More on Hidden Markov Models and their Applications

Lecture by Margrit Betke, Yiwen Gu

Reading: Rabiner'89, Vogler'98

Four Classes of HMM Outputs

$b_j(0)$	Scalar Output	Vector Output
Symbols: Discrete Event Space	e.g., word b_j (“baby”)	e.g., weather info b_j ([sunny, windy])
Numerical Measurements: Continuous Event Space	e.g., temperature b_j (60 F)	e.g., 3D position b_j ([x,y,z])

Last class: Discrete Scalar Output

$b_j(0)$	Scalar Output	Vector Output
Symbols: **Discrete Event Space**	e.g., word b_j (“baby”)	e.g., weather info b_j ([sunny, windy])
Numerical Measurements: **Continuous Event Space**	e.g., temperature b_j (60 F)	e.g., 3D position b_j ([x,y,z])

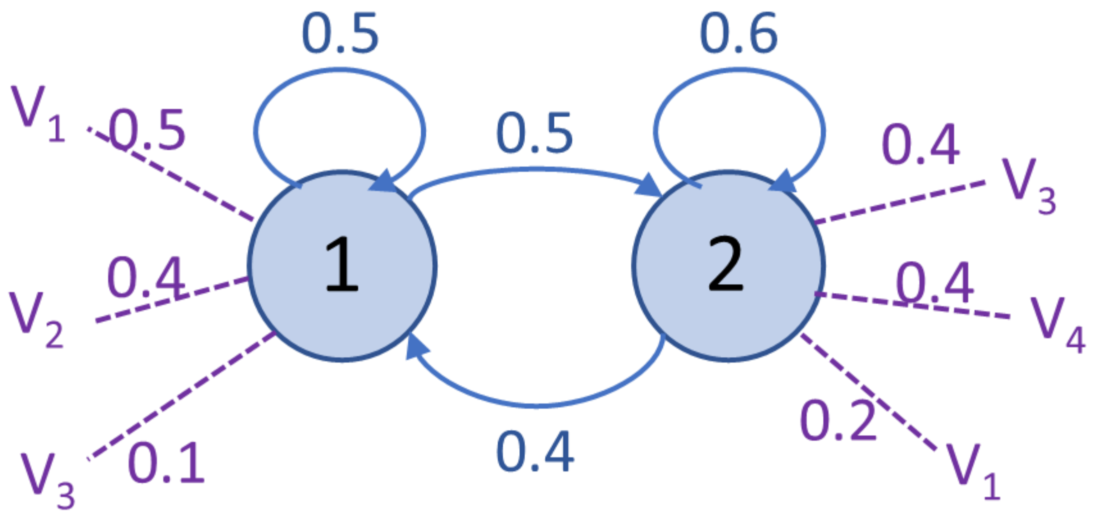
Today: Continuous Output

$b_j(0)$	Scalar Output	Vector Output
Symbols: Discrete Event Space	e.g., word b_j (“baby”)	e.g., weather info b_j ([sunny, windy])
Numerical Measurements: Continuous Event Space	e.g., temperature b_j (60 F)	e.g., 3D position b_j ([x,y,z])

Today: Continuous Output

$b_j(O)$	Scalar Output	Vector Output
Symbols: Discrete Event Space	e.g., word b_j (“baby”)	e.g., weather info b_j ([sunny, windy])
Numerical Measurements: Continuous Event Space	e.g., temperature b_j (60 F) Density Function, e.g., Normal/Gaussian: $\mathcal{N}$(rand. var, mean, variance)	e.g., 3D position b_j ([x,y,z]) $\mathcal{N}$(rand. var, mean, covariance matrix)

Recall Discrete

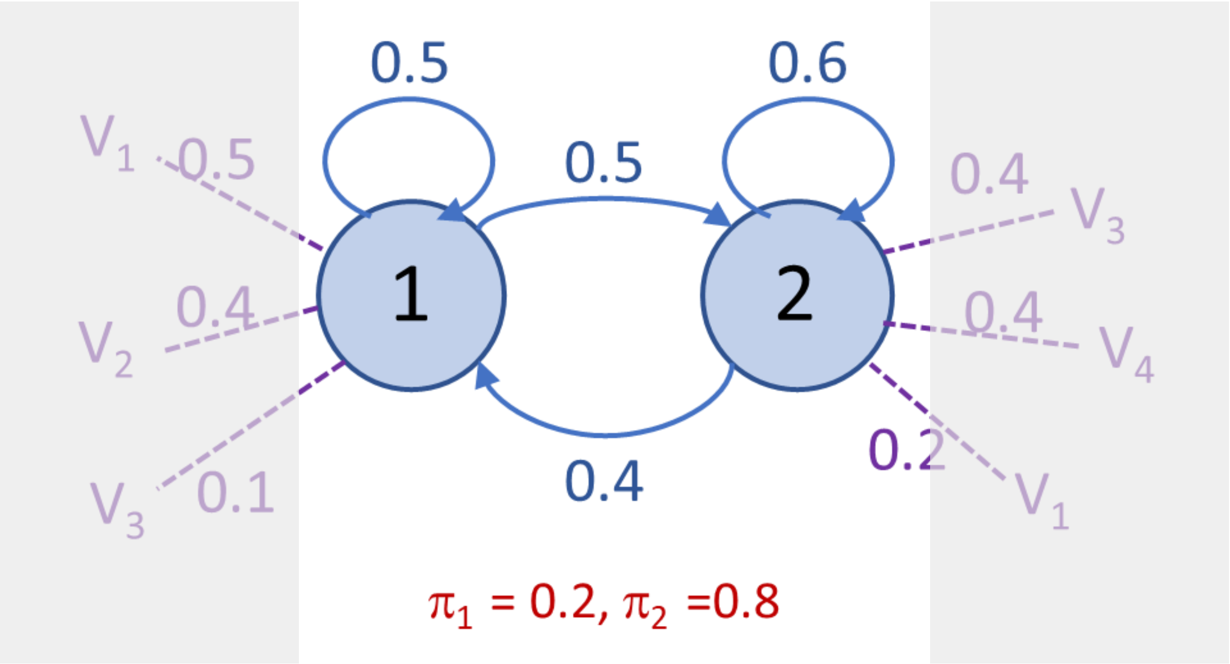


$\pi_1 = 0.2, \pi_2 = 0.8$

$b_j(k) = \text{Prob}(V_k \text{ at } t \mid q_t = S_j), \quad 1 \le j \le N, 1 \le k \le M$

$$\sum_{k=1}^{M} b_j(k) = 1$$

Continuous



$\pi_1 = 0.2, \pi_2 = 0.8$

$b_j(k) = \text{Prob}(V_k \text{ at } t \mid q_t = S_j), \quad 1 \le j \le N, 1 \le k \le M$

$$\sum_{k=1}^{M} b_j(k) = 1$$

$$\int_{-\infty}^{\infty} b_j(x)\, dx = 1$$

Mini-Intro to Estimation

Why needed?

We need to estimate the output probabilities when we train a hidden Markov model.

Mini-Intro to Estimation

Scalar world: Given x_1 , …, x_n measurements (= samples)

Average: $\bar{x} = \frac{1}{n}(x_1 + \ldots + x_n)$ (= sample mean)

Sample variance: $s^2 = \frac{1}{n}\sum_{t=1}^{n}(x_t - \bar{x})^2$

$(\bar{x}, s^2)$ are good estimates for (μ, σ^2) of a normal density function (Gaussian) $\mathcal{N}$

Mini-Intro to Estimation

Vector World: $\mathbf{x_i} = [x_{\mathrm{i}}^1, x_{\mathrm{i}}^2, \ldots, x_{\mathrm{i}}^k]^{\mathrm{T}} \epsilon \mathrm{R}^k$

Sample mean: $\bar{\mathbf{x}} = \frac{1}{n}(\mathbf{x_1} + \ldots + \mathbf{x_n})$

Sample variance: $\mathrm{s}^2 = \frac{1}{n}\sum_{t=1}^{n}(\mathbf{x_t} - \bar{\mathbf{x}})(\mathbf{x_t} - \bar{\mathbf{x}})^T$

$(\bar{\mathbf{x}}, \mathrm{s}^2)$ are good estimates for $(\boldsymbol{\mu}, \Sigma)$ of a normal density function (multivariate Gaussian) $\mathcal{N}$

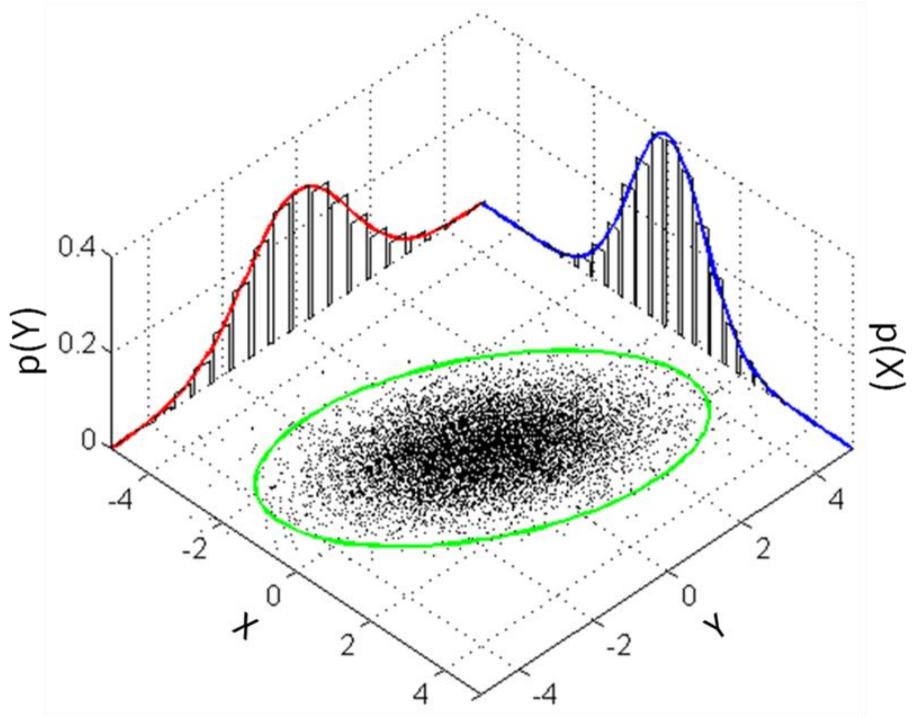
$$f_{\mathbf{X}}(x_1, \ldots, x_k) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right)}{\sqrt{(2\pi)^k |\boldsymbol{\Sigma}|}}$$

Bivariate Gaussian

$\mathbf{x_i} = [x_i^1, x_i^2, \ldots, x_i^k]^T \epsilon \mathrm{R}^k$

Bivarivate :=> k=2 => $\mathbf{x_i} = [x_i, y_i]^T$

$$f(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2[1-\rho^2]}\left[\left(\frac{x-\mu_X}{\sigma_X}\right)^2 - 2\rho\left(\frac{x-\mu_X}{\sigma_X}\right)\left(\frac{y-\mu_Y}{\sigma_Y}\right) + \left(\frac{y-\mu_Y}{\sigma_Y}\right)^2\right]\right)$$



$$\boldsymbol{\mu} = \begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix},$$

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{pmatrix}.$$

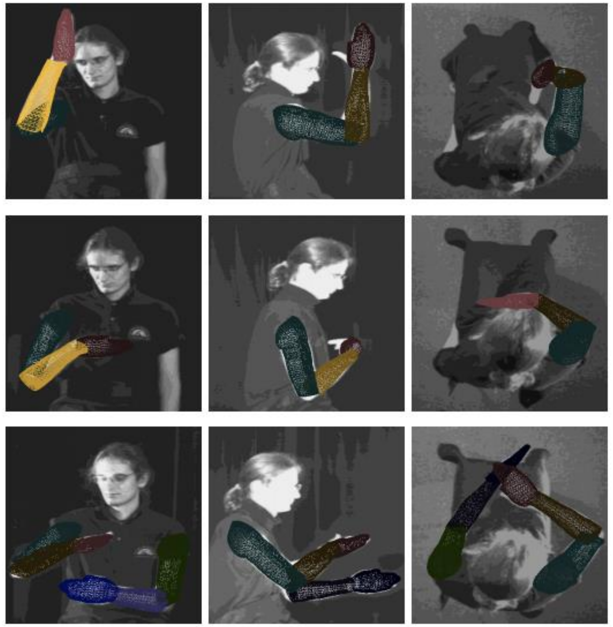
ρ = correlation coefficient

Image credit: User Bscan, Wikipedia

Examples of HMM Applications

- American Sign Language (ASL) Recognition
  - ASL Recognition Based on a Coupling Between HMMs and 3D Motion Analysis (by Vogler & Metaxas)
- Speech Recognition

Data



- 53 sign vocabulary

Category	Signs used
Nouns	America, Christian, Christmas, book, brother, chair, college, family, father, friend, interpreter, language, mail, mother, name, paper, president, school, sign, sister, teacher
Pronouns	I, my, you, your, how, what, where, why
Verbs	act, can, give, have, interpret, like, make, read, sit, teach, try, visit, want, will, win
Adjectives	deaf, good, happy, relieved, sad
Other	if, from, for, hi

- Features: wrist position, orientation, velocities in 3D space

Isolated Recognition

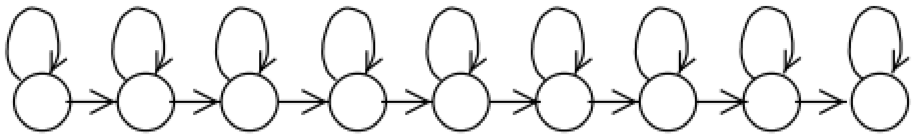
Recognize one **single** sign at a time

- Assuming each sign can be individually extracted and recognized
- Pause between individual signs as boundaries
- → 40 signs & 656 examples: ¾ training, ¼ testing (178 samples)

(with each sign has ≥ 6 examples for training and ≥ 2 examples for testing)

Vogler & Metaxas (ASL Recognition)

Isolated Recognition

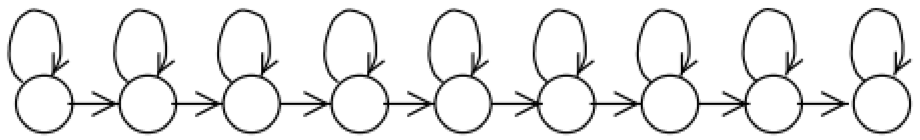


Model Design: Empirical Process

- **Number of States:** depends on frame rate, the complexity of the sign
- **Transitions**
- **Outputs:** Gaussians densities $\mathcal{N}$(mean, covariance matrices)
  - **Mixure (Multivariate) Gaussian would be better but not chosen due to lack of training data**

Vogler & Metaxas (ASL Recognition)

Isolated Recognition



Model Design:

Empirical Process

- **Number of States:** depends on frame rate, the complexity of the sign
- **Transitions**
- **Outputs:** Gaussians densities $\mathcal{N}$(mean, covariance matrices)
    - **Mixure (Multivariate) Gaussian would be better but not chosen due to lack of training data**

Training

- For each sign in the dictionary, the training procedure then computes the mean and covariance matrix over the data available for that sign and assigns them **uniformly** as the **initial output probabilities** to all states in the corresponding HMM.
    - It also assigns **initial transition probabilities** uniformly to the HMM's states.
- The training procedure then runs the **Viterbi** algorithm repeatedly on the training samples, **so as to align the training data** along the HMM's states.
    - The aligned data are then used to estimate better output probabilities for each state individually.
- After constructing these bootstrapped HMMs, the training procedure finishes by **reestimating each HMM** in turn with the **Baum-Welch** reestimation algorithm.

Isolated Recognition

Results:

Using 3D wrist position (Cartesian coordinates) only: 98.4% ± 1%

Adding wrist orientation: 98.3% ± 1%

Using just velocities: 96.9% ± 1.2%

The coordinate system was **right-handed**, with the origin at the base of the signer's spine and the **x axis facing up**.

Features	μ	σ	B	W	N
x, y, z	98.42%	0.99%	100.0%	93.8%	463
r_{xy}, θ_{xy}, z	98.72%	0.79%	100.0%	95.5%	494
$r_{xy}, r_{xz}, \theta_{xy}, \theta_{xz}, x, y, z$	98.78%	0.78%	100.0%	94.9%	882
r, θ, ϕ	96.48%	1.31%	100.0%	93.3%	210
$\dot{x}, \dot{y}, \dot{z}$	96.87%	1.21%	100.0%	93.3%	167
x, y, z, δ	98.25%	0.92%	100.0%	95.5%	167
$\dot{r}_{xy}, \dot{\theta}_{xy}, \dot{z}$	96.28%	1.04%	98.9%	93.8%	120
$\dot{r}, \dot{\theta}, \dot{\phi}$	95.89%	1.29%	98.9%	92.1%	150

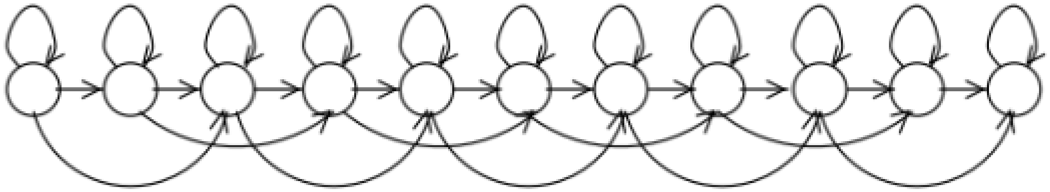
Continuous Recognition

Recognize **an entire stream** of signs at a time

- The Coarticulation Problem
  - Coarticulation: that the pronunciation of a sign is influenced by the preceding and following signs.
  - In ASL: a wide range of movements are inserted between signs
- Context-dependent HMM
- → 486 ASL sentences (2345 Signs): 389 training, 97 testing (456 Signs, covering full vocabulary)
- Recognition rate: 87%

Continuous Recognition

Model Design:



Input features: $(x, y, z, \theta_{xy}, \theta_{xz}, \dot{x}, \dot{y}, \dot{z}, \delta)$

Results:

Type of experiment	Word accuracy	Details
3D context independent	87.71%	H=416, D=8, S=32 I=16, N=456
3D context dependent	89.91%	H=424, D=6, S=26 I=14, N=456
2D context dependent	83.63%	H=394, D=14, S=44 I=16, N=452

Word Error Rate (WER) = (S+D+I)/N = (S+D+I)/(S+D+C)
where
S: num of Substitutions,
D: num of Deletions,
I: num of Insertions,
C: num of Correct words,
(the table left uses H)
N: num of words in the reference,
N= (S+D+C)

Word Accuracy = 1 – WER = (C-I)/(S+D+C)

Question: what is the WER here?

- Ground-truth (i.e. reference): *This is an example of the word error rate calculation for Boston University's CS 640.*
- Model output: *This is example the world error rate calculation for Boston University's see CS 640.*
- WER = ?

Word Error Rate (WER) =
(S+D+I)/N = (S+D+I)/(S+D+C)
where
S: num of Substitutions,
D: num of Deletions,
I: num of Insertions,
C: num of Correct words,
(the table left uses H)
N: num of words in the reference,
N= (S+D+C)

Question: what is the WER here?

- Ground-truth (i.e. reference): *This is an example of the word error rate calculation for Boston University's CS 640.*
- Model output: *This is example the world error rate calculation for Boston University's see CS 640.*
- WER = (S+D+I)/N = (S+D+I)/(S+D+C) = ?
- N = 15

Question: what is the WER here?

- Ground-truth (i.e. reference): *This is an example of the word error rate calculation for Boston University's CS 640.*
- Model output: *This is example the world error rate calculation for Boston University's see CS 640.*
- WER = (S+D+I)/N = (S+D+I)/(S+D+C) = ?
- N = 15
- S=1, D=2, I=1, C=12
- WER = (1+2+1)/(1+2+12)= 4/15 = 26.6%

BOSTON UNIVERSITY

Vogler & Metaxas

Difficulties:

Feature selection: Variability, reliability, information content

Intra and inter signer variability (e.g., length of sign)

Gaussian densities sometimes not good model

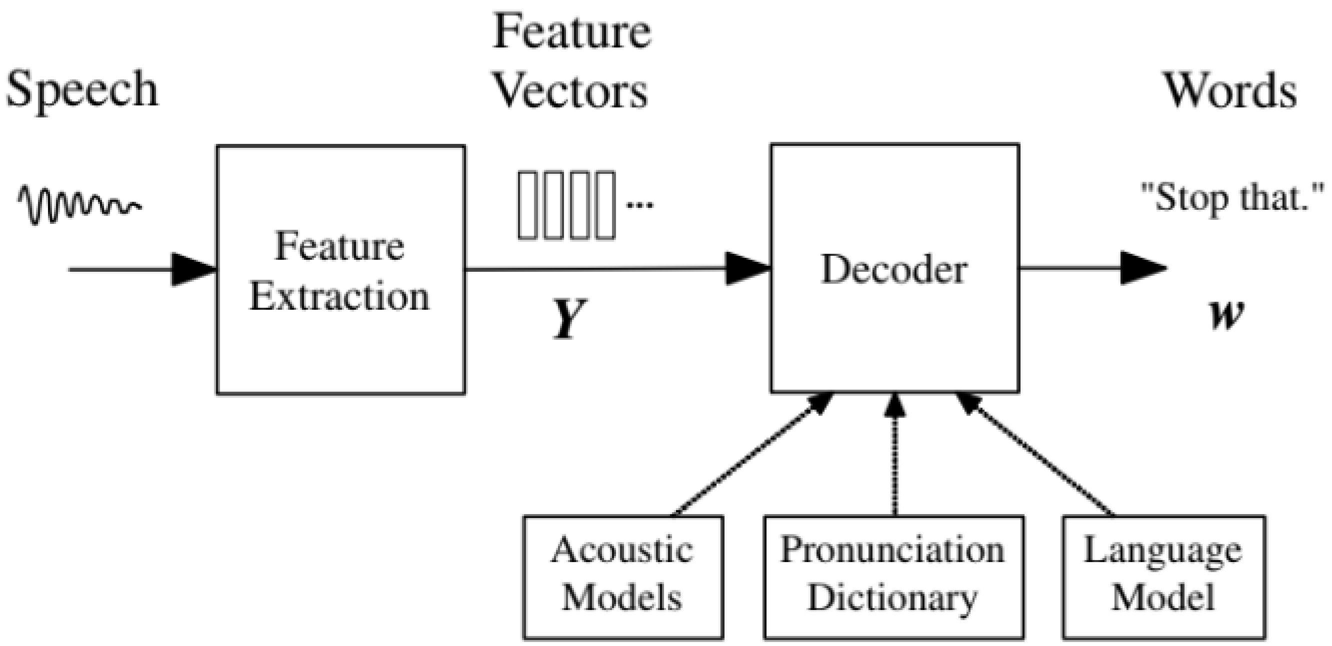
Speed up of Recognition:

Add “Beam searching” to Viterbi Algorithm:

Threshold on $d_t(i)$. If too low, partial path probability too low. Probably does not contribute to most likely path

-> Set to zero.

Speech Recognition



$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}} \{P(\boldsymbol{w}|\boldsymbol{Y})\}.$$

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}} \{\underbrace{p(\boldsymbol{Y}|\boldsymbol{w})}_{\text{acoustic model}}\underbrace{P(\boldsymbol{w})}_{\text{language model}}\}.$$

Continued in the 2024-cs640-speech-recognition.pdf

Learning Outcomes

- Understand HMM with **continuous** outputs and how it is applied in the ASL and speech recognition
- Be aware of the importance in feature selection
- Know how to evaluate ASL and speech recognition model
    - **WER** and Word Accuracy

† Highlighted (bold font) learning outcomes in the 2024-cs640-speech-recognition.pdf