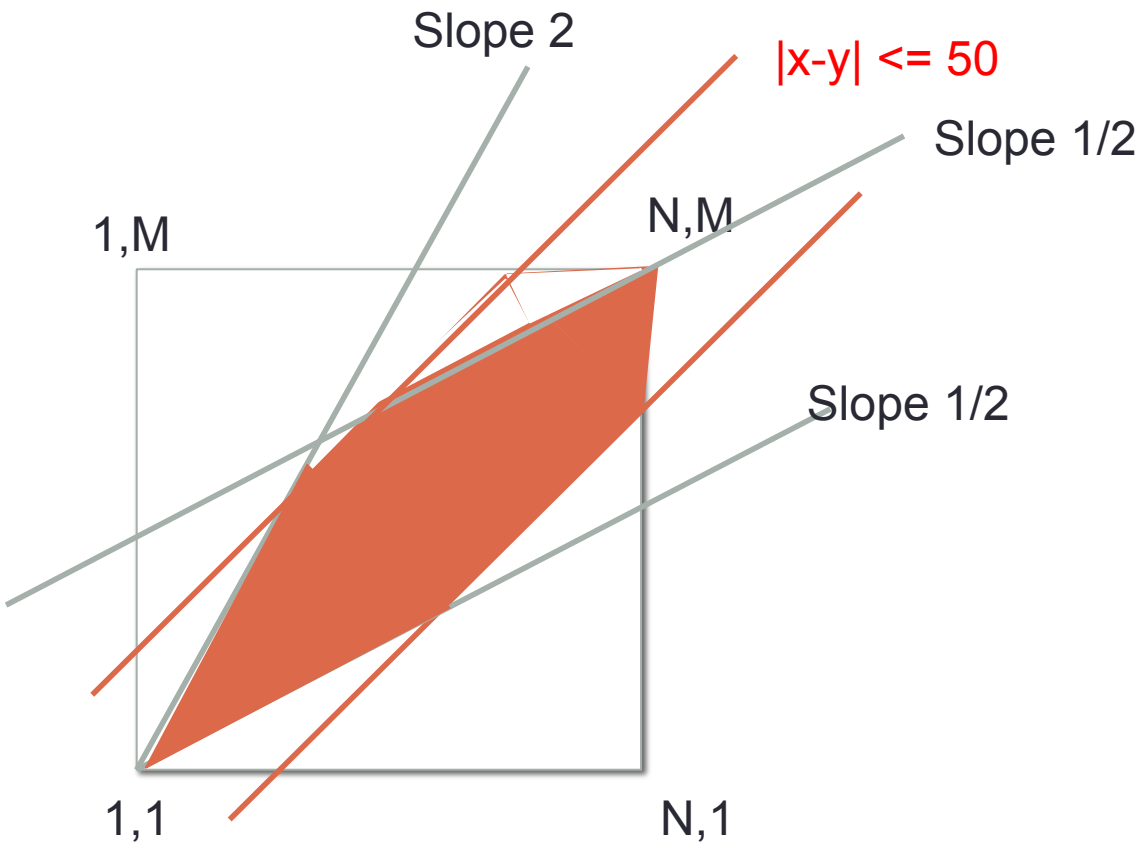
- Derived from local constraints
- Can also be additional constraints
  - | x –y | <= 4



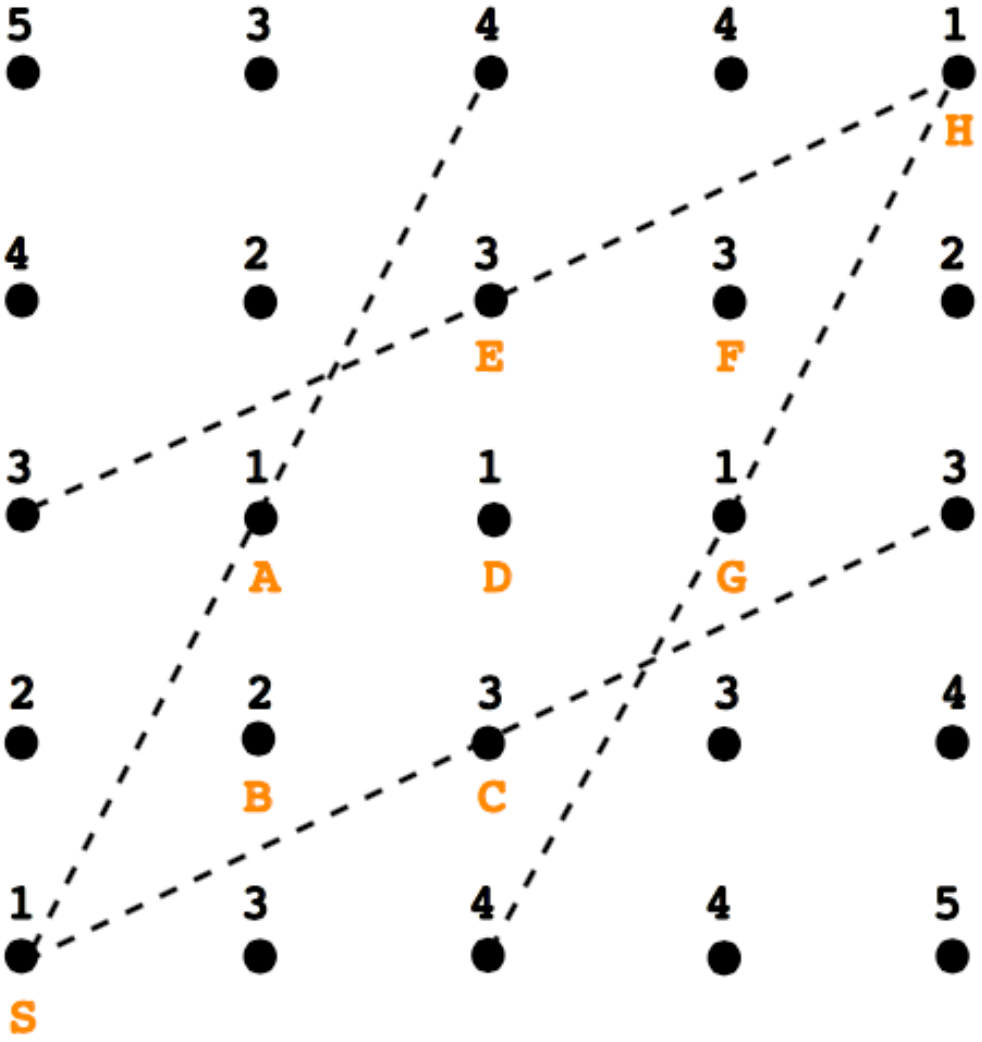
Local constraints exclude portions of search space



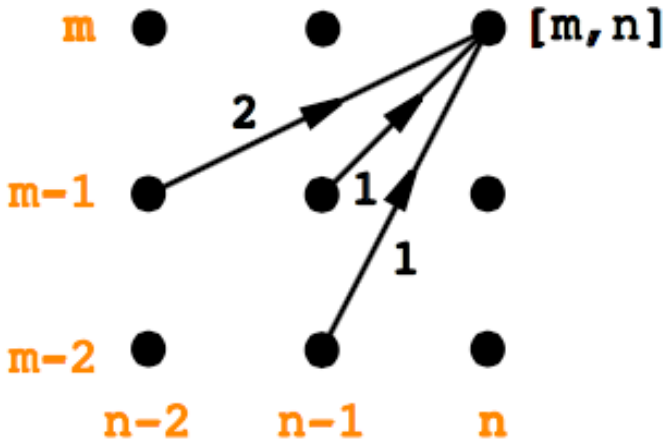
Global constraint



Search



Breath first, depth first search?

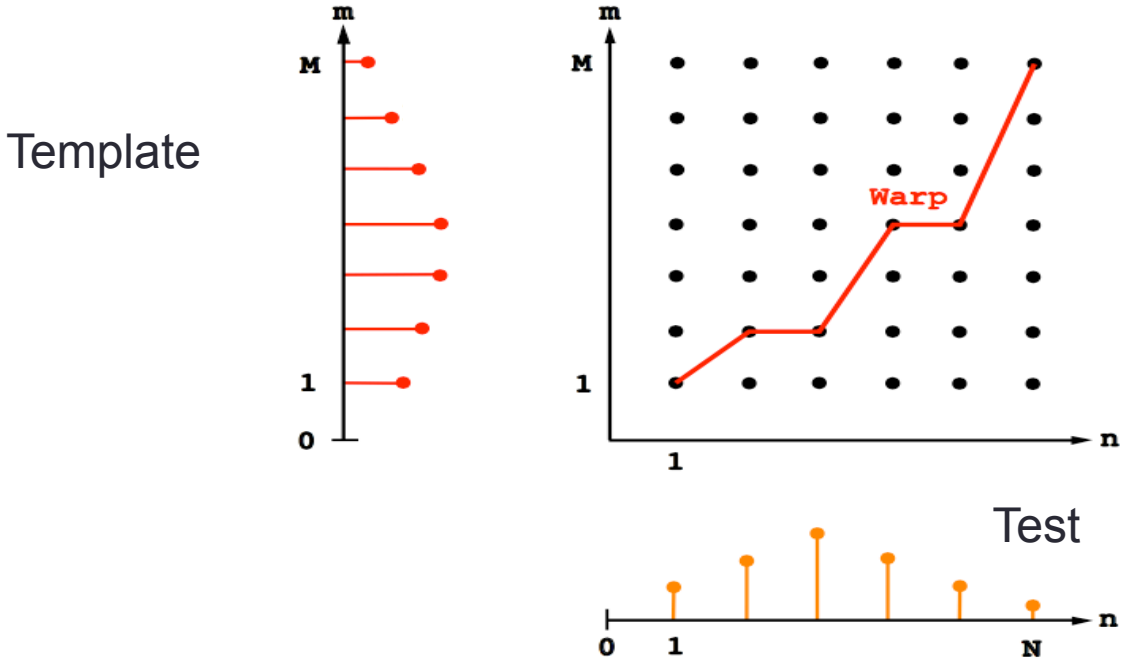


Expensive

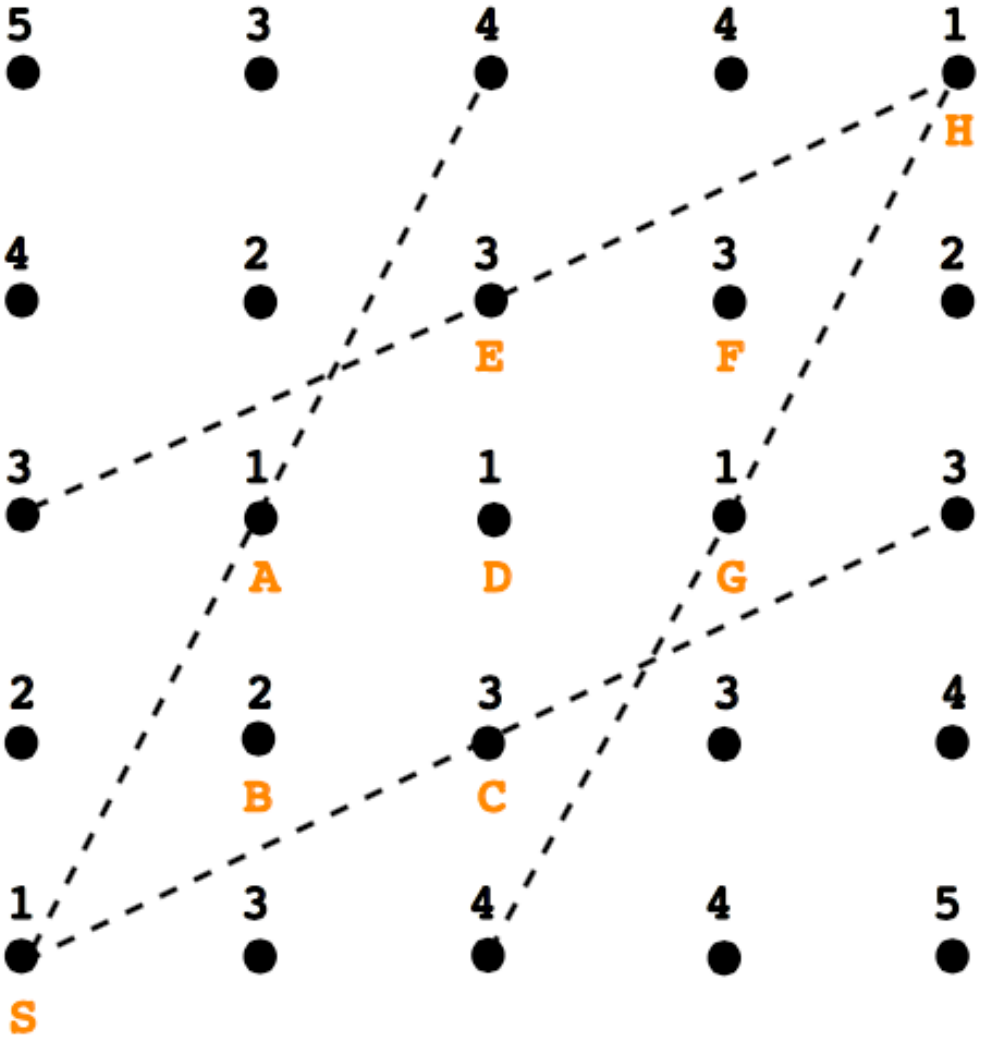
Dynamic Programing (DP)

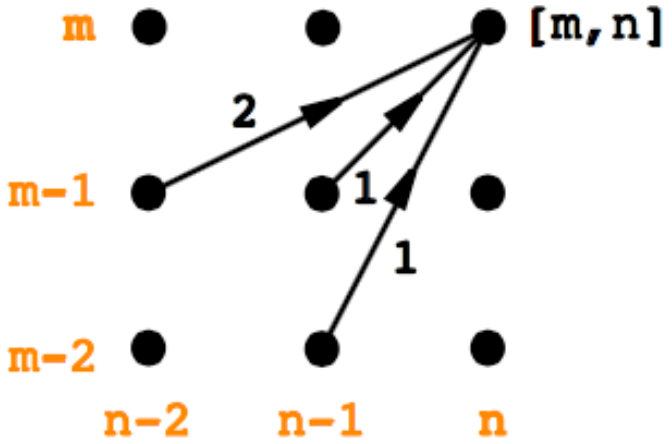
- We can solve DTW with dynamic programing
- Let d(x,y) be the shortest distance to point (x,y)
  - $d(x,y) = \min_{x',y'}(d(x',y') + \text{distance from } (x,y) \text{ to } (x',y'))$

where (x',y') are points that has a valid path to (x,y)

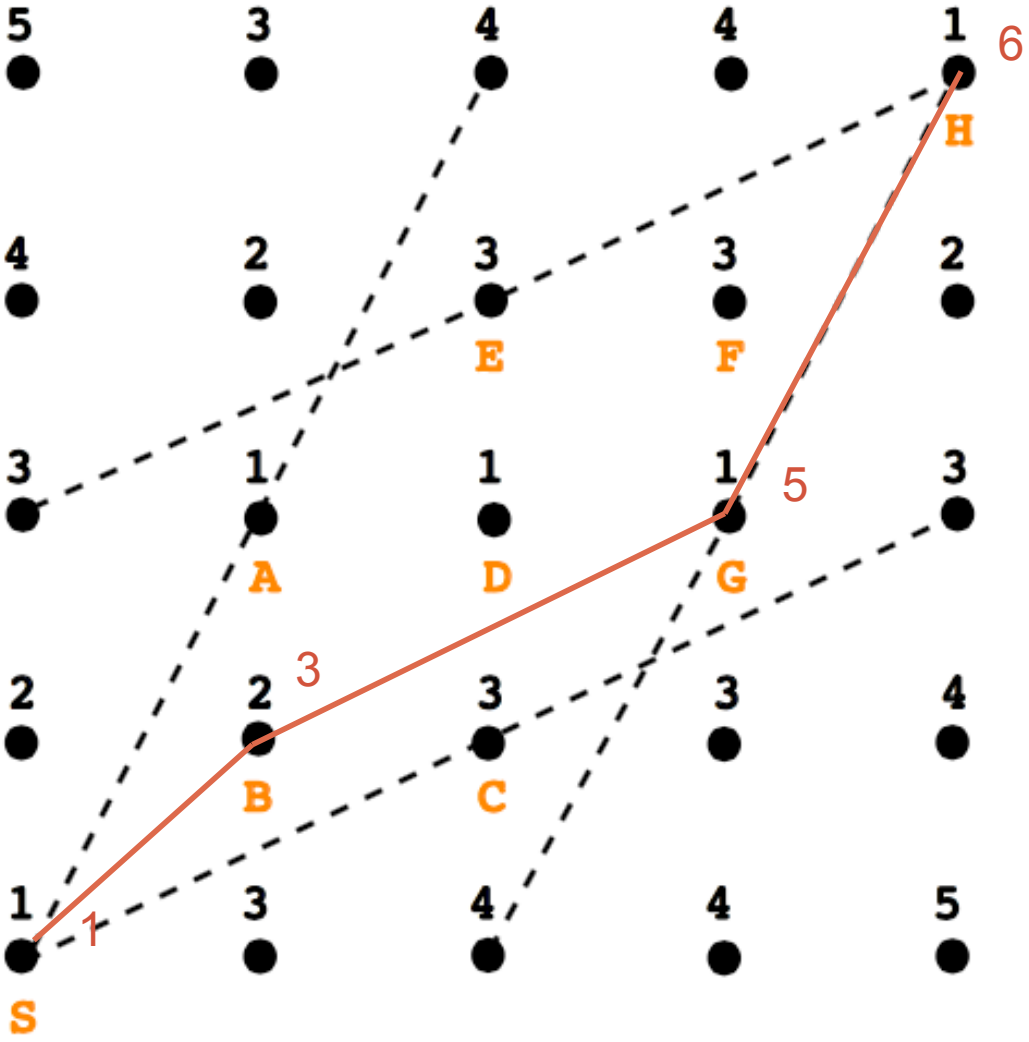


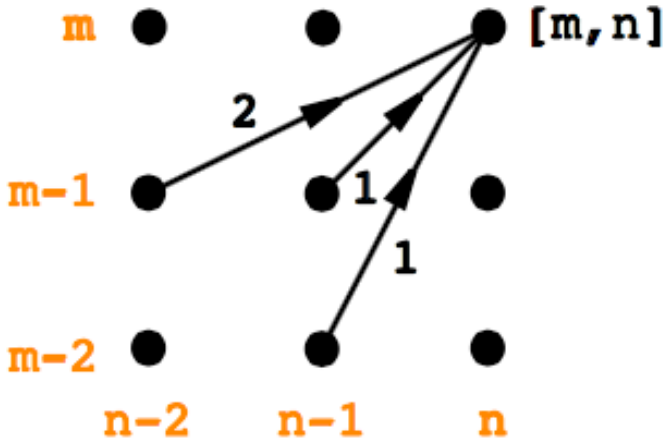
DTW example



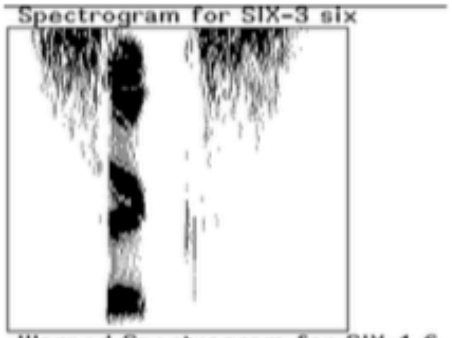


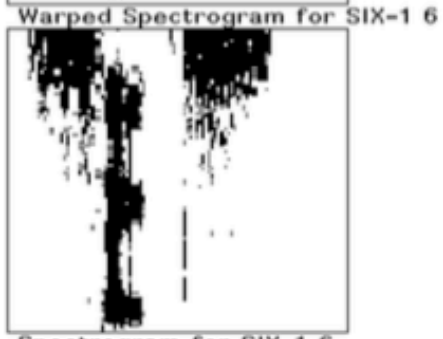
DTW example

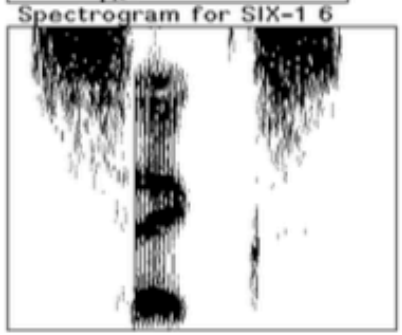

5 3 4 4 1
6
H
4 2 3 3 2
E F
3 1 1 1 3
5
A D G
3
2 2 3 3 4
B C
1 3 4 4 5
1
S


m
[m,n]
2
m-1
1
1
m-2
n-2
n-1
n

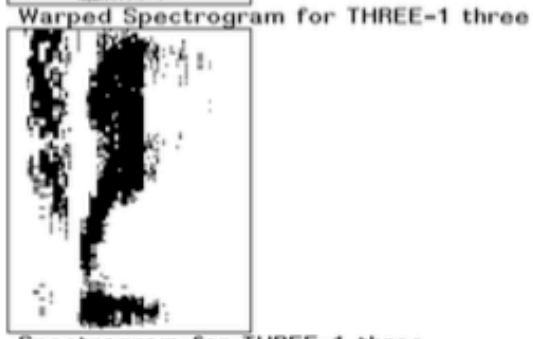
Dynamic Time Warping

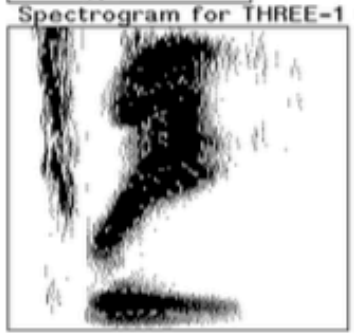






Match

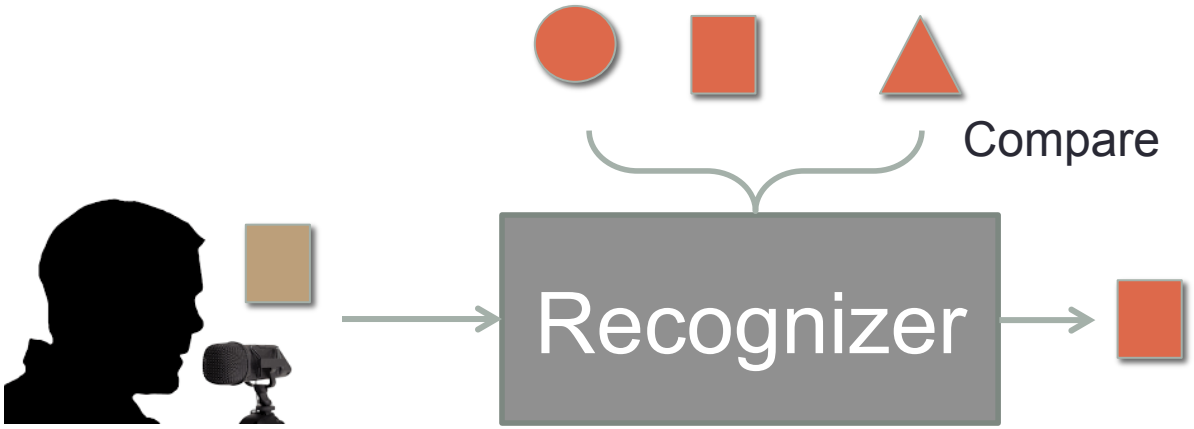




Mismatch

Downside to template matching

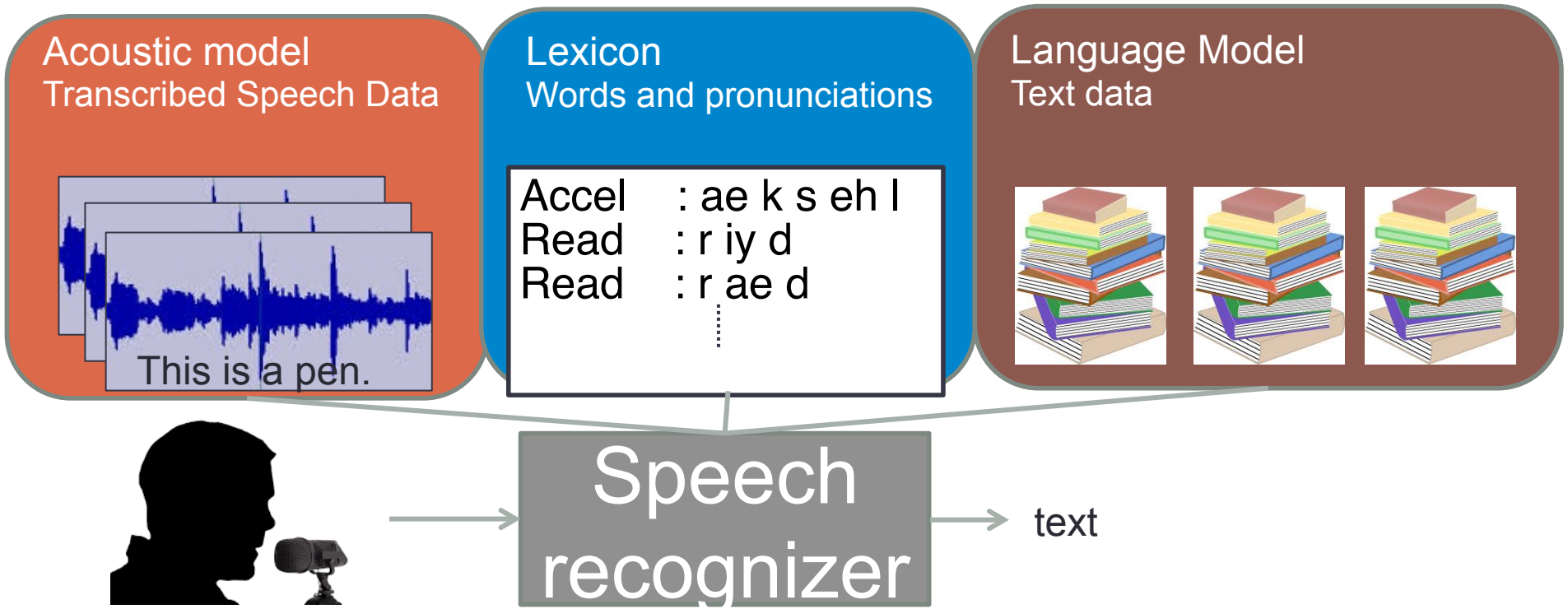
- Inflexible
  - Need new template for new words
  - Slow, 1000 words means 1000 DTWs
- Does not generalize well
  - Usually one template is not enough per class
  - Different gender, different accept, age, background noise, etc.
- Cannot do continuous speech
- Still used for key phrase detection, or speech based search



The modern speech recognizer

Turn the one recognizer block into several flexible modules

X - waveform, L - pronunciation, W - words



The ASR Equation

X - waveform, L - pronunciation, W - words

$$W^* = argmax_W P(W \mid X)$$

$$= argmax_W \frac{P(X \mid W)P(W)}{P(X)}$$

$$= argmax_W P(X \mid W)P(W)$$

$$P(X|W) = \sum_L P(X, L \mid W)$$

$$= \sum_L P(X \mid W, L)P(L \mid W)$$

$$= \sum_L P(X \mid L)P(L \mid W)$$

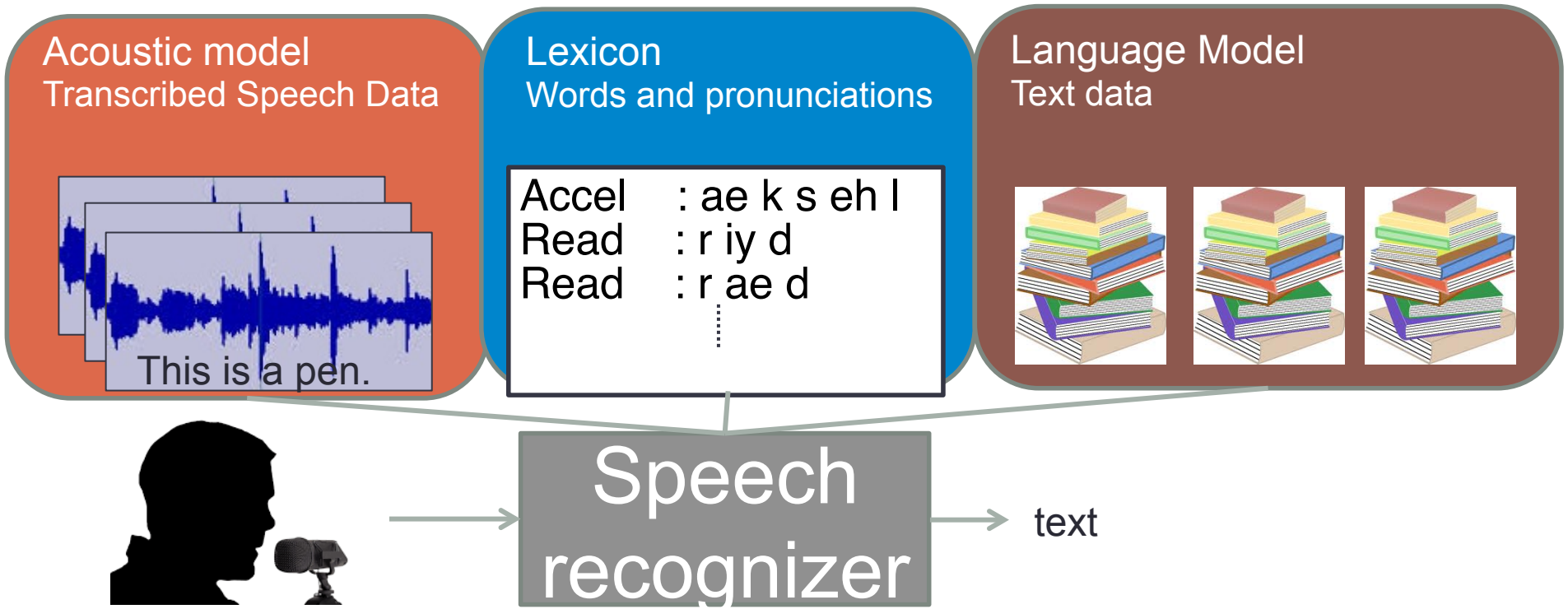
$$= argmax_W \sum_L P(X \mid L)P(L \mid W)P(W)$$

$$= argmax_{W,L} P(X \mid L)P(L \mid W)P(W)$$

The ASR equation

$$= argmax_{W,L} P(X \mid L) P(L \mid W) P(W)$$

X - waveform, L - pronunciation, W - words



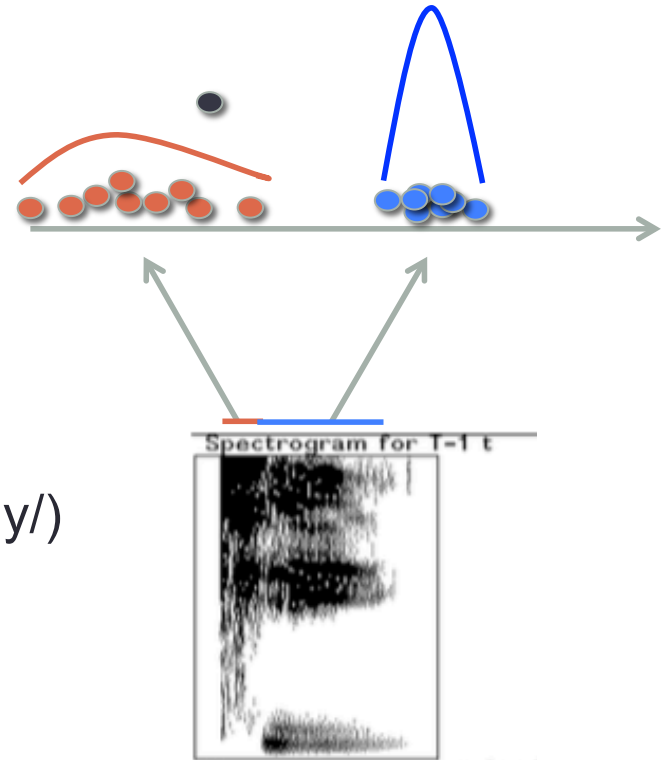
Acoustic Model

- Instead of modeling words, model subwords
  - Syllable
  - Phoneme
  - Sub-phoneme
- Can construct new words using parts
- Less class, more data per class, less overfitting problem

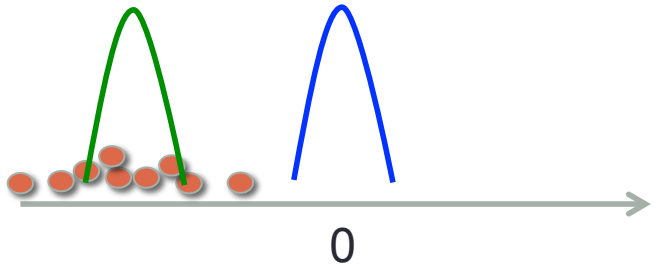
Overfit is when a model considered irrelevant information as important

Acoustic Model big picture

- Training
  - Find all frames with phoneme /iy/
  - Train a P(x | /iy/)
  - Repeat for all phonemes
- Testing
  - For each frame give probability of P(x | /iy/)
  - Repeat for all phonemes
  - Get most likely phoneme
    - Need to consider other constraint



How do we know which model is a good fit for the data



- Model A N(0,2) Model B N(-3,2)
- Maximum likelihood
  - Best model is the model that maximize the probability (likelihood) of the data
  - If each data point is independently drawn from the model, what is the probability
  - Find M that maximizes $P(x_1 \mid M)\, P(x_2 \mid M)\, P(x_3 \mid M) \ldots P(x_n \mid M)$

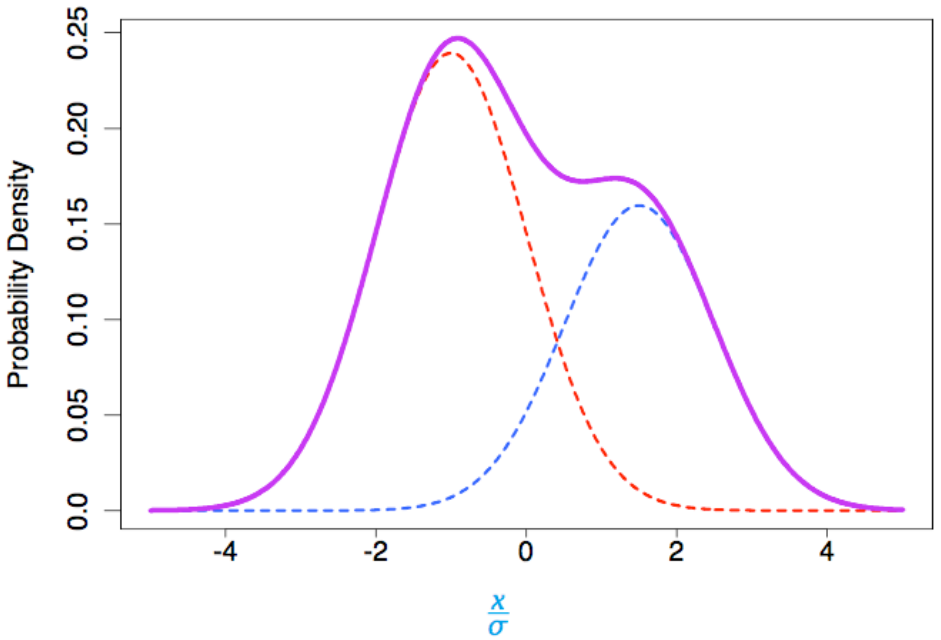
Likelihood of the data

- The best Guassian fit has the same mean and variance as the data

Gaussian Mixture Models (GMMs)

- Model each subword using GMMs
  - Gaussians cannot handle multi-modal data well
  - Consider a class can be further divided into additional factors
    - gender
    - microphone
- Mixing weight makes sure the overall probability sums to 1

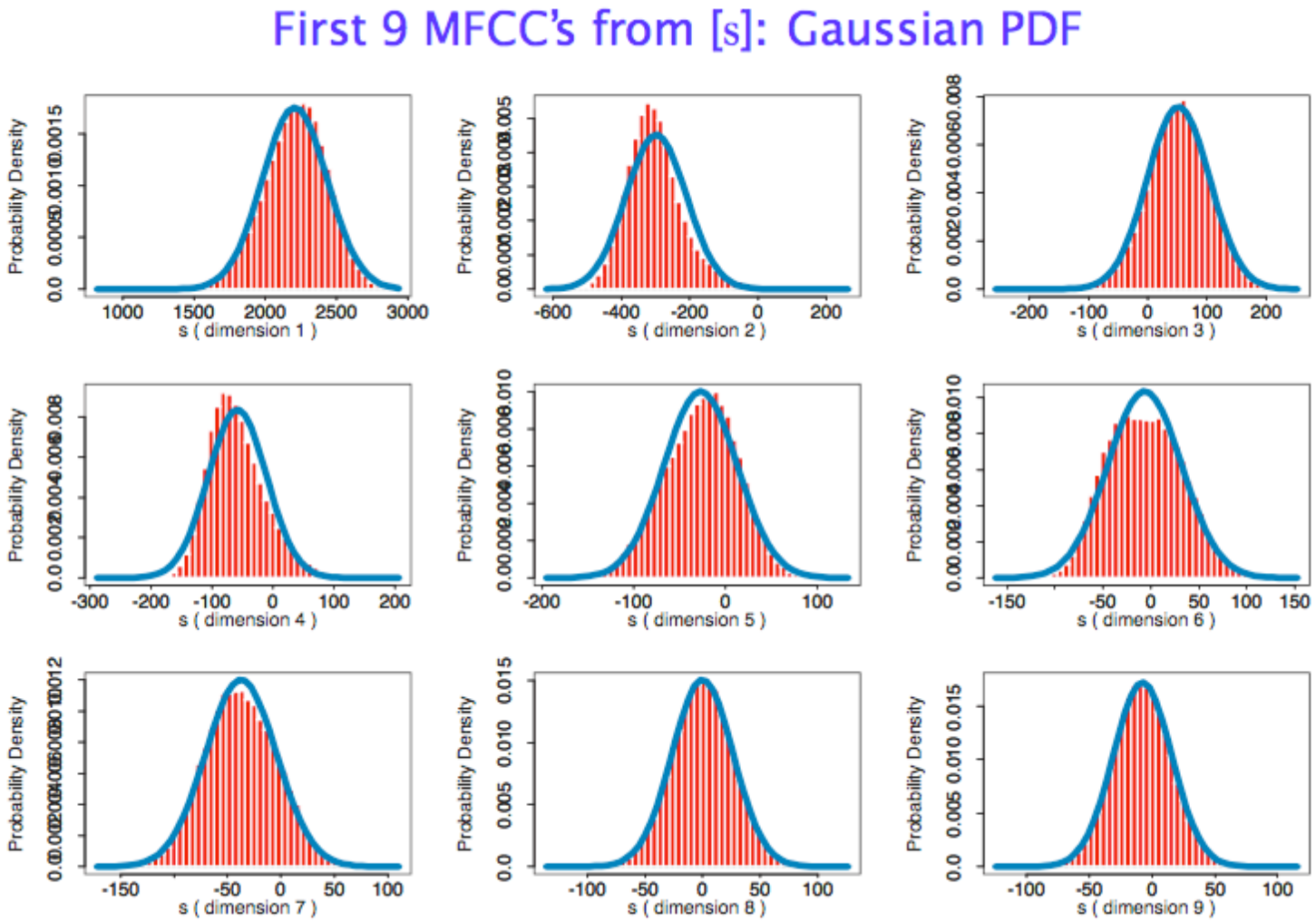
$$P(x) \sim \sum_{k=1}^{K} w_k N(\mu_k, \sigma_k)$$



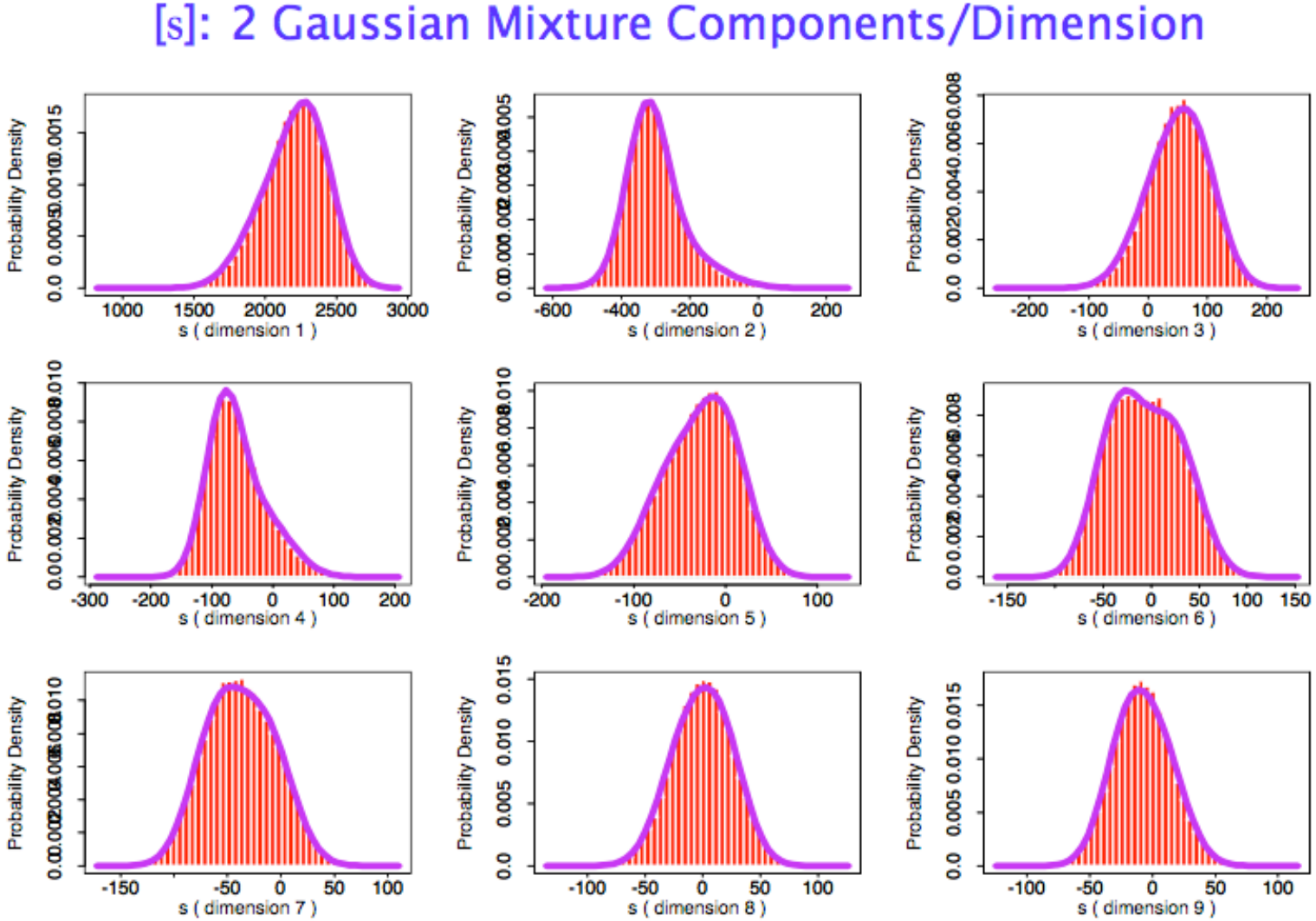
$p(x) = 0.6p_1(x) + 0.4p_2(x)$

$p_1(x) \sim N(-\sigma, \sigma^2)$ $p_2(x) \sim N(1.5\sigma, \sigma^2)$

Model of one Gaussian



Mixture of two Gaussians



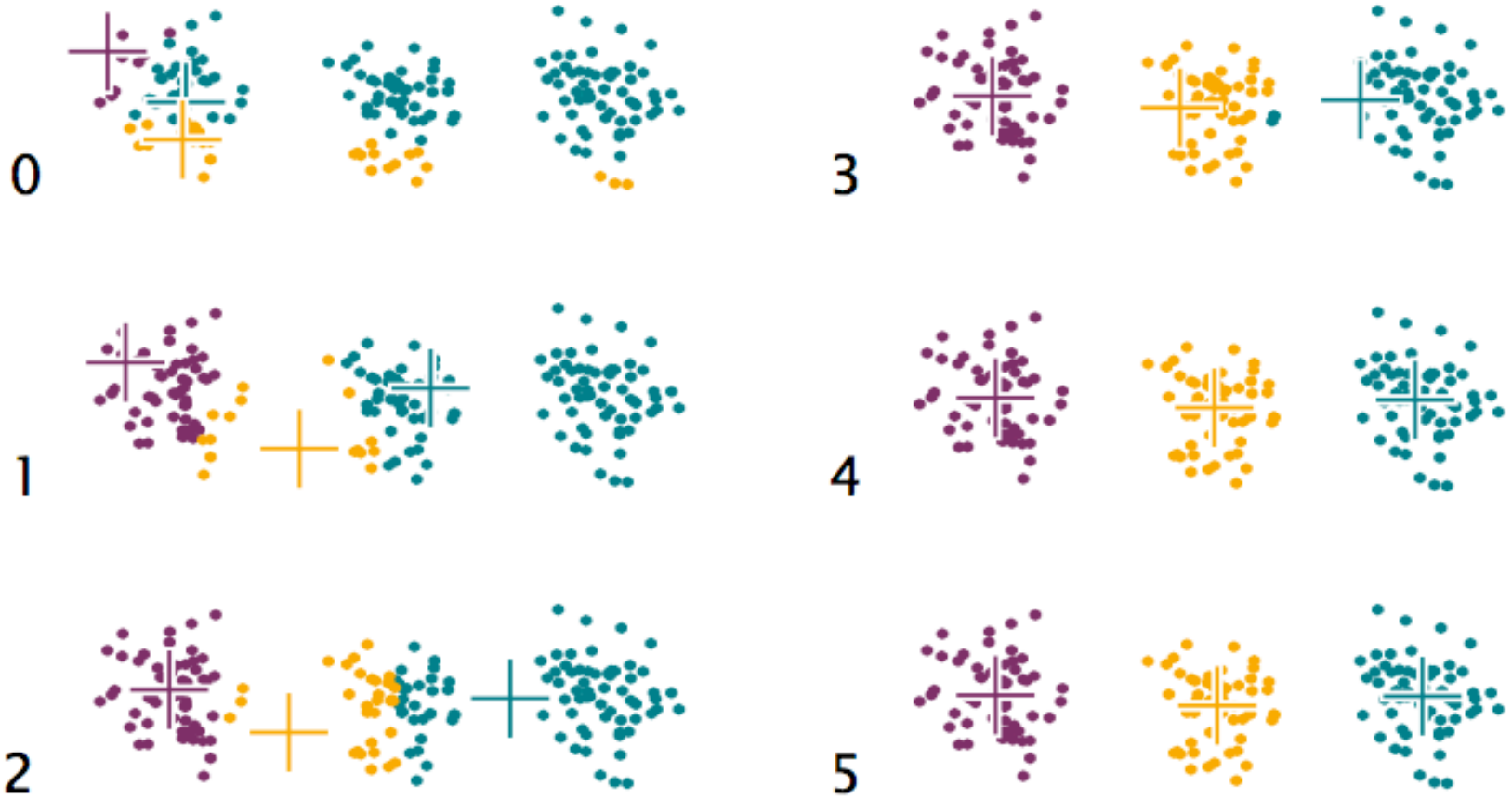
How to estimate GMM parameters

- For a Gaussian, we need to know the mean and variance of data
- For GMM, if we know which data belong to which Gaussian, we can treat it as a regular Gaussian
  - Hard to do
- Let's look at a simpler problem, K-mean clustering

K-mean clustering

- Task: cluster data into groups
- K-mean algorithm
  - Initialization: Pick K data points as cluster centers
  - Assign: Assign data points to the closest centers
  - Update: Re-compute cluster center
  - Repeat: Assign and Update

K-mean clustering



EM algorithm for GMM

- Task: cluster data into Gaussians
- EM algorithm
  - Initialization: Randomly initialize parameters Gaussians
  - Expectation: Assign data points to the closest Gaussians
  - Maximization: Re-compute Gaussians parameters according to assigned data points (find mean and variance)
  - Repeat: Expectation and Maximization

- Note: assigning data points is actually a soft assignment (with probability)

Co-articulation effect

- Phoneme /iy/ in followed by /ng/ is different than followed by /n/
- Need to model phoneme sequences
  - We call this context ***dependent*** model
  - Model sequence of two phonemes – diphone /x-a/
  - 3 phonemes – triphone /a-x-b/
  - 5 phonemes – quintphone /a-b-x-c-d/
  - /x/ is call the center phoneme
- Modeling triphones has N^3 classes!
  - Not enough data
  - Need to cluster some triphones to the same class
  - Some triphones we never seen

Tree clustering

- Goal: reduce the amount of context dependent classes
- Cluster many triphones into senones
- Cluster according to the center phones
- The center phone is split according to questions
- Select the best question according to maximum likelihood.
- Stop when not enough data to form a cluster

Likelihood $P(x_1 \mid M)\ P(x_2 \mid M)\ P(x_3 \mid M) \ldots P(x_n \mid M)$

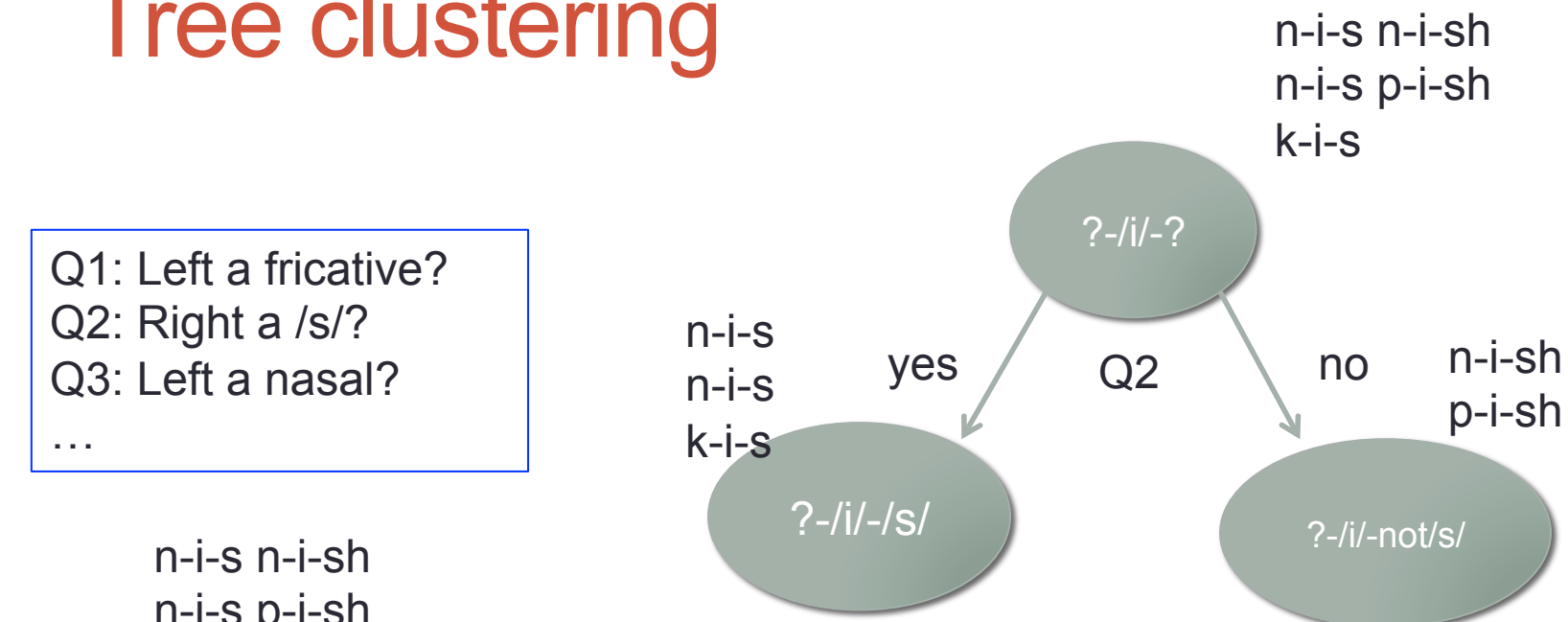
Tree clustering

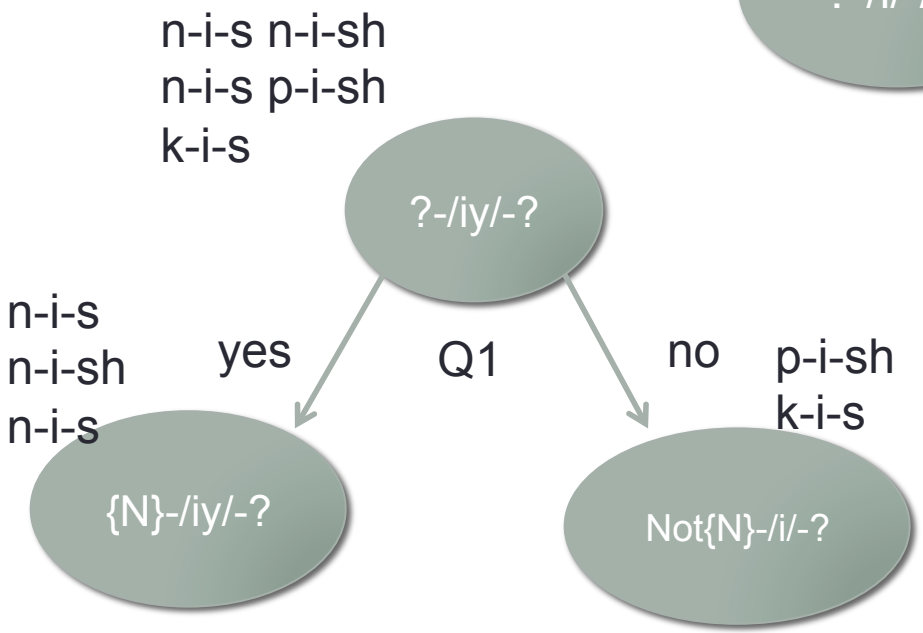
n-i-s n-i-sh
n-i-s p-i-sh
k-i-s

Q1: Left a fricative?
Q2: Right a /s/?
Q3: Left a nasal?
…

Tree clustering

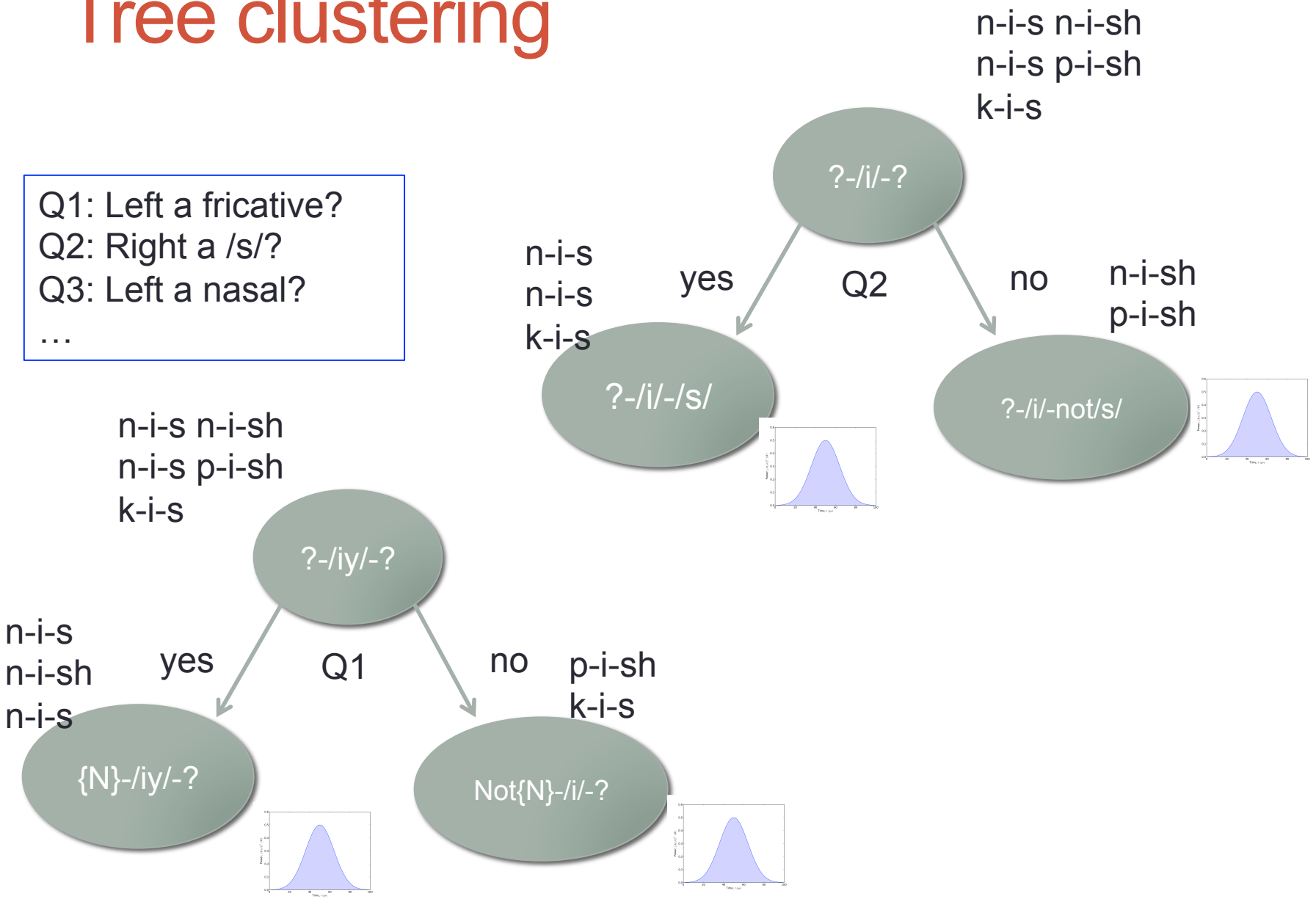
Q1: Left a fricative?
Q2: Right a /s/?
Q3: Left a nasal?
…





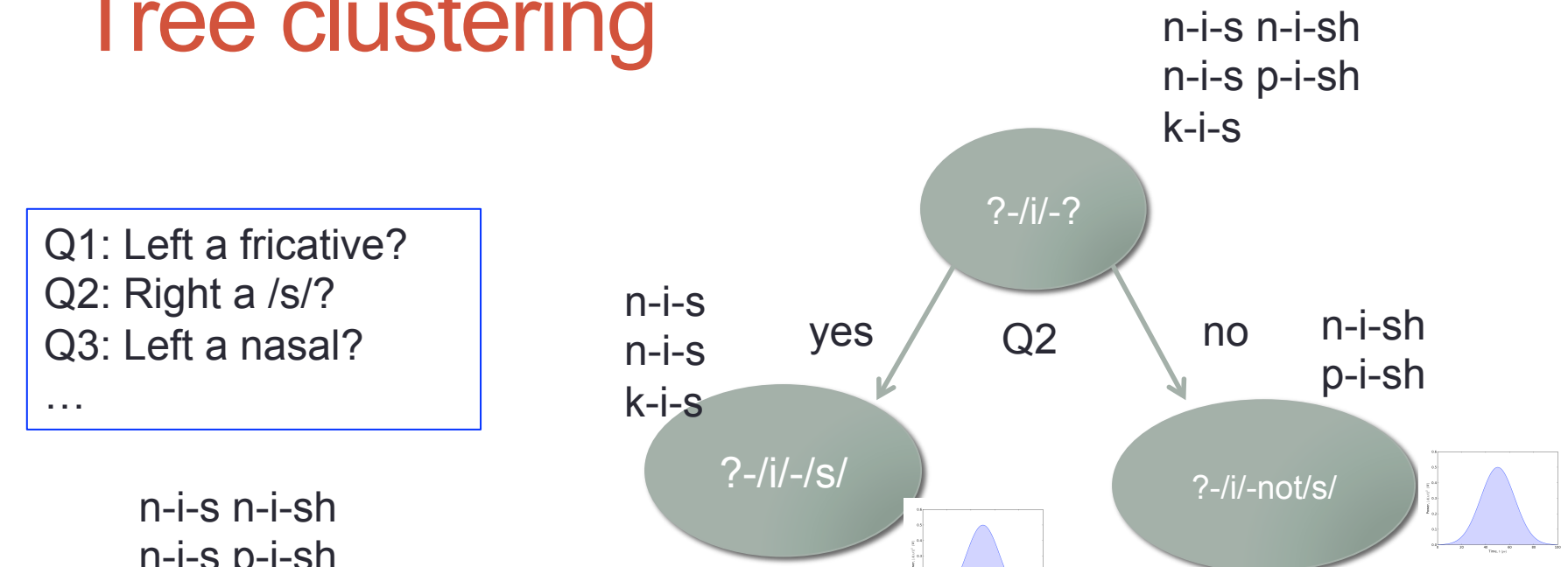
Tree clustering

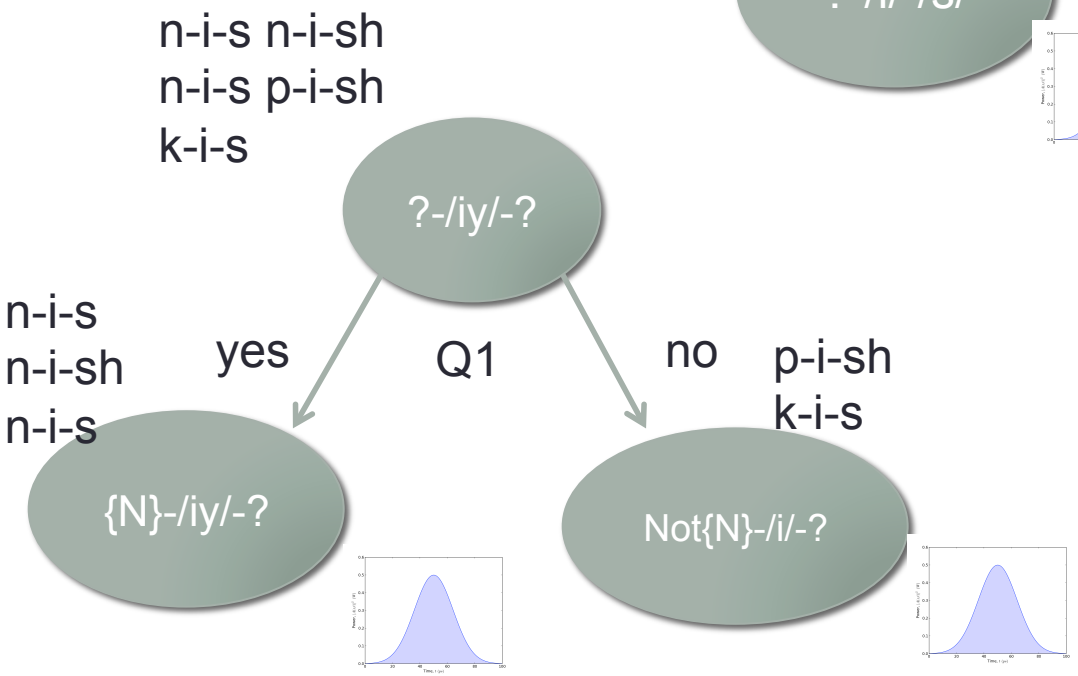
Q1: Left a fricative?
Q2: Right a /s/?
Q3: Left a nasal?
…



Tree clustering

Q1: Left a fricative?
Q2: Right a /s/?
Q3: Left a nasal?
…

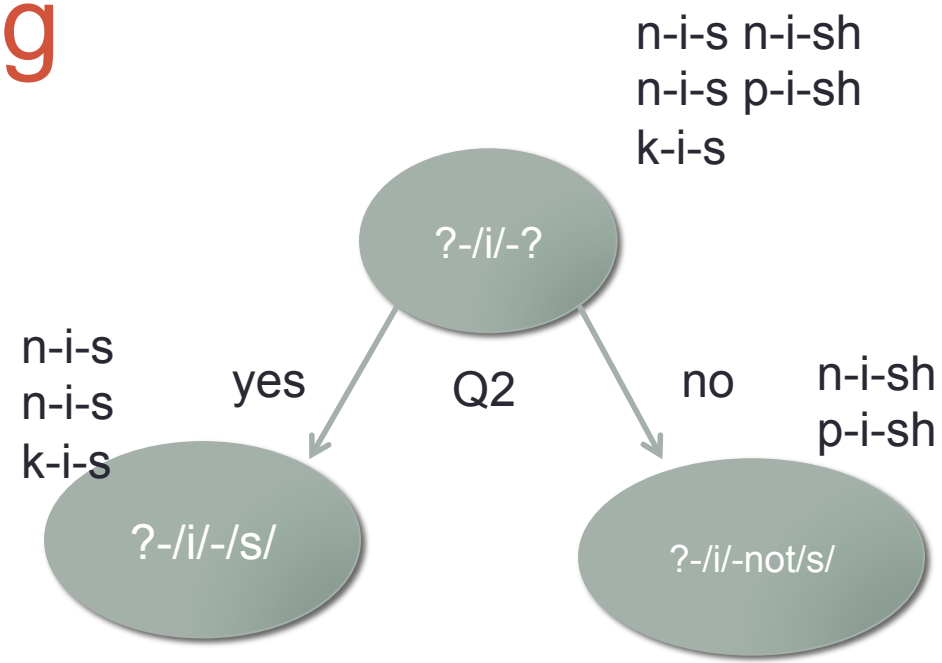




Likelihood Q2 > Likelihood Q1

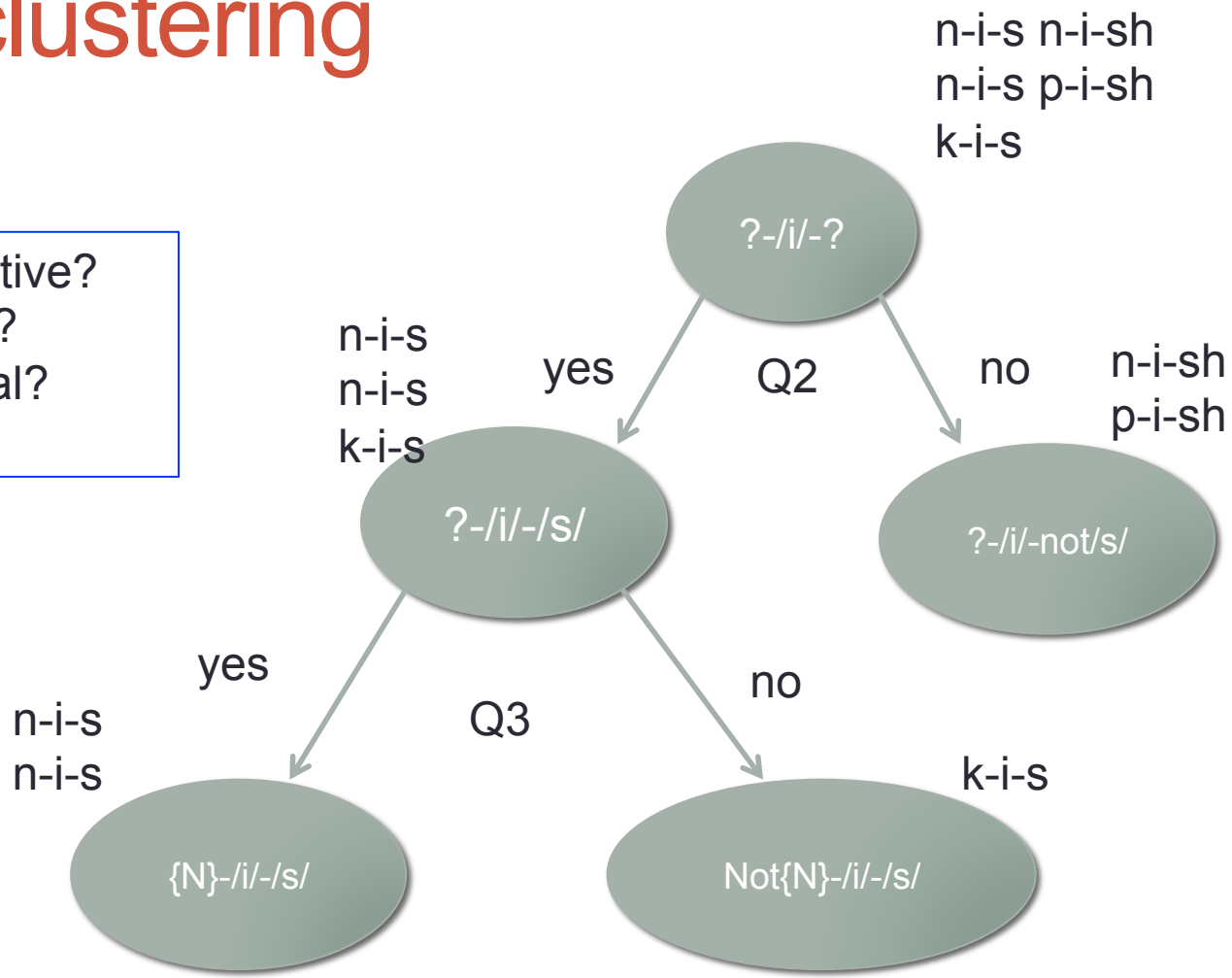
Tree clustering

Q1: Left a fricative?
Q2: Right a /s/?
Q3: Left a nasal?
…



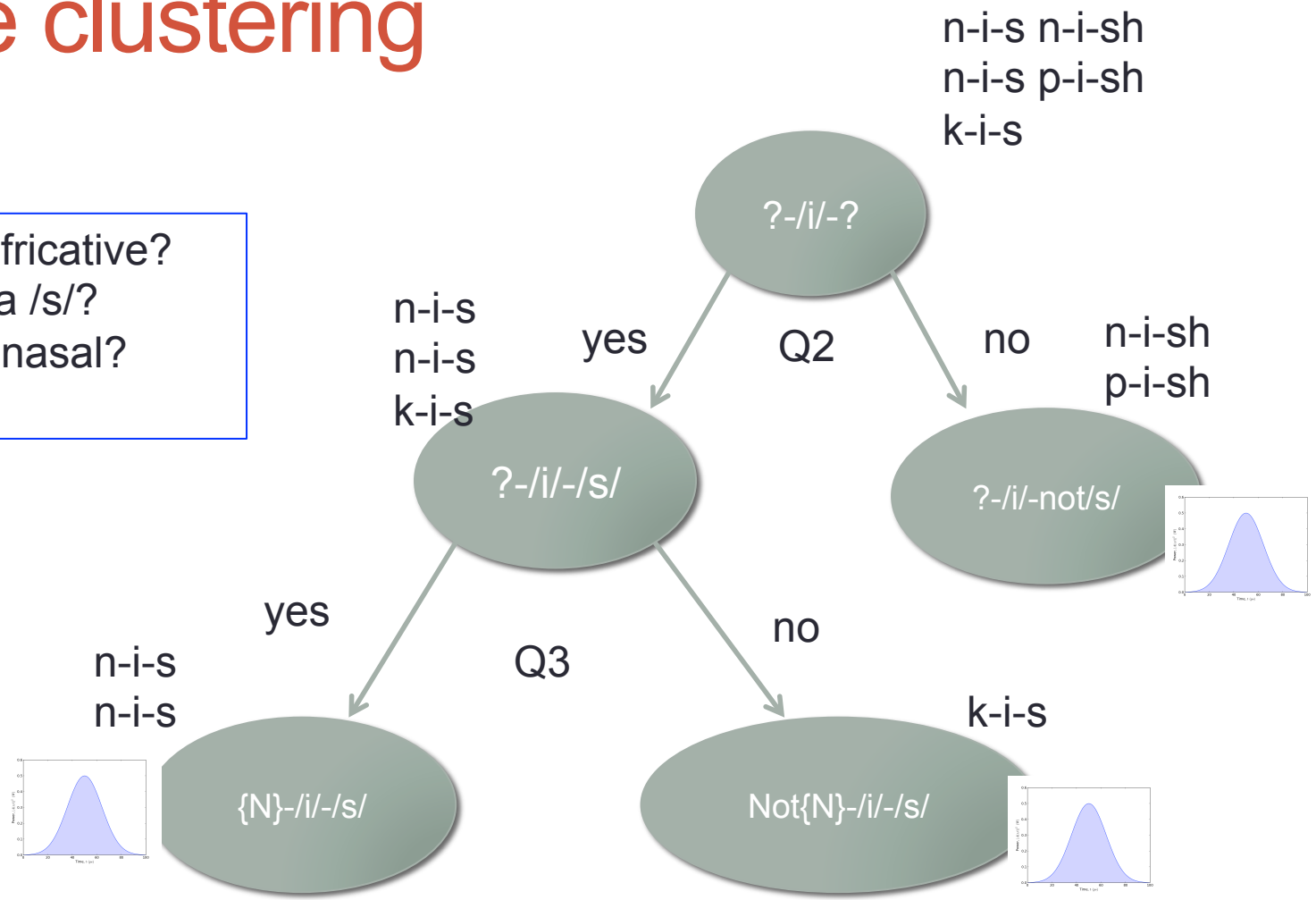
Tree clustering

Q1: Left a fricative?
Q2: Right a /s/?
Q3: Left a nasal?
…



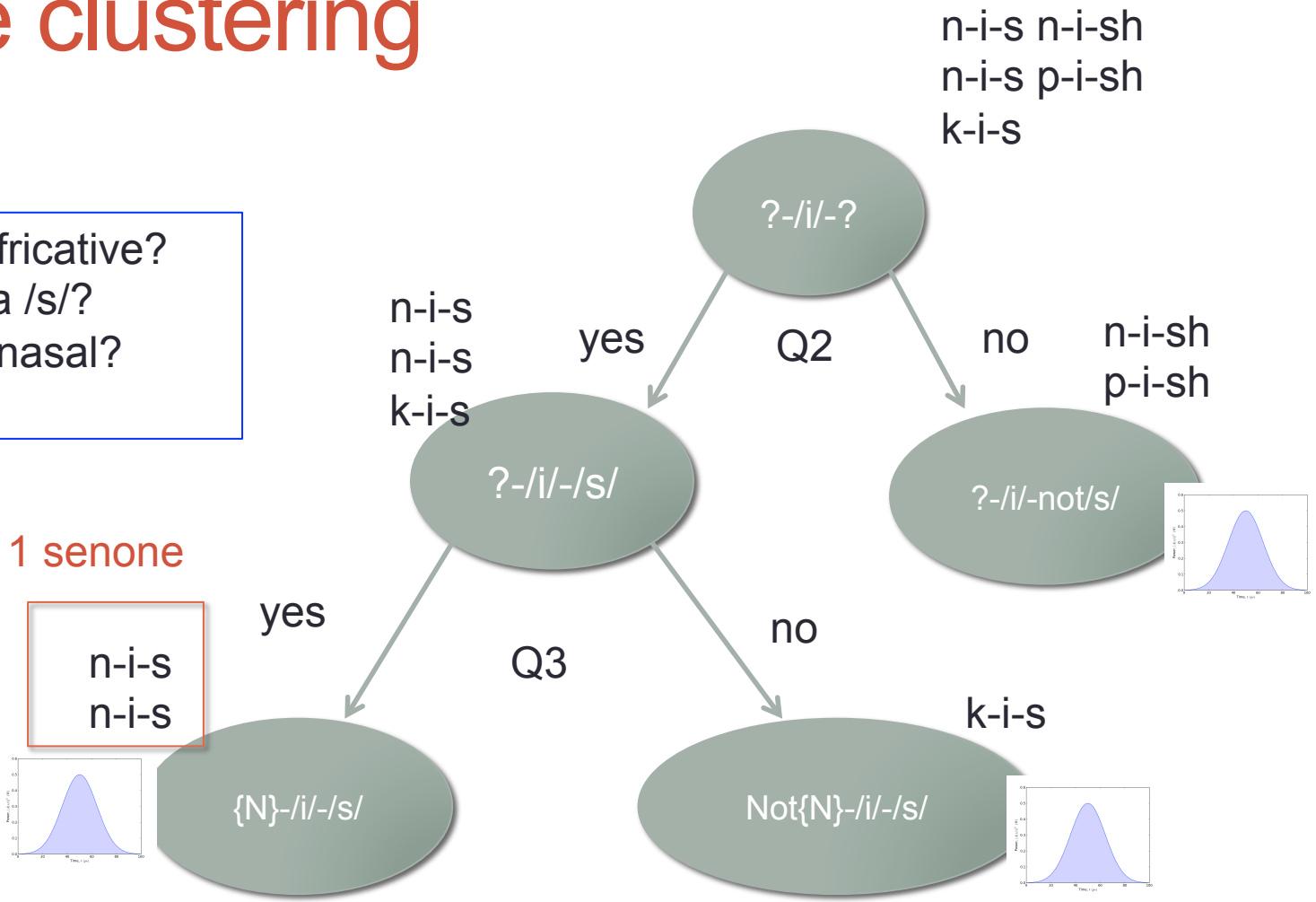
Tree clustering

Q1: Left a fricative?
Q2: Right a /s/?
Q3: Left a nasal?
…



Tree clustering

Q1: Left a fricative?
Q2: Right a /s/?
Q3: Left a nasal?
…



You now have a new word that needs b-i-s. Which model?

Homework

Piazza.com

Sign up Url: piazza.com/test_university/summer2018/2110432
Access code: 2110432

MATLAB

- Not required to do the homework
- But it's nice
- To get it, you only need to apply for an account at https://www.mathworks.com/
- Using @student.chula.ac.th, @cbs.chula.ac.th, or @md.chula.ac.th
- Activation key is 80280-30759-39553-05640-43995
- 1 year renewable
- Unlimited license (multiple installs), non-commercial use
- Comes with all toolboxes