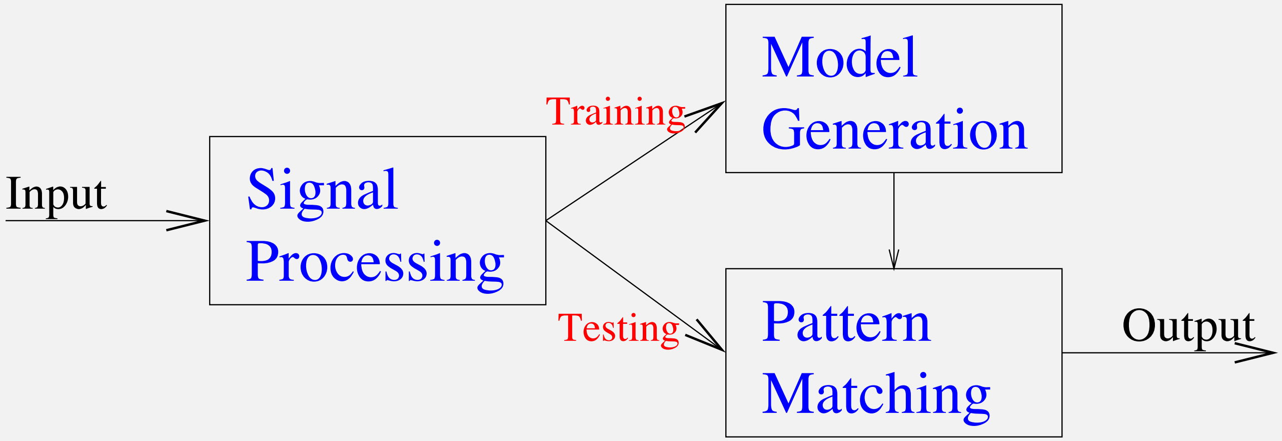
Gaussian Mixture Model (GMM)
and
Hidden Markov Model (HMM)

Samudravijaya K

Majority of the slides are taken from S.Umesh's tutorial on ASR (WiSSAP 2006).

Pattern Recognition



Input

Signal Processing

Training

Testing

Model Generation

Pattern Matching

Output

GMM: static patterns

HMM: sequential patterns

Basic Probability

Joint and Conditional probability

$$p(A,B) = p(A|B)\ p(B) = p(B|A)\ p(A)$$

Bayes' rule

$$p(A|B) = \frac{p(B|A)\ p(A)}{p(B)}$$

If A_is are mutually exclusive events,

$$p(B) = \sum_i p(B|A_i)\ p(A_i)$$

$$p(A|B) = \frac{p(B|A)\ p(A)}{\sum_i p(B|A_i)\ p(A_i)}$$

Normal Distribution

Many phenomenon are described by Gaussian pdf

$$p(x|\boldsymbol{\theta}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right) \tag{1}$$

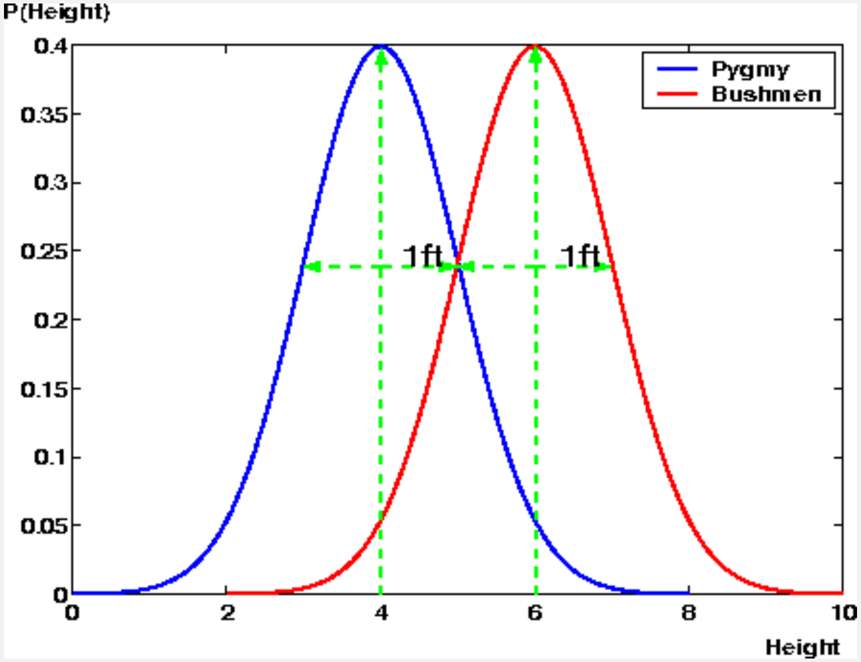
pdf is parameterised by $\boldsymbol{\theta} = [\mu, \sigma^2]$ where $\mathrm{mean} = \mu$ and variance$=\sigma^2$.

A convenient pdf: second order statistics is sufficient.

Example: Heights of Pygmies $\Rightarrow$ Gaussian pdf with $\mu = 4ft$ & std-dev$(\sigma) = 1ft$

OR: Heights of bushmen $\Rightarrow$ Gaussian pdf with $\mu = 6ft$ & std-dev$(\sigma) = 1ft$

Question:If we arbitrarily pick a person from a population $\Rightarrow$
what is the probability of the height being a particular value?

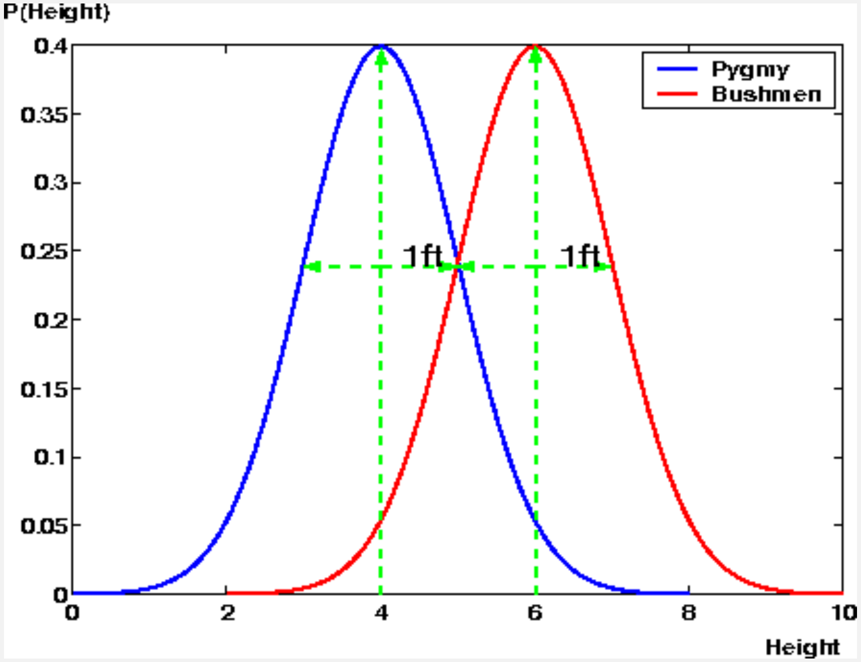


If I pick arbitrarily a Pygmy, say x, then

$$\text{Pr(Height of x=4'1")} = \frac{1}{\sqrt{2\pi.1}} \exp\left(-\frac{1}{2.1}(4'1'' - 4)^2\right) \qquad (2)$$

Note: Here mean and variances are fixed, only the observations, x, change.

Also see: $Pr(x = 4'1'') \gg Pr(x = 5')$ AND $Pr(x = 4'1'') \gg Pr(x = 3')$



Conversely: Given a person's height is $4'1''$ ⇒

Person is more likely to be a pygmy than bushman.

If we observe heights of many persons – say $3'6'', 4'1'', 3'8'', 4'5'', 4'7'', 4', 6'5''$ *and* all are from *same* population (i.e. either pygmy or bushmen.)

⇒ then more certain we are that the population is pygmy.

More the observations ⇒ better will be our decision

Likelihood Function

$x[0], x[1], \ldots, x[N-1]$

$\Rightarrow$ set of independent observations from *pdf* parameterised by θ.

Previous Example: $x[0], x[1], \ldots, x[N-1]$ are heights observed and θ is the mean of density which is unknown (σ^2 assumed known).

$$
\begin{aligned}
L(\boldsymbol{X};\theta) = p(x_0 \ldots x_{N-1};\theta) &= \prod_{i=0}^{N} p(x_i;\theta) \\
&= \frac{1}{(2\pi\sigma^2)^{N|2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=0}^{N}(x_i-\theta)^2\right) \qquad (3)
\end{aligned}
$$

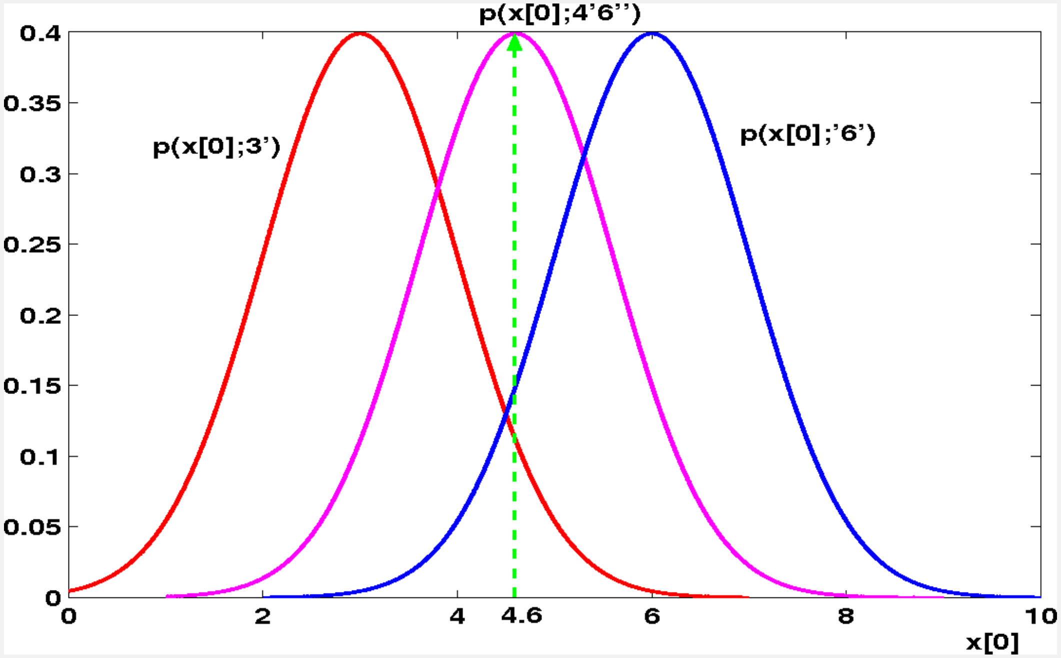
$L(\boldsymbol{X};\theta)$ is a function of θ and is called Likelihood Function

Given: $x_0 \ldots x_{N-1}$, $\Rightarrow$ what can we say about value of θ, i.e. best estimate of θ.

Maximum Likelihood Estimation

Example: We know height of a person $x[0] = 4'4''$.

Most likely to have come from which *pdf* $\Rightarrow \theta = 3',\ 4'6''$ or $6'$?



Maximum of $L(x[0]; \theta = 3'), L(x[0]; 4'6'')$ and $L(x[0]; \theta = 6') \Rightarrow$ choose $\widehat{\theta} = 4'6''$.

If θ is just a parameter, we will choose $\arg \max_{\theta} L(x[0]; \theta)$.

Maximum Likelihood Estimator

Given $x[0], x[1], \ldots, x[N-1]$ and *pdf* parameterised by $\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ . \\ . \\ \theta_{m-1} \end{bmatrix}$

We form Likelihood function $L(\boldsymbol{X}; \boldsymbol{\theta}) = \prod_{i=0}^{N} p(x_i; \boldsymbol{\theta})$

$$\widehat{\boldsymbol{\theta}}_{MLE} = \arg \max_{\boldsymbol{\theta}} L(\boldsymbol{X}; \boldsymbol{\theta})$$

For height problem:

$\Rightarrow$ can show $(\widehat{\theta})_{MLE} = \frac{1}{N} \sum x_i$

$\Rightarrow$ Estimate of mean of Gaussian $=$ sample mean of measured heights.

Bayesian Estimation

- MLE $\Rightarrow$ θ is assumed unknown but deterministic
- Bayesian Approach: θ is assumed random with pdf $p(\theta)$ $\Rightarrow$ Prior Knowledge.

$$\underbrace{p(\theta|\boldsymbol{x})}_{\text{Aposterior}} = \frac{p(\boldsymbol{x}|\theta)p(\theta)}{p(\boldsymbol{x})} \propto p(\boldsymbol{x}|\theta)\underbrace{p(\theta)}_{\text{Prior}}$$

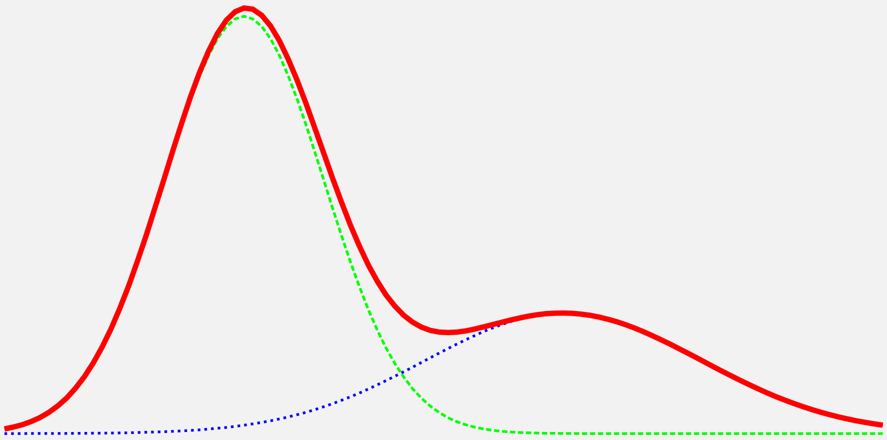
- Height problem: Unknown mean is random $\Rightarrow$ *pdf* Gaussian $\mathcal{N}(\gamma, \nu^2)$

$$p(\mu) = \frac{1}{\sqrt{2\pi\nu^2}} \exp\left(-\frac{1}{2\nu^2}(\mu-\gamma)^2\right)$$

$$\text{Then : } \quad (\widehat{\mu})_{Bayesian} = \frac{\sigma^2\gamma + n\nu^2\bar{x}}{\sigma^2 + n\nu^2}$$

$\Rightarrow$ Weighted average of sample mean and *a prior* mean

Gaussian Mixture Model



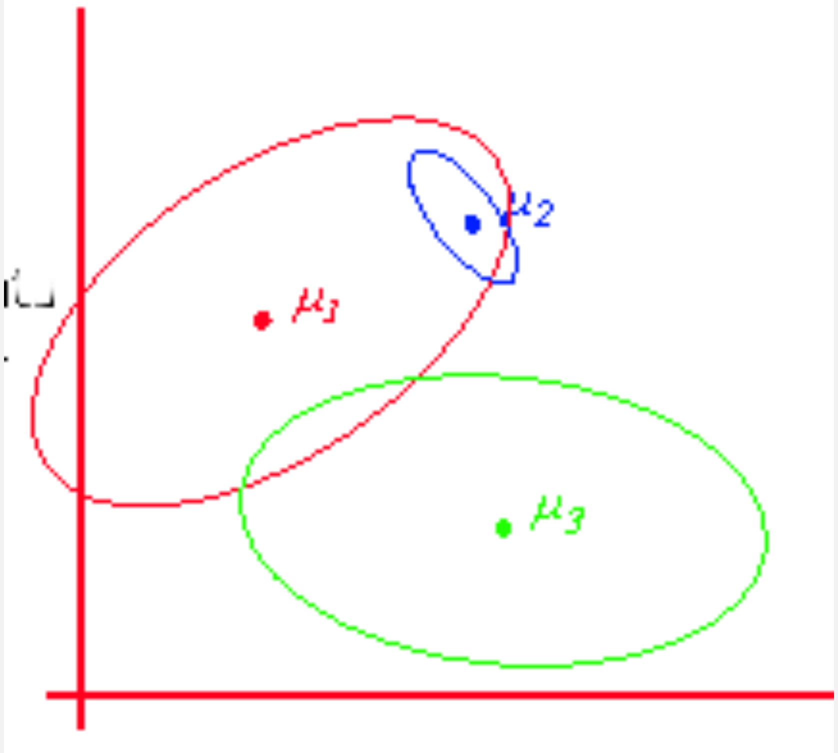
$$p(x) \;=\; \alpha\ p(x|N(\mu_1;\sigma_1)) \;+\; (1-\alpha)\ p(x|N(\mu_2;\sigma_2))$$

$$p(x) \;=\; \sum_{m=1}^{M} w_m\ p(x|N(\mu_m;\sigma_m)), \quad \sum w_i = 1$$

Characteristics of GMM:

Just like ANNs are universal approximators of functions, GMMs are universal approximators of densities (provided sufficient no. of mixtures are used); true for diagonal GMMs as well.

General Assumption in GMM

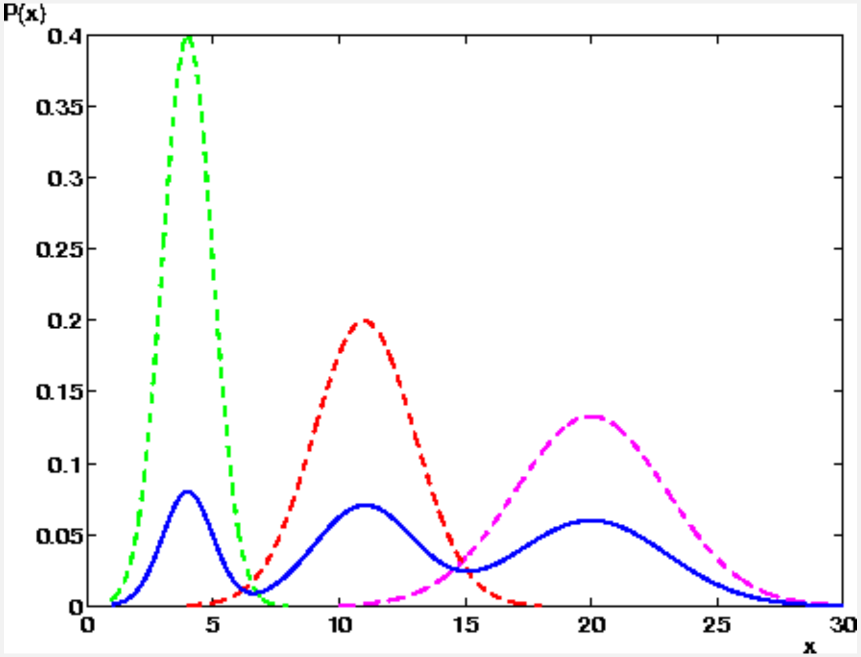


- Assume that there are M components.

- Each component generates data from a Gaussian with mean μ_m and covariance matrix Σ_m.

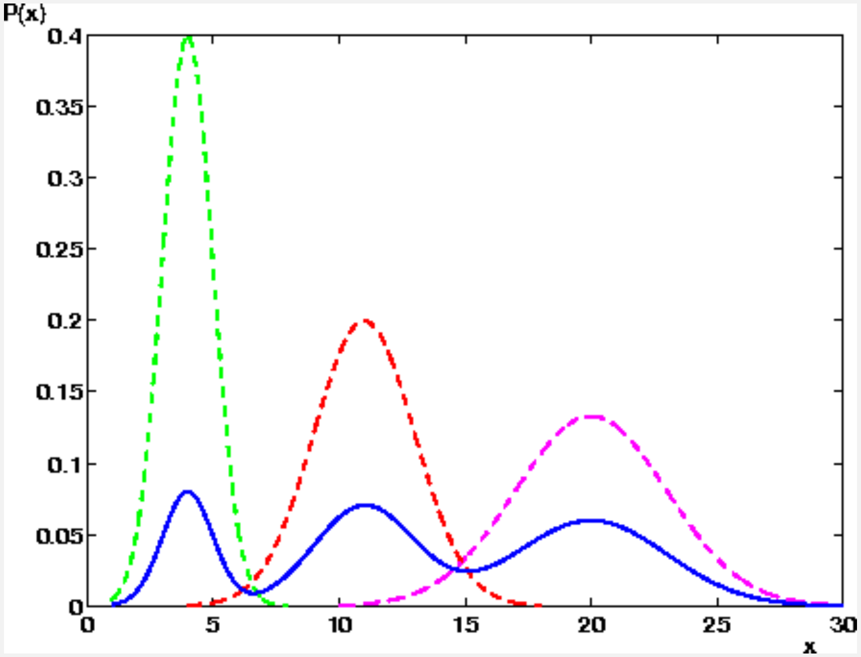
GMM

Consider the following probability density function shown in solid blue



It is useful to parameterise or “model” this seemingly arbitrary “blue” pdf

Gaussian Mixture Model (Contd.)



Actually – pdf is a mixture of 3 Gaussians, i.e.

$$p(x) = c_1 N(x; \mu_1, \sigma_1) + c_2 N(x; \mu_2, \sigma_2) + c_3 N(x; \mu_3, \sigma_3) \quad \text{and} \sum c_i = 1 \tag{4}$$

pdf parameters: $c_1, c_2, c_3, \mu_1, \mu_2, \mu_3, \sigma_1, \sigma_2, \sigma_3$

Observation from GMM

Experiment: An urn contains balls of 3 different colurs: red, blue or green. Behind a curtain, a person picks a ball from urn

If red ball $\Rightarrow$ generate $x[i]$ from $N(x;\mu_1,\sigma_1)$

If blue ball $\Rightarrow$ generate $x[i]$ from $N(x;\mu_2,\sigma_2)$

If green ball $\Rightarrow$ generate $x[i]$ from $N(x;\mu_3,\sigma_3)$

We have access *only* to observations $x[0], x[1], \ldots, x[N-1]$

$$\text{Therefore} : p(x[i];\boldsymbol{\theta}) = c_1 N(x;\mu_1,\sigma_1) + c_2 N(x;\mu_2,\sigma_2) + c_3 N(x;\mu_3,\sigma_3)$$

but we do not which urn $x[i]$ comes from!

Can we estimate component $\boldsymbol{\theta} = [c_1\, c_2\, c_3\, \mu_1\, \mu_2\, \mu_3\, \sigma_1\, \sigma_2\, \sigma_3]^T$ from the observations?

$$\arg\ \max_{\boldsymbol{\theta}}\ \ p(\boldsymbol{X};\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{N} p(x_i;\boldsymbol{\theta}) \tag{5}$$

Estimation of Parameters of GMM

Easier Problem: We know the component for each observation

Obs:	x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]	x[7]	x[8]	x[9]	x[10]	x[11]	x[12]
Comp.	1	2	2	1	3	1	3	3	2	2	3	3	3

$\boldsymbol{X}_1 = \{x[0], x[3], x[5]\ \}$ belong to $p_1(x; \mu_1, \sigma_1)$

$\boldsymbol{X}_2 = \{x[1], x[2], x[8], x[9]\ \}$ belongs to $p_2(x; \mu_2, \sigma_2)$

$\boldsymbol{X}_3 = \{x[3], x[6], x[7], x[10], x[11], x[12]\}$ belongs to $p_3(x; \mu_3, \sigma_3)$

From: $\boldsymbol{X}_1 = \{x[0], x[3], x[5]\ \}$

$\widehat{c}_1 = \frac{3}{13}$

and $\widehat{\mu}_1 = \frac{1}{3}\{x[0] + x[3] + x[5]\}$

$$\widehat{\sigma}_1^2 = \frac{1}{3}\left\{(x[0] - \widehat{\mu}_1)^2 + (x[2] - \widehat{\mu}_1)^2 + (x[5] - \widehat{\mu}_1)^2\right\}$$

In practice we do *not* know which observation come from which *pdf*.

$\Rightarrow$ How do we solve for $\arg\max_{\boldsymbol{\theta}} p(X; \boldsymbol{\theta})$?

Incomplete & Complete Data

$x[0], x[1], \ldots, x[N-1]$ ⇒ incomplete data,

Introduce another set of variables $y[0], y[1], \ldots, y[N-1]$
such that $y[i] = 1$ if $x[i] \in p_1$, $y[i] = 2$ if $x[i] \in p_2$ and $y[i] = 3$ if $x[i] \in p_3$

Obs:	x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]	x[7]	x[8]	x[9]	x[10]	x[11]	x[12
Comp.	1	2	2	1	3	1	3	3	2	2	3	3	3
miss:	y[0]	y[1]	y[2]	y[3]	y[4]	y[5]	y[6]	y[7]	y[8]	y[9]	y[10]	y[11]	y[12

$y[i]$ = missing data—unobserved data ⇒ information about component

$\boldsymbol{z} = (\boldsymbol{x}; \boldsymbol{y})$ is complete data ⇒ observations and which density they come from

$$p(\boldsymbol{z}; \boldsymbol{\theta}) = p(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\theta})$$

$$\Rightarrow \widehat{\boldsymbol{\theta}}_{MLE} = \arg \max_{\boldsymbol{\theta}} p(\boldsymbol{z}; \boldsymbol{\theta})$$

Question: *But how do we find which observation belongs to which density ?*

Given observation $x[0]$ and $\boldsymbol{\theta^g}$, what is the probability of $x[0]$ coming from first distribution?

$$p\left(y[0]=1|x[0];\boldsymbol{\theta^g}\right)$$

$$=\frac{p(y[0]{=}1,x[0];\boldsymbol{\theta^g})}{p(x[0];\boldsymbol{\theta^g})}$$

$$=\frac{p(x[0]|y[0]{=}1;\mu_1^g,\sigma_1^g)\ \cdot\ p(y[0]{=}1)}{\sum_{j=1}^{3}p(x[0]|y[0]{=}j;\boldsymbol{\theta^g})\ p(y[0]{=}j)}$$

$$=\frac{p(x[0]|y[0]{=}1,\mu_1^g,\sigma_1^g)\ \cdot\ c_1^g}{p(x[0]|y[0]{=}1;\mu_1^g,\sigma_1^g)\ c_1^g+p(x[0]|y[0]{=}2;\mu_2^g,\sigma_2^g)\ c_2^g+\ldots}$$

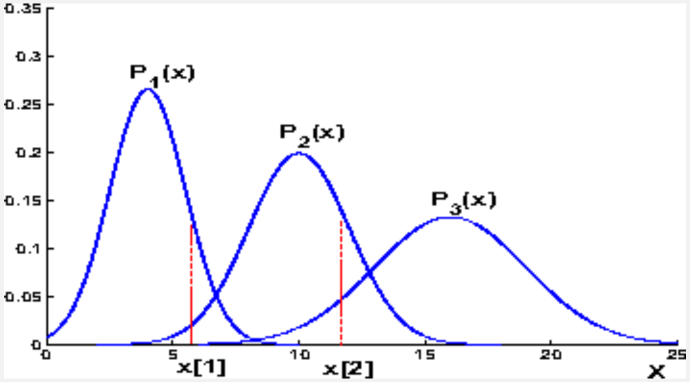
All parameters are known $\Rightarrow$ we can calculate $p(y[0]=1|x[0];\boldsymbol{\theta^g})$

(Similarly calculate $p(y[0]=2|x[0];\boldsymbol{\theta^g})$, $p(y[0]=3|x[0];\boldsymbol{\theta^g})$)

Which density? $\Rightarrow y[0]=\arg\max_{i} p(y[0]=i|x[0];\boldsymbol{\theta^g})$ – Hard allocation

Parameter Estimation for Hard Allocation

	x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
$p(y[j]=1\|x[j];\boldsymbol{\theta}^g)$	0.5	0.6	0.2	0.1	0.2	0.4	0.2
$p(y[j]=2\|x[j];\boldsymbol{\theta}^g)$	0.25	0.3	0.75	0.3	0.7	0.5	0.6
$p(y[j]=3\|x[j];\boldsymbol{\theta}^g)$	0.25	0.1	0.05	0.6	0.1	0.1	0.2
Hard Assign.	y[0]=1	y[1]=1	y[2]=2	y[3]=3	y[4]=2	y[5]=2	y[6]=2



Updated Parameters: $\widehat{c}_1 = \frac{2}{7}$ $\quad \widehat{c}_2 = \frac{4}{7}$ $\quad \widehat{c}_3 = \frac{1}{7}$ (different from initial guess!)

Similarly (for Gaussian) find: $\widehat{\mu}_i, \quad \widehat{\sigma}_i^2$ for i^{th} pdf

Parameter Estimation for Soft Assignment

	x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
$p(y[j]=1\|x[j];\boldsymbol{\theta^g})$	0.5	0.6	0.2	0.1	0.2	0.4	0. 2
$p(y[j]=2\|x[j];\boldsymbol{\theta^g})$	0.25	0.3	0.75	0.3	0.7	0.5	0.6
$p(y[j]=3\|x[j];\boldsymbol{\theta^g})$	0.25	0.1	0.05	0.6	0.1	0.1	0.2

Example: Prob. of each sample belonging to component 1

$$p(y[0]=1|x[0];\boldsymbol{\theta^g}),\ p(y[1]=1|x[1];\boldsymbol{\theta^g})\ p(y[2]=1|x[2];\boldsymbol{\theta^g}),\ \cdots\cdots$$

Average probability that a sample belongs to Comp.#1 is

$$\begin{aligned}\widehat{c}_1^{new} &= \frac{1}{N}\sum_{i=1}^{N} p(y[i]=1|x[i];\boldsymbol{\theta^g}) \\ &= \frac{0.5+0.6+0.2+0.1+0.2+0.4+0.2}{7} = \frac{2.2}{7}\end{aligned}$$

Soft Assignment – Estimation of Means & Variances

Recall: Prob. of sample j belonging to component i

$$p(y[j] = i | x[j]; \boldsymbol{\theta^g})$$

Soft Assignment: Parameters estimated by taking weighted average !

$$\mu_1^{new} = \frac{\sum_{i=1}^{N} x_i \cdot p(y[i] = 1 \mid x[i]; \boldsymbol{\theta^g})}{\sum_{i=1}^{N} p(y[i] = 1 \mid x[i]; \boldsymbol{\theta^g})}$$

$$(\sigma_1^2)^{new} = \frac{\sum_{i=1}^{N} (x_i - \widehat{\mu}_1)^2 \cdot p(y[i] = 1 \mid x[i]; \boldsymbol{\theta^g})}{\sum_{i=1}^{N} p(y[i] = 1 \mid x[i]; \boldsymbol{\theta^g})}$$

These are updated parameters starting with initial guess $\boldsymbol{\theta^g}$

Maximum Likelihood Estimation of Parameters of GMM

1. Make initial guess of parameters: $\boldsymbol{\theta^g} = c_1^g, c_2^g, c_3^g, \mu_1^g, \mu_2^g, \mu_3^g, \sigma_1^g, \sigma_2^g, \sigma_3^g$

2. Knowing parameters $\boldsymbol{\theta^g}$, find Prob. of sample x_i belonging to j^{th} component.

$$p[y[i] = j \mid x[i]; \boldsymbol{\theta^g}] \qquad \text{for } i = 1, 2, \ldots N \Rightarrow \text{no. of observations}$$

$$\text{for } j = 1, 2, \ldots M \Rightarrow \text{no. of components}$$

3.

$$\widehat{c}_j^{new} = \frac{1}{N} \sum_{i=1}^{N} p(y[i] = j \mid x[i]; \boldsymbol{\theta^g})$$

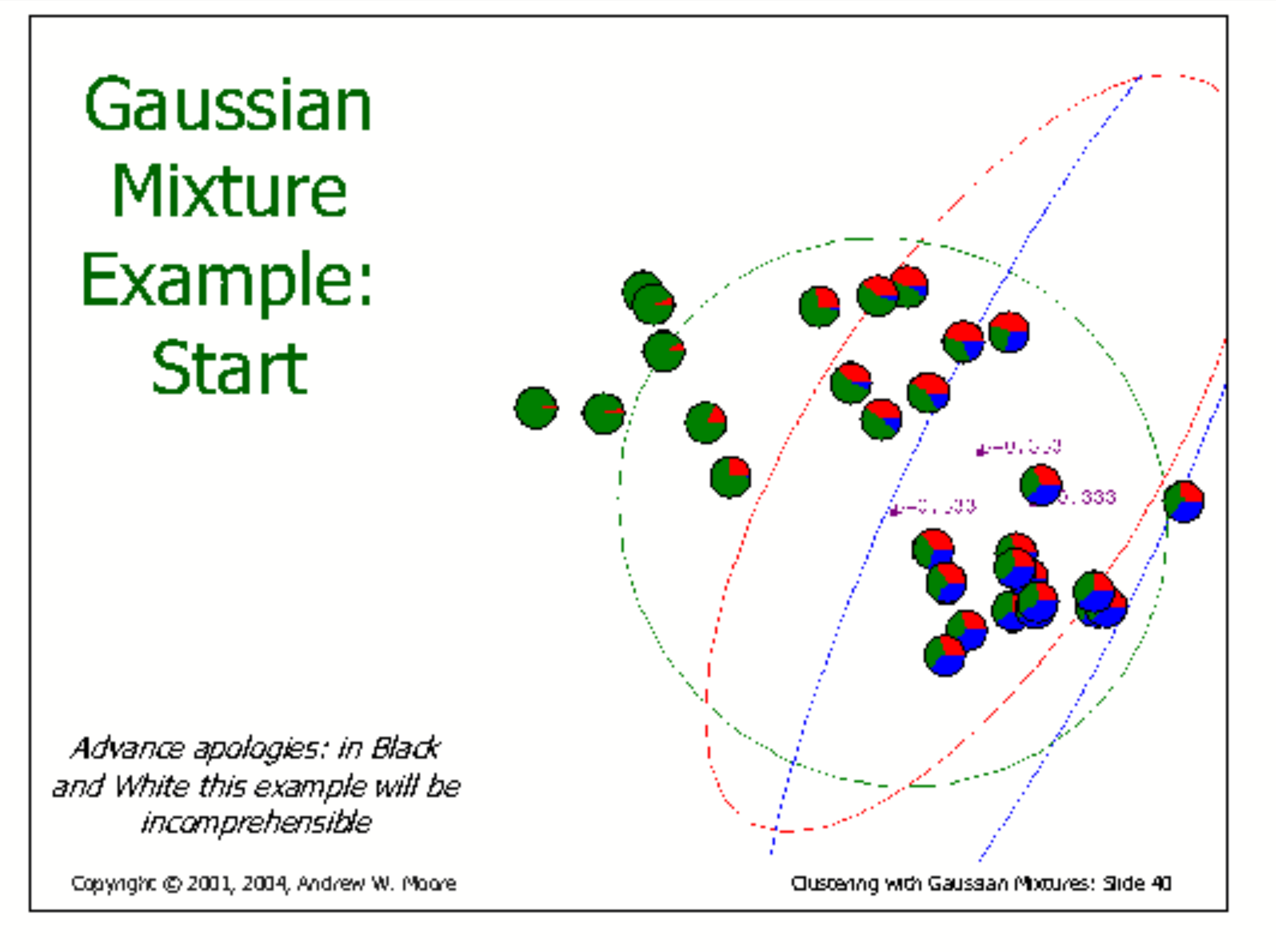
4.

$$\mu_j^{new} = \frac{\sum_{i=1}^{N} x_i \cdot p(y[i] = j \mid x[i]; \boldsymbol{\theta^g})}{\sum_{i=1}^{N} p(y[i] = j \mid x[i]; \boldsymbol{\theta^g})}$$

5.

$$(\sigma_j^2)^{new} = \frac{\sum_{i=1}^{N} (x_i - \widehat{\mu}_1)^2 \cdot p(y[i] = j \mid x[i]; \boldsymbol{\theta^g})}{\sum_{i=1}^{N} p(y[i] = j \mid x[i]; \boldsymbol{\theta^g})}$$

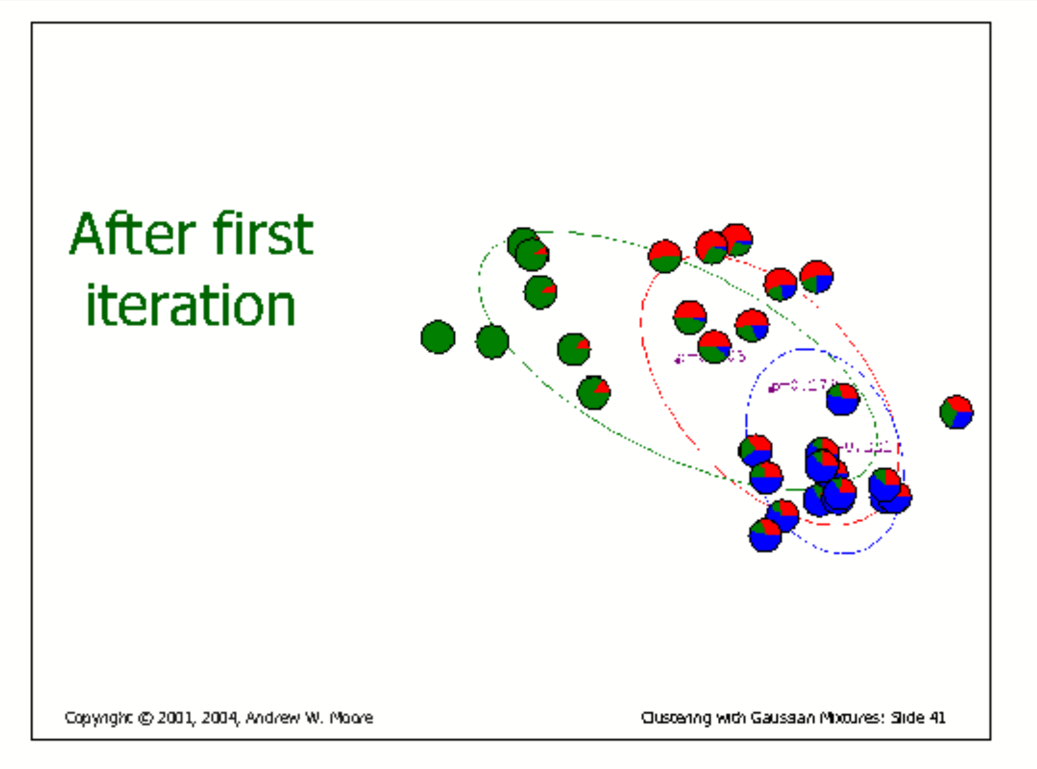
6. Go back to (2) and repeat until convergence



Gaussian Mixture Example: Start

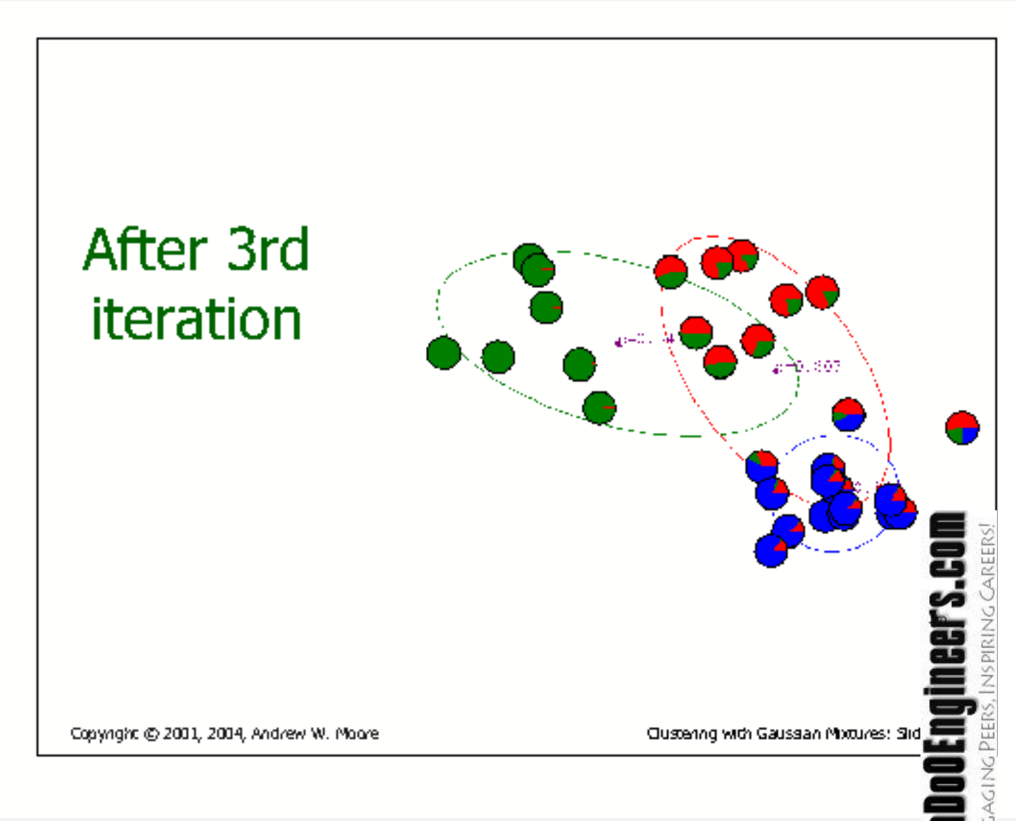
Advance apologies: in Black and White this example will be incomprehensible

Clustering with Gaussian Mixtures: Slide 40



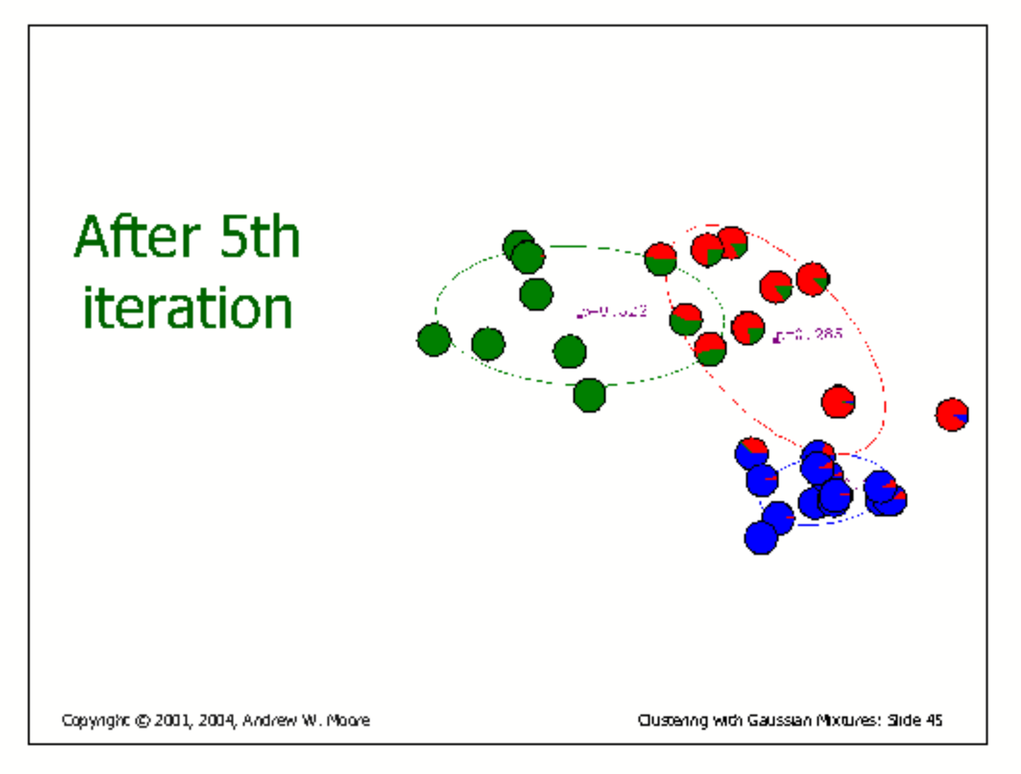
After first iteration

Clustering with Gaussian Mixtures: Slide 41



After 3rd iteration

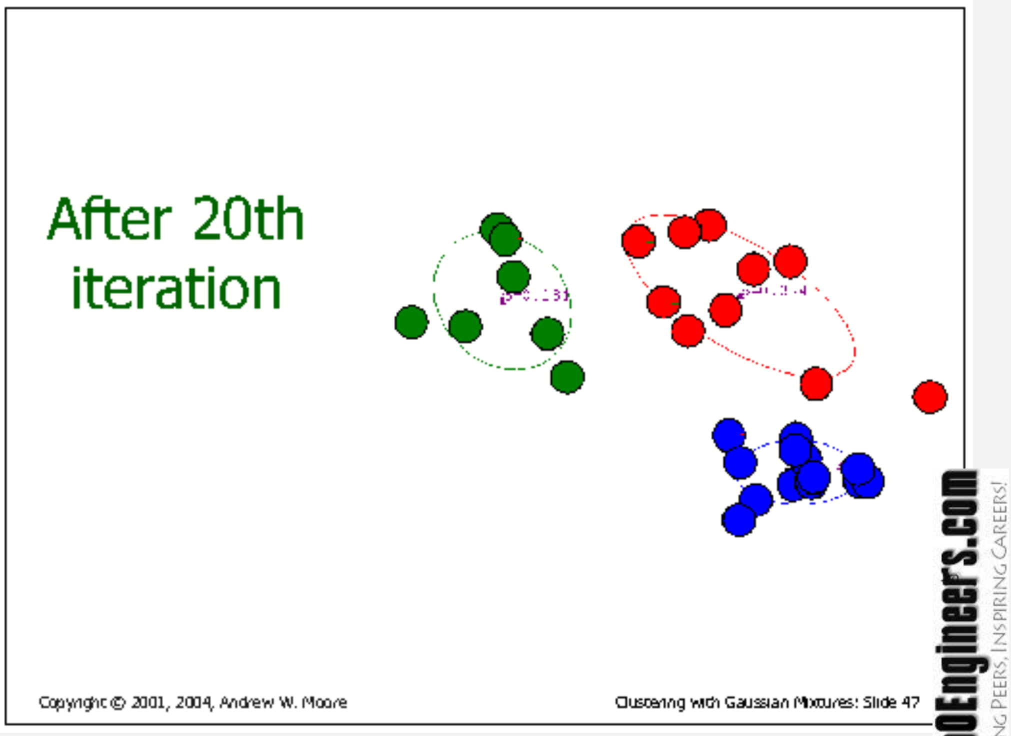
Clustering with Gaussian Mixtures: Slid



After 5th iteration

Copyright © 2001, 2004, Andrew W. Moore

Clustering with Gaussian Mixtures: Slide 45



After 20th iteration

Copyright © 2001, 2004, Andrew W. Moore

Clustering with Gaussian Mixtures: Slide 47

Live demonstration: http://www.neurosci.aist.go.jp/˜ akaho/MixtureEM.html

Practical Issues

- E.M. can get stuck in local minima.
- EM is very sensitive to initial conditions; a good initial guess helps; k-means algorithm is used prior to application of EM algorithm

Size of a GMM

Bayesian Information Criterion (BIC) value of a GMM can be defined as follows:

$$BIC(G \mid X) = logp(X \mid \hat{G}) - \frac{d}{2}logN$$

where

$\hat{G}$ represent the GMM with the ML parameter configuration

d represents the number of parameters in G

N is the size of the dataset

the first term is the log-likelihood term;

the second term is the model complexity penalty term.

BIC selects the best GMM corresponding to the largest BIC value by trading off these two terms.

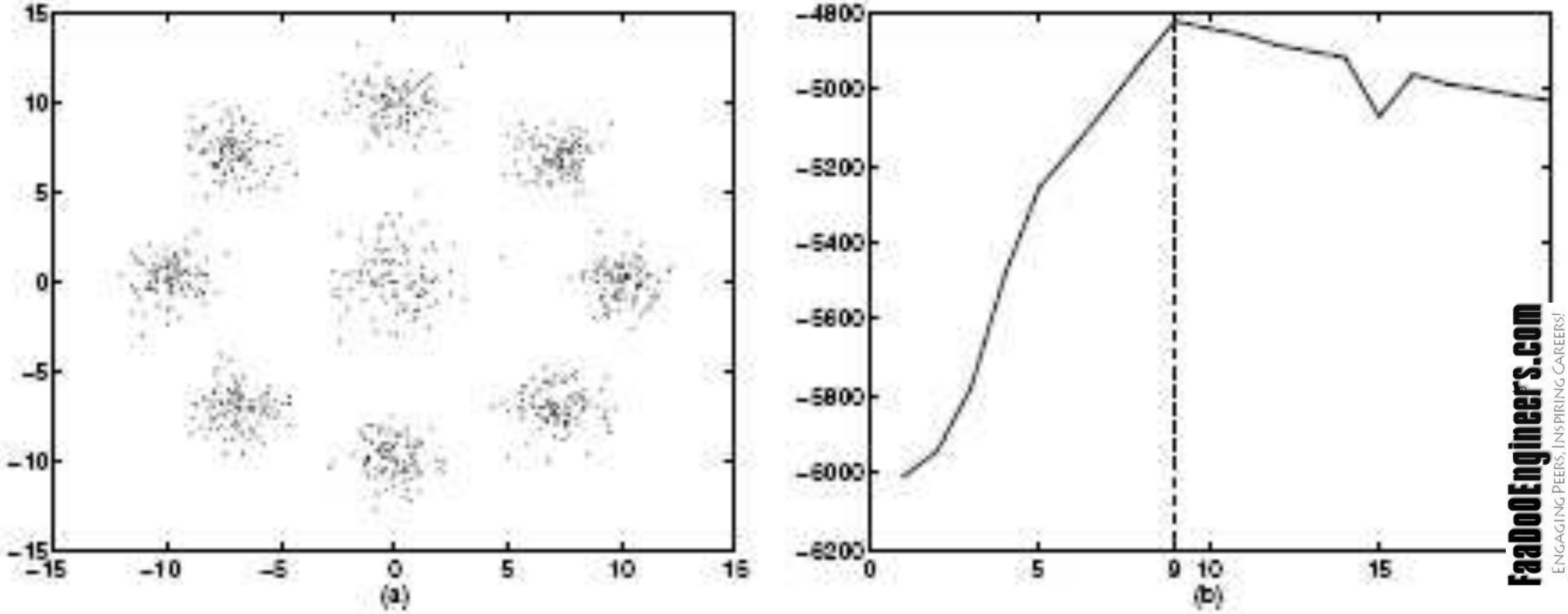


Fig. 1. Data set and the corresponding BIC value curve

source: *Boosting GMM and Its Two Applications*, F.Wang, C.Zhang and N.Lu in N.C.Oza et al. (Eds.) LNCS 3541, pp. 12-21, 2005

The BIC criterion can discover the true GMM size effectively as shown in the figure.

Maximum A Posteriori (MAP)

- Sometimes, it is difficult to get sufficient number of examples for robust estimation of parameters.

- However, one may have access to large number of similar examples which can be utilized.

- Adapt the target distribution from such a distribution. For example, adapt a speaker independent model to a new speaker using small amount of adaptation data.

MAP Adaptation

ML Estimation

$$\widehat{\theta}_{MLE} = \arg \max_{\theta} p(X|\theta)$$

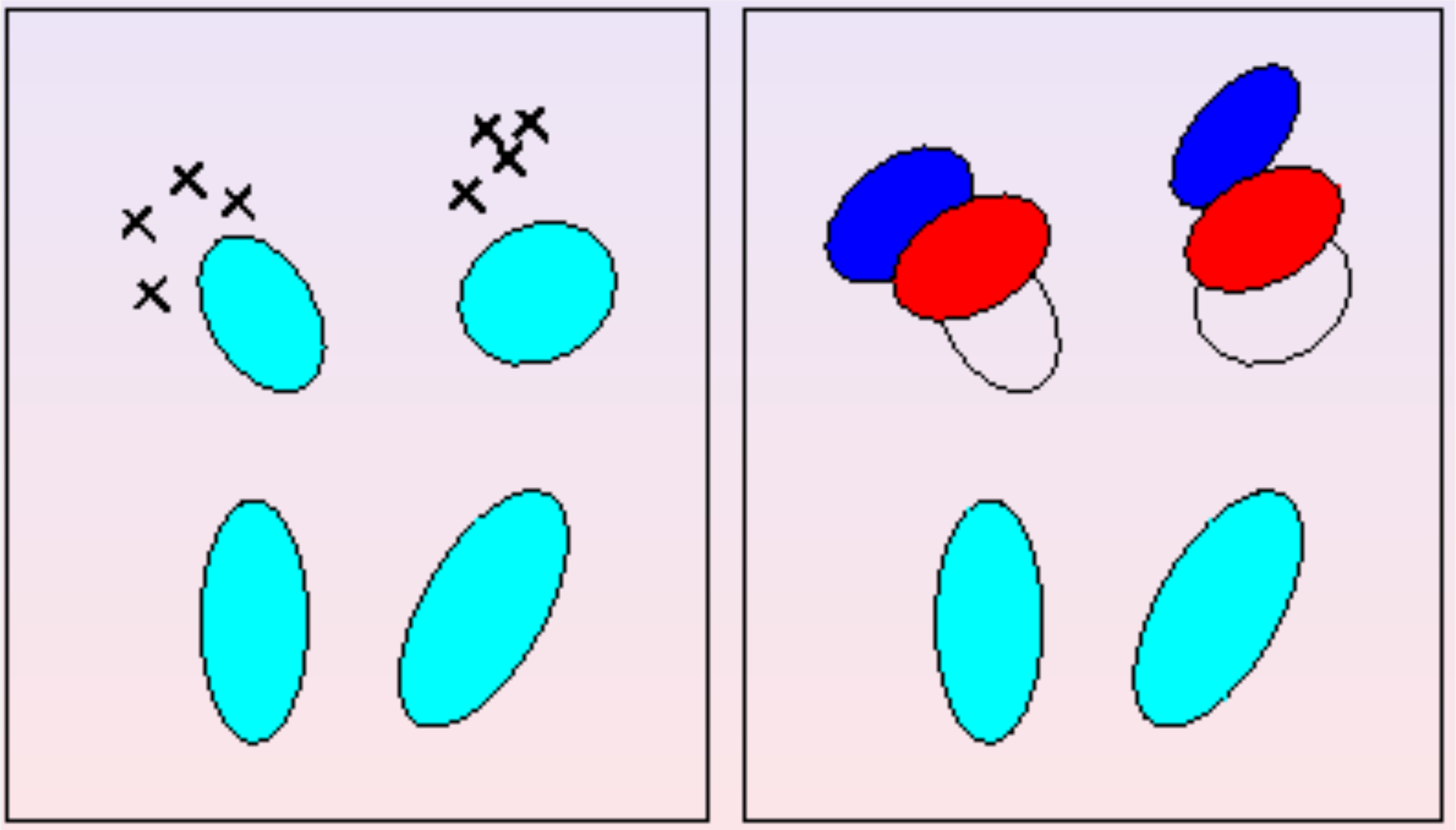
MAP Estimation

$$\begin{aligned}\widehat{\theta}_{MAP} &= \arg \max_{\theta} p(\theta|X) \\ &= \arg \max_{\theta} \frac{p(X|\theta)\ p(\theta)}{p(X)} \\ &= \arg \max_{\theta} p(X|\theta)\ p(\theta)\end{aligned}$$

$p(\theta)$ is the $a\ priori$ distribution of parameters θ.

A conjugate prior is chosen such that the corresponding posterior belongs to the same functional family as the prior.

Simple Implementation of MAP-GMMs



Source: Statistical Machine Learning from Data: GMM; Samy Bengio

Simple Implementation

Train a prior model p with a large amount of available data (say, from multiple speakers). Adapt the parameters to a new speaker using some adaptation data (**X**).

Let $\alpha = [0, 1]$ be a parameter that describes the faith on the prior model.

Adapted weight of j^{th} mixture

$$\hat{w}_j = \left[\alpha w_j^p + (1 - \alpha) \sum_i p(j|x_i) \right] \gamma$$

Here γ is a normalization factor such that $\sum w_j = 1$.

Simple Implementation (contd.)

means

$$\hat{\mu}_j = \alpha \mu_j^p + (1-\alpha) \frac{\sum_i p(j|x_i)\ x_i}{\sum_i p(j|x_i)}$$

Weighted average of sample mean and *a prior* mean

variances

$$\hat{\sigma}_j = \alpha \left(\sigma_j^p + \mu_j^p {\mu_j^p}' \right) + (1-\alpha) \frac{\sum_i p(j|x_i)\ x_i x_i'}{\sum_i p(j|x_i)} - \hat{\mu}_j {\hat{\mu}_j}'$$

HMM

- Primary role of speech signal is to carry a message; sequence of sounds (phonemes) encode a sequence of words.
- The acoustic manifestation of a phoneme is mostly determined by:
  - Configuration of articulators (jaw, tongue, lip)
  - physiology and emotional state of speaker
  - Phonetic context
- HMM models sequential patterns; speech is a sequential pattern
- Most text dependent speaker recognition systems use HMMs
- Text verification involves verification/recognition of phonemes

Phoneme recognition

Consider two phonemes classes /aa/ and /iy/.

Problem: Determine to which class a given sound belongs.

Processing of speech signal results in a sequence of feature (observation) vectors: $\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T$ (say MFCC vectors)

We say the speech is /aa/ if: $p(aa|\boldsymbol{O}) \; > \; p(iy|\boldsymbol{O})$

Using Bayes Rule

$$\frac{\overbrace{p(\boldsymbol{O}|aa)}^{AcousticModel} \;\; p(aa)}{p(\boldsymbol{O})} \;\; Vs. \;\; \frac{p(\boldsymbol{O}|iy) \;\; \overbrace{p(iy)}^{PriorProb}}{p(\boldsymbol{O})}$$

Given $p(\boldsymbol{O}|aa)$, $p(aa)$, $p(\boldsymbol{O}|iy)$ and $p(iy)$ $\Rightarrow$ which is more probable ?

Parameter Estimation of Acoustic Model

How do we find the density function $p_{aa}(.)$ and $p_{iy}(.)$.

We assume a parametric model: $\Rightarrow$ $p_{aa}()$ parameterised by $\boldsymbol{\theta_{aa}}$
$\Rightarrow$ $p_{ij}()$ parameterised by $\boldsymbol{\theta_{iy}}$

Training Phase: Collect many examples of /aa/ being said
$\Rightarrow$ Compute corresponding observations $\boldsymbol{o_1, \ldots, o_{T_{aa}}}$

Use the *Maximum Likelihood Principle*

$$\widehat{\boldsymbol{\theta_{aa}}} = \arg \max_{\boldsymbol{\theta_{aa}}} p(\boldsymbol{O}; \boldsymbol{\theta}_{aa})$$

Recall: if the *pdf* is modelled as a Gaussian Mixture Model
. $\Rightarrow$ then we use EM Algorithm

Modelling of Phoneme

To enunciate /aa/ in a word ⇒ Our Articulators are moving from a configuration for previous phoneme to /aa/ and then proceeding to move to configuration of next phoneme.

Can think of 3 distinct time periods:

⇒ Transition from previous phoneme

⇒ Steady state

⇒ Transition to next phoneme

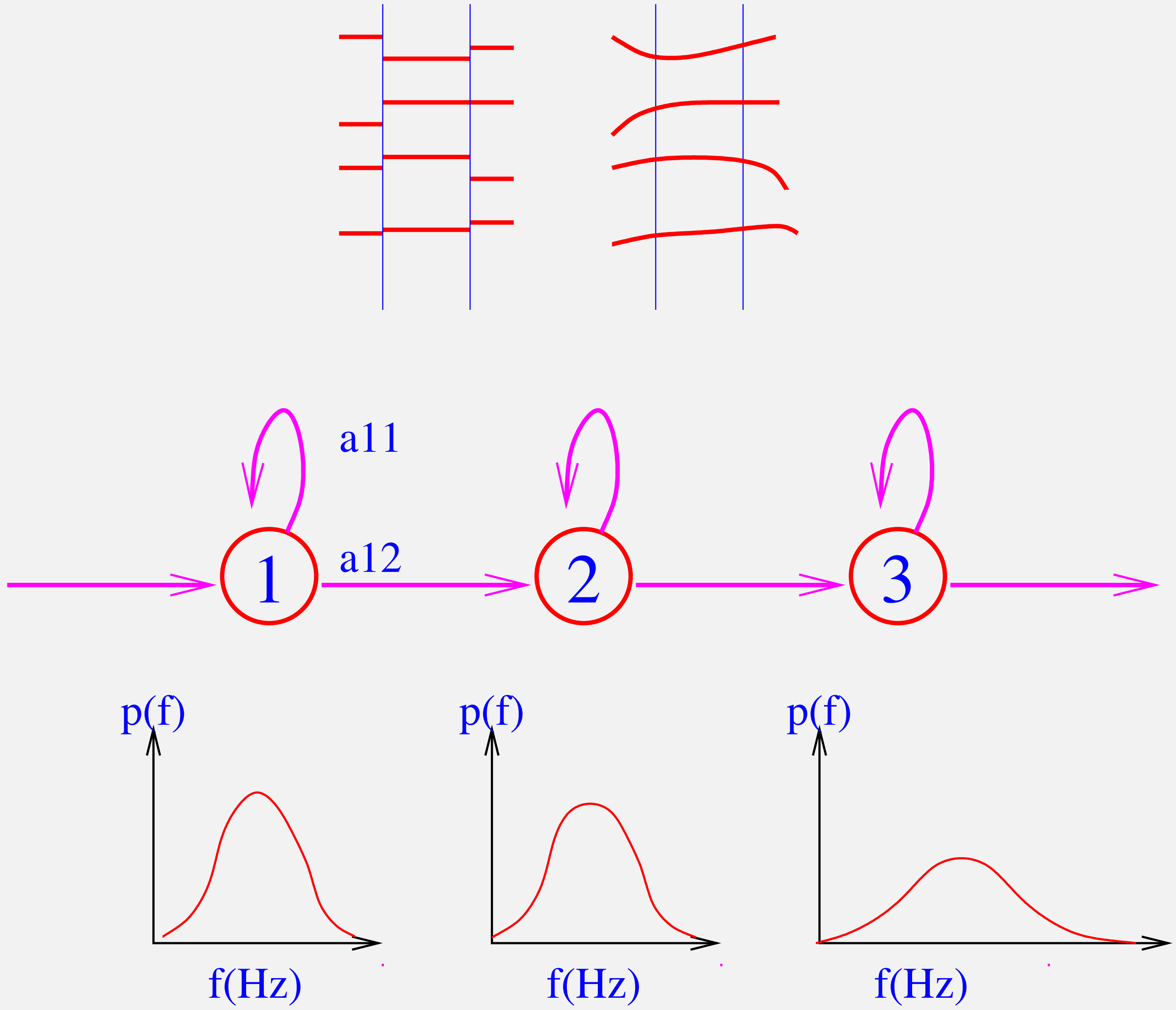
Features for 3 “time-interval ”are quite different

⇒ Use different density functions to model the three time intervals

⇒ model as $p_{aa^1}(;\boldsymbol{\theta_{aa}}^1)$ $p_{aa^2}(;\boldsymbol{\theta_{aa}}^2)$ $p_{aa^3}(;\boldsymbol{\theta_{aa}}^3)$

Also need to model the *time durations* of these time-intervals – transition probs.

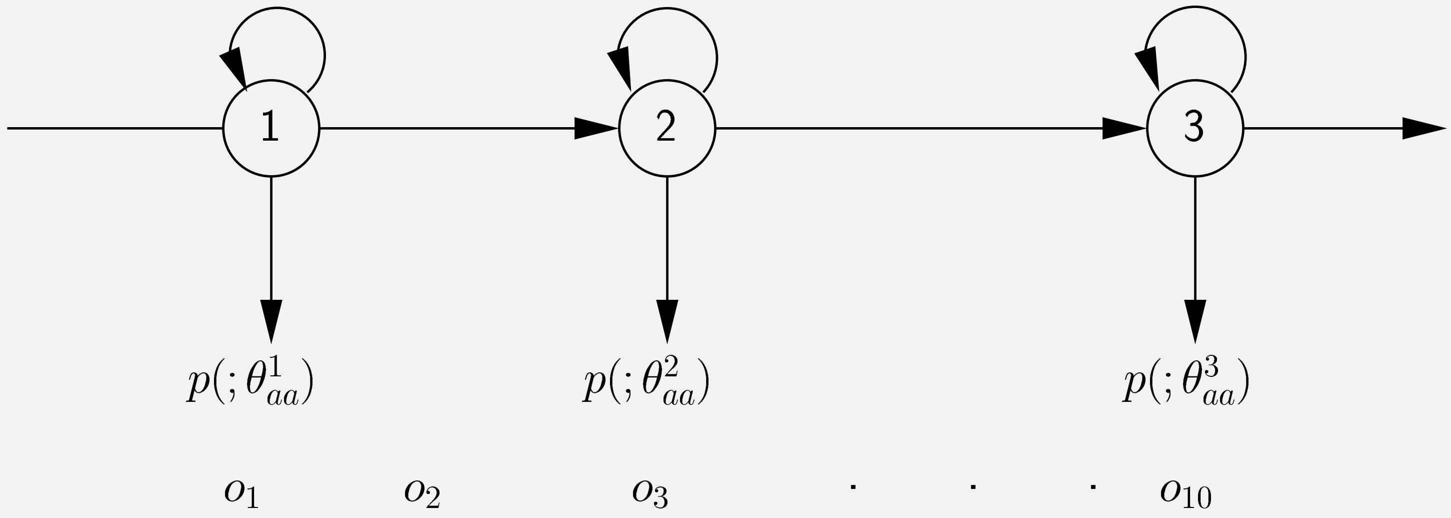
Stochastic Model (HMM)



HMM Model of Phoneme

- Use term “State” for each of the three time periods.

- Prob. of $\boldsymbol{o_t}$ from j^{th} state, i.e. $p_{aa^j}(\boldsymbol{o_t}; \boldsymbol{\theta_{aa}}^j) \Rightarrow$ denoted as $b_j(\boldsymbol{o_t})$



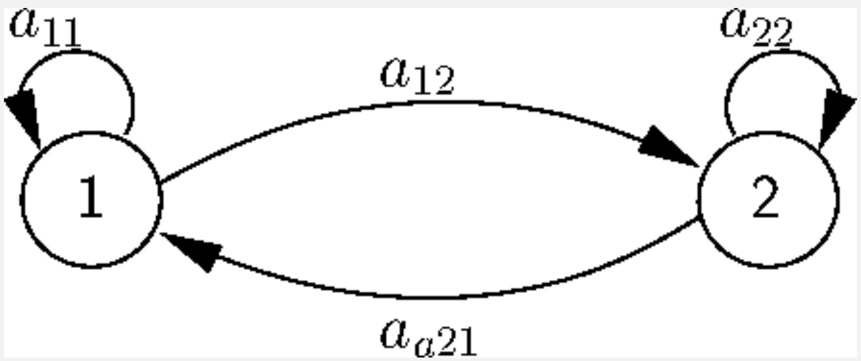
- Observation, $\boldsymbol{o_t}$, is generated by which state density?
  - Only observations are seen, the state-sequence is “hidden”
  - Recall: In GMM, the “mixture component is “hidden”

Probability of Observation

Recall: To classify, we evaluate $Pr(\boldsymbol{O}|\Lambda)$ – where Λ are parameters of models

In /aa/ Vs /iy/ calculation: $\Rightarrow Pr(\boldsymbol{O}|\Lambda_{aa})$ Vs $Pr(\boldsymbol{O}|\Lambda_{iy})$

Example: 2-state HMM model and 3 observations $\boldsymbol{o_1}\ \boldsymbol{o_2}\ \boldsymbol{o_3}$



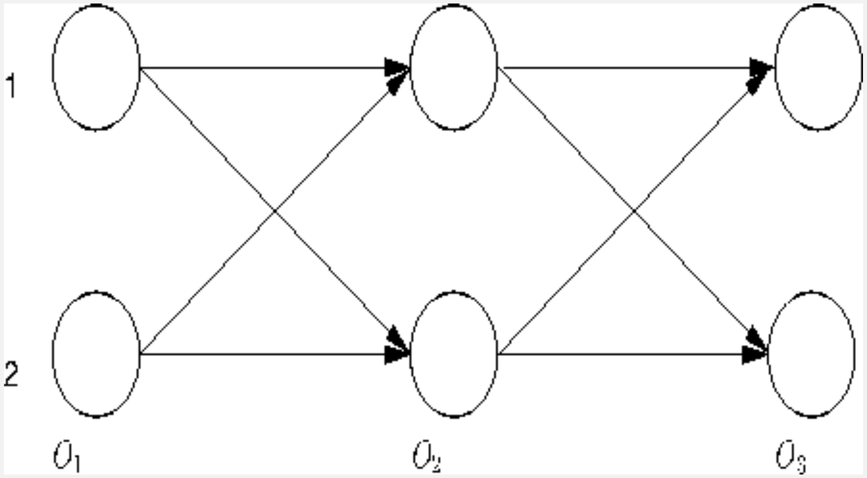
Model parameters are assumed known:

Transition Prob. $\Rightarrow a_{11},\ a_{12},\ a_{21},\ \text{and}\ a_{22}$ – model time durations

State density $\Rightarrow b_1(\boldsymbol{o_t})$ and $b_2(\boldsymbol{o_t})$.

$b_j(\boldsymbol{o_t})$ are usually modelled as single Gaussian with parameter $\mu_j,\ \sigma_j^2$ or by GMMs

Probability of Observation through one Path



$T = 3$ observations and $N = 2$ nodes $\Rightarrow 8$ paths thru 2 nodes for 3 observations

Example: Path P_1 through states $1,\ 1,\ 1.$

$Pr\{\boldsymbol{O}|P_1, \Lambda\} = b_1(\boldsymbol{o_1}) \cdot b_1(\boldsymbol{o_2}) \cdot b_1(\boldsymbol{o_3})$

Prob. of Path $P_1 = Pr\{P_1|\Lambda\} = a_{01} \cdot a_{11} \cdot a_{11}$

$$Pr\{\boldsymbol{O},\ P_1|\Lambda\} = Pr\{\boldsymbol{O}|P_1, \Lambda\} \cdot Pr\{P_1|\Lambda\} = a_{01}b_1(\boldsymbol{o_1}).a_{11}b_1(\boldsymbol{o_2}).a_{11}b_1(\boldsymbol{o_3})$$

Probability of Observation

Path	$\boldsymbol{o_1}$	$\boldsymbol{o_2}$	$\boldsymbol{o_3}$	$p(O, P_i \mid \Lambda)$
P_1	1	1	1	$a_{01}b_1(\boldsymbol{o}_1).a_{11}b_1(\boldsymbol{o}_2).a_{11}b_1(\boldsymbol{o}_3)$
P_2	1	1	2	$a_{01}b_1(\boldsymbol{o}_1).a_{11}b_1(\boldsymbol{o}_2).a_{12}b_2(\boldsymbol{o}_3)$
P_3	1	2	1	$a_{01}b_1(\boldsymbol{o}_1).a_{12}b_2(\boldsymbol{o}_2).a_{21}b_1(\boldsymbol{o}_3)$
P_4	1	2	2	$a_{01}b_1(\boldsymbol{o}_1).a_{12}b_2(\boldsymbol{o}_2).a_{22}b_2(\boldsymbol{o}_3)$
P_5	2	1	1	$a_{02}b_2(\boldsymbol{o}_1).a_{21}b_1(\boldsymbol{o}_2).a_{11}b_1(\boldsymbol{o}_3)$
P_6	1	1	2	$a_{02}b_2(\boldsymbol{o}_1).a_{21}b_1(\boldsymbol{o}_2).a_{12}b_2(\boldsymbol{o}_3)$
P_7	1	1	1	$a_{02}b_2(\boldsymbol{o}_1).a_{22}b_1(\boldsymbol{o}_2).a_{21}b_1(\boldsymbol{o}_3)$
P_8	1	1	2	$a_{02}b_2(\boldsymbol{o}_1).a_{22}b_1(\boldsymbol{o}_2).a_{22}b_2(\boldsymbol{o}_3)$

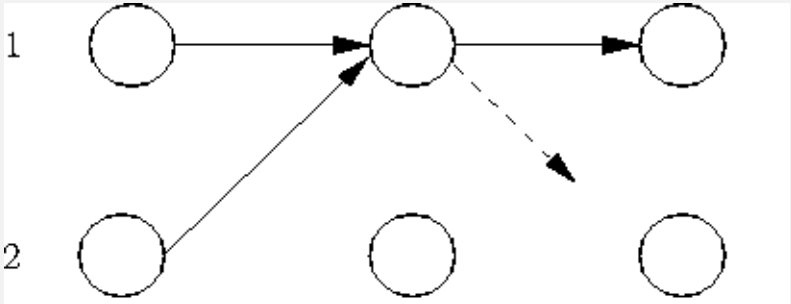
$$p(\boldsymbol{O}|\Lambda) = \sum_{P_i} P\{\boldsymbol{O}, P_i|\Lambda\} = \sum_{P_i} P\{\boldsymbol{O}|P_i, \Lambda\} \cdot P\{P_i, \Lambda\}$$

Forward Algorithm ⇒ Avoid Repeat Calculations:

$$\overbrace{a_{01}b_1(\boldsymbol{o_1})a_{11}b_1(\boldsymbol{o}_2).a_{11}b_1(\boldsymbol{o}_3) + a_{02}b_2(\boldsymbol{o}_1).a_{21}b_1(\boldsymbol{o}_2).a_{11}b_1(\boldsymbol{o}_3)}^{\text{Two Multiplications}}$$

$$= \underbrace{[a_{01}.b_1(\boldsymbol{o}_1)a_{11}b_1(\boldsymbol{o}_2) + a_{02}b_2(\boldsymbol{o}_1)a_{21}b_1(\boldsymbol{o}_2)]a_{11}b_1(\boldsymbol{o}_3)}_{\text{One Multiplication}}$$

Forward Algorithm – Recursion



$$[a_{01}b_1(o_1)a_{11}b_1(o_2) + a_{02}b_2(o_1)a_{21}b_1(o_2)].a_{11}b_1(o_3)$$

$$[a_{01}b_1(o_1)a_{11}b_1(o_2) + a_{02}b_2(o_1)a_{21}b_1(o_2)].a_{12}b_2(o_3)$$

$$\begin{aligned}
\text{Let :}\alpha_1(t=1) &= a_{01}b_1(\boldsymbol{o_1}) \\
\text{Let :}\alpha_2(t=1) &= a_{02}b_2(\boldsymbol{o_2})
\end{aligned}$$

$$\begin{aligned}
\text{Recursion : } \alpha_1(t=2) &= [a_{01}b_1(\boldsymbol{o_1}).a_{11} + a_{02}b_2(\boldsymbol{o_1}).a_{21}].b_1(\boldsymbol{o_2}) \\
&= [\alpha_1(t=1).a_{11} + \alpha_2(t=1).a_{21}].b_1(\boldsymbol{o_2})
\end{aligned}$$

General Recursion in Forward Algorithm

$$\begin{aligned}\alpha_j(t) &= \left[\sum \alpha_i(t-1)a_{ij}\right].b_j(\boldsymbol{o_t}) \\ &= P\{\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots \boldsymbol{o}_t, s_t = j|\Lambda\}\end{aligned}$$

Note

$\alpha_j(t) \Rightarrow$ Sum of probabilities of all paths ending at node *j* at time *t* with partial observation sequence $\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_t$

The probability of the entire observation $(\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T)$, therefore, is

$$\begin{aligned}p(\mathbf{0}|\Lambda) &= \sum_{j=1}^{N} P\{\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T, S_T = j|\Lambda\} \\ &= \sum_{j=1}^{N} \alpha_j(T)\end{aligned}$$

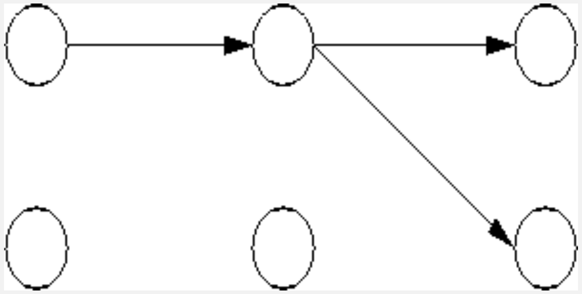
where N=No. of nodes

Backward Algorithm

- analogous to Forward, but coming from the last time instant T

Example: $a_{01}b_1(\boldsymbol{o_1}).a_{11}b_1(\boldsymbol{o_2}).a_{11}b_1(\boldsymbol{o_3}) + a_{01}b_1(\boldsymbol{o_1}).a_{11}b_1(\boldsymbol{o_2})a_{12}b_2(\boldsymbol{o_3}) + \ldots$

$$= [a_{01}.b_1(\boldsymbol{o_1}).a_{11}b_1(\boldsymbol{o_2})].(a_{11}b_1(\boldsymbol{o_3}) + a_{12}b_2(\boldsymbol{o_3}))$$



$$\begin{aligned}
\beta_1(t=2) &= p\{\boldsymbol{o_3}|s_{t=2}=1,\Lambda\} \\
&= p\{\boldsymbol{o_3}, s_{t=3}=1|s_{t=2}=1;\Lambda\} + p\{\boldsymbol{o_3}, s_{t=3}=2|s_{t=2}=1,\Lambda\} \\
&= \begin{array}{l} p\{\boldsymbol{o_3}|s_{t=3} = 1, s_{t=2} = 1,\Lambda\}.p\{s_{t=3} = 1|s_{t=2} = 1,\Lambda\} + \\ p\{\boldsymbol{o_3}|s_{t=3}=2, s_{t=2}=1,\Lambda\}.p\{s_{t=3}=2|s_{t=2}=1,\Lambda\} \end{array} \\
&= b_1(\boldsymbol{o_3}).a_{11} + b_2(\boldsymbol{o_3}).a_{12}
\end{aligned}$$

General Recursion in Backward Algorithm

$\beta_j(t) \Rightarrow$ Given that we are at node j at time t
Sum of probabilities of all paths such that
partial sequence $\boldsymbol{o_{t+1}, \ldots, o_T}$ are observed

$$\begin{aligned}\beta_i(t) &= \underbrace{\sum_{j=1}^{N}[a_{ij}b_j(\boldsymbol{o_{t+1}})]}_{\text{Going to each node from } i^{th} \text{ node}} \qquad \underbrace{\beta_j(t+1)}_{\substack{\text{Prob. of observation } \boldsymbol{o_{t+2}\ldots o_T} \text{ given} \\ \text{now we are in } j^{th} \text{ node at } t+1}} \\ &= p\{\boldsymbol{o_{t+1}, \ldots, o_t}|s_t = i, \Lambda\}\end{aligned}$$

Estimation of Parameters of HMM Model

- Given known Model parameters, Λ:
  - Evaluated $p(\boldsymbol{O}|\Lambda) \Rightarrow$ useful for classification
  - Efficient Implementation: Use Forward or Backward Algo.
- Given set of observation vectors, $\boldsymbol{o_t}$ how do we estimate parameters of HMM?
  - Do not know which states $\boldsymbol{o_t}$ come from
    - ∗ Analogous to GMM – do not know which component
  - Use a special case of EM – Baum-Welch Algorithm
  - Use following relations from Forward/Backward

$$p(\boldsymbol{O}|\Lambda) = \alpha_N(T) = \beta_1(T) = \sum_{j=1}^{N} \alpha_j(t)\beta_j(t)$$

Parameter Estimation for Known State Sequence

Assume each state is modelled as a single Gaussian:

$\widehat{\mu}_j$ = Sample mean of observations assigned to state j.
$\widehat{\sigma}_j^2$ = Variance of the observations assigned to state j.

and

$$\text{Trans. Prob. from state } i \text{ to } j = \frac{\text{No. of times transition was made from } i \text{ to } j}{\text{Total number of times we made transition from } i}$$

In practice since we do not know which state generated the observation
$\Rightarrow$ So we will do probabilistic assignment.

Review of GMM Parameter Estimation

Do not know which component of the GMM generated output observation.

Given initial model parameters $\boldsymbol{\Lambda^g}$, and observation sequence $x_1, \ldots, x_T$.
Find probability x_i comes from component $j \Rightarrow$ Soft Assignment

$$p[component = 1 | x_i; \Lambda^g] = \frac{p[component = 1, x_i | \Lambda_g]}{p[x_i | \Lambda_g]}$$

So, re-estimation equations are:

$$\widehat{C}_j = \frac{1}{T} \sum_{i=1}^{T} p(comp = j | x_i; \Lambda_g)$$

$$\widehat{\mu}_j^{new} = \frac{\sum_{i=1}^{T} x_i p(comp = j | x_i; \Lambda^g)}{\sum_{i=1}^{T} p(comp = j | x_i; \Lambda^g)} \qquad \widehat{\sigma}_j^2 = \frac{\sum_{i=1}^{T} (x_i - \widehat{\mu}_j)^2 p(comp = j | x_i; \Lambda_g)}{\sum_{i=1}^{T} p(comp = j | x_i; \Lambda^g)}$$

A similar analogy holds for hidden Markov models

Baum-Welch Algorithm

Here: We do not know which observation $\boldsymbol{o_t}$ comes from which state s_i

Again like GMM we will assume initial guess parameter Λ^g

Then prob. of being in "state=i at time=t" and "state=j at time=t+1" is

$$\widehat{\tau}_t(i,j) \;=\; p\{q_t = i, q_{t+1} = j | \boldsymbol{O}, \Lambda^g\} \;=\; \frac{p\{q_t = i, q_{t+1} = j, \boldsymbol{O} | \Lambda^g\}}{p\{\boldsymbol{O}|\Lambda^g\}}$$

where $p\{\boldsymbol{O}|\Lambda^g\} = \alpha_N(T) = \sum_i \alpha_i(T)$

Baum-Welch Algorithm

Then prob. of being in "state=i at time=t" and "state=j at time=t+1" is

$$\widehat{\tau}_t(i,j) \;=\; p\{q_t = i, q_{t+1} = j | \boldsymbol{O}, \Lambda^g\} \;=\; \frac{p\{q_t = i, q_{t+1} = j, \boldsymbol{O} | \Lambda^g\}}{p\{\boldsymbol{O}|\Lambda^g\}}$$

where $p\{\boldsymbol{O}|\Lambda^g\} = \alpha_N(T) = \sum_i \alpha_i(T)$

From ideas of Forward-Backward Algorithm, numerator is

$$p\{q_t = i, q_{t+1} = j, \boldsymbol{O}|\Lambda^g\} \;=\; \alpha_i(t).a_{ij}b_j(\boldsymbol{o_{t+1}}).\beta_j(t+1)$$

$$\text{So } \widehat{\tau}_t(i,j) \;=\; \frac{\alpha_i(t).a_{ij}b_j(\boldsymbol{o_{t+1}})\beta_j(t+1)}{\alpha_N(t)}$$

Estimating Transition Probability

$$\text{Trans. Prob. from state } i \text{ to } j = \frac{\text{No. of times transition was made from } i \text{ to } j}{\text{Total number of times we made transition from } i}$$

$\widehat{\tau}_t(i,j) \Rightarrow$ prob. of being in "state=i at time=t" and "state=j at time=t+1"

If we average $\widehat{\tau}_t(i,j)$ over all time-instants, we get the number of times the system was in i^{th} state and made a transition to j^{th} state. So, a revised estimation of transition probability is

$$\widehat{a}_{ij}^{new} = \frac{\sum_{t=1}^{T-1} \tau_t(i,j)}{\sum_{t=1}^{T} (\underbrace{\sum_{j=1}^{N} \tau_t(i,j)}_{\substack{\text{all transitions out} \\ \text{of i at time=t}}})}$$

Estimating State-Density Parameters

Analogous to GMM: which observation belonged to which component,

New estimates for the state pdf parameters are (assuming single Gaussian)

$$\widehat{\mu}_i = \frac{\sum_{t=1}^{T} \gamma_i(t) \boldsymbol{o}_t}{\sum_{t=1}^{T} \gamma_i(t)}$$

$$\widehat{\sum}_i = \frac{\sum_{t=1}^{T} \gamma_i(t) (\underline{\boldsymbol{o_t}} - \widehat{\mu}_i)(\boldsymbol{o_t} - \widehat{\mu}_i)^T}{\sum_{t=1}^{T} \gamma_i(t)}$$

These are weighted averages ⇒ weighted by Prob. of being in state j at t

- Given observation ⇒ HMM model parameters estimated iteratively
- $p(\boldsymbol{O}|\Lambda)$ ⇒ evaluated efficiently by Forward/Backward algorithm

Viterbi Algorithm

Given the observation sequence,

- the goal is to find corresponding state-sequence that generated it
- there are many possible combination (N^T) of state sequence ⇒ many paths.

One possible criterion : Choose the state sequence corresponding to path that with maximum probability

$$\max_i P\{\boldsymbol{O}, P_i | \Lambda\}$$

Word : represented as sequence of phones

Phone : represented as sequence states

Optimal state-sequence ⇒ Optimal phone-sequence ⇒ Word sequence

Viterbi Algorithm and Forward Algorithm

Recall Forward Algorithm : We found probability over each path and summed over all possible paths

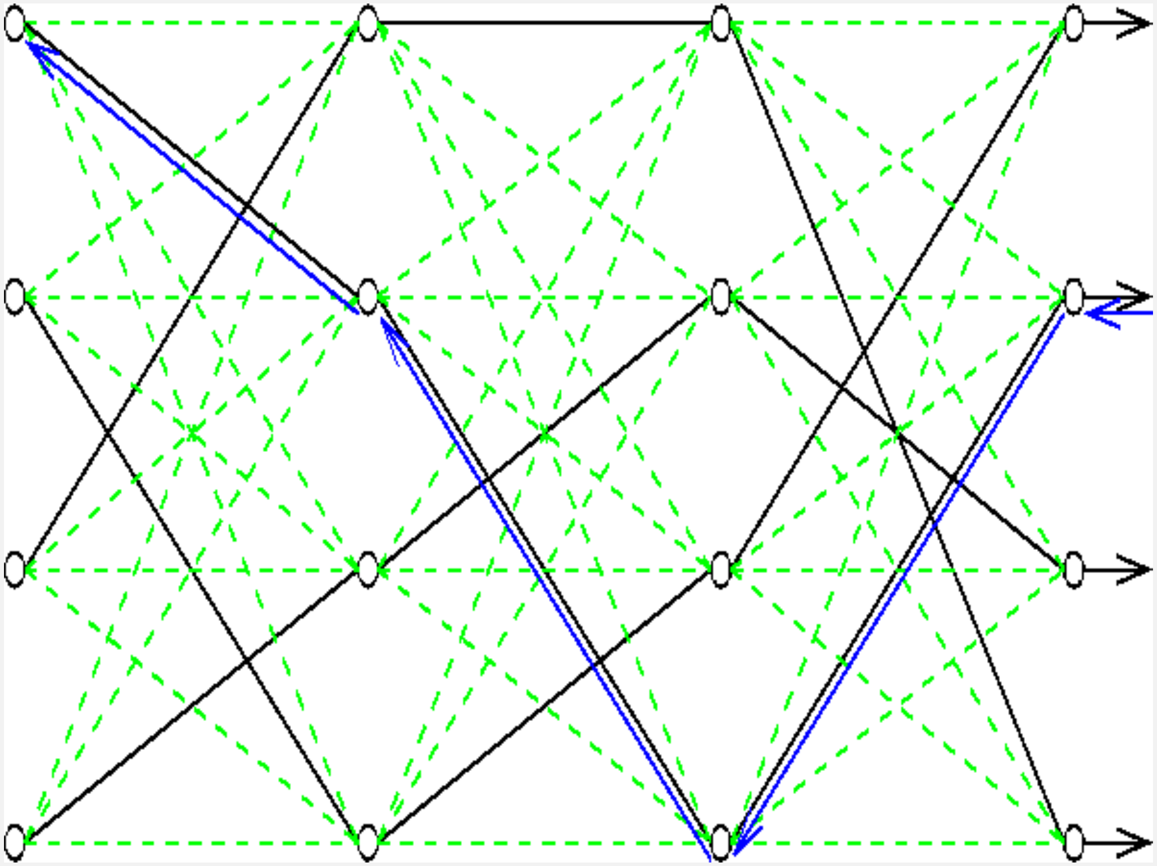
$$\sum_{i=1}^{N^T} p\{\boldsymbol{O}, P_i|\Lambda\}$$

Viterbi is just special case of Forward algo.

$$\text{At each node}\begin{cases}\text{instead of sum of prob. of all paths}\\ \text{choose path with max prob.}\end{cases}$$

In Practice: $p(\boldsymbol{O}|\Lambda)$ approximated by Viterbi (instead of Forward Algo.)

Viterbi Algorithm



Decoding

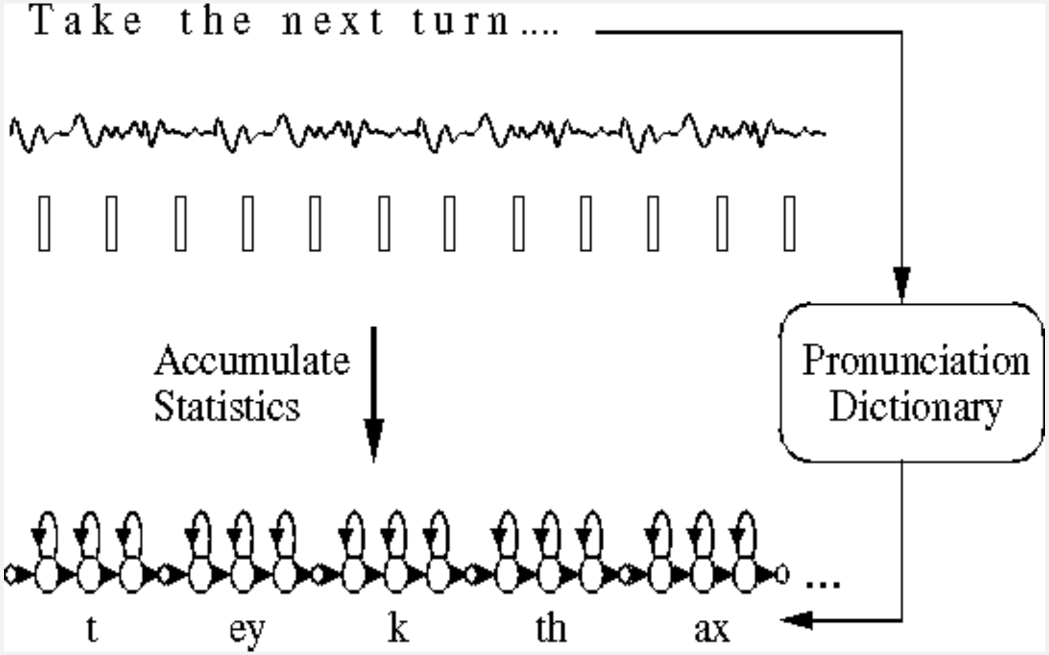
- Recall: Desired transcription $\widehat{\boldsymbol{W}}$ obtained by maximising

$$\widehat{\boldsymbol{W}} = \arg\max_{\boldsymbol{W}} p(\boldsymbol{W}|\boldsymbol{O})$$

- Search over all possible $\boldsymbol{W}$ – astronomically large!

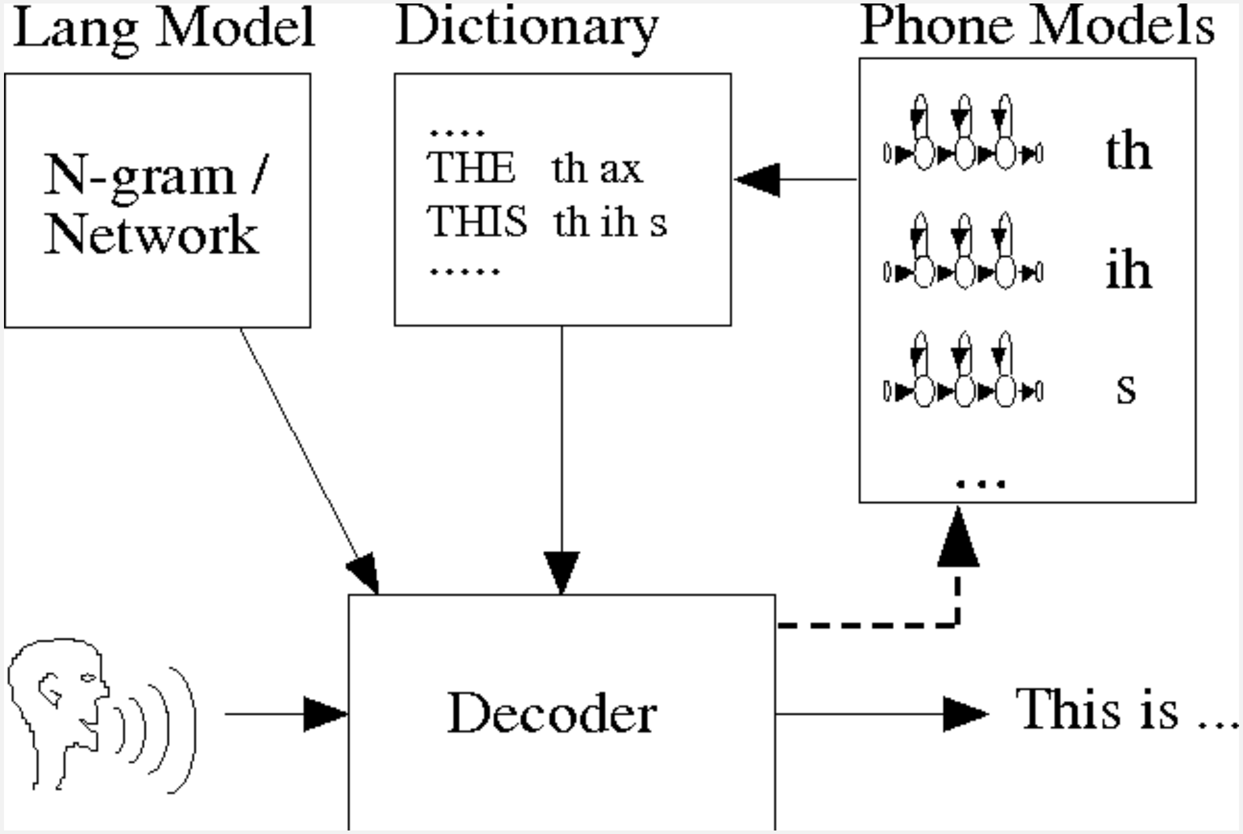
- Viterbi Search – find most likely path through a HMM
  - Sequence of phones (states) which is most probable
  - Mostly: most probable sequence of phones correspond to most probable sequence of words

Training of a Speech Recognition System



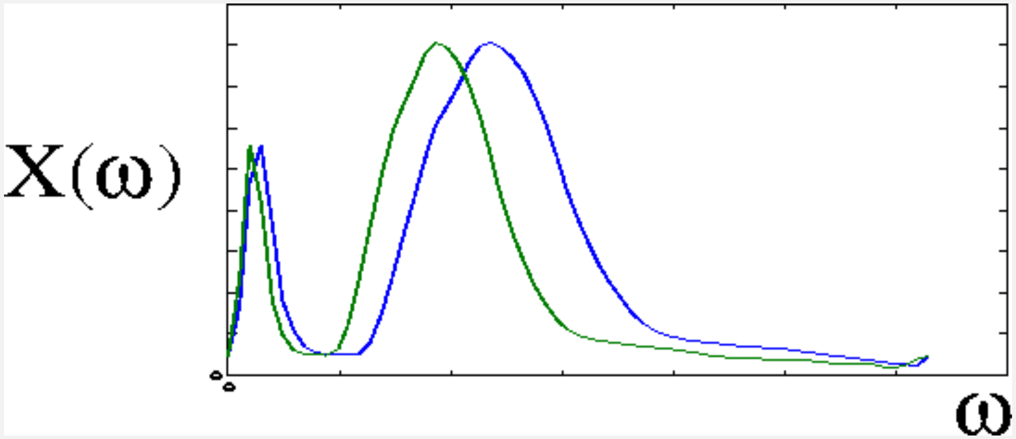
- HMM parameter's estimated using large databases – 100 hours
  - ∗ Parameters estimated using Maximum Likelihood Criterion

Recognition



Speaker Recognition

Spectra (formants) of a given sound are different for different speakers.



Spectra of 2 speakers for *one* "frame" of /iy/

Derive speaker dependent model of a new speaker by MAP adaptation of Speaker-Independent (SI) model using small amount of adaptation data; use for speaker recognition.

References

- *Pattern Classification*, R.O.Duda, P.E.Hart and D.G.Stork, John Wiley, 2001.
- *Introduction to Statistical Pattern Recognition*, K.Fukunaga, Academic Press, 1990.
- *The EM Algorithm and Extensions*, Geoffrey J. McLachlan and Thriyambakam Krishnan, Wiley-Interscience; 2nd edition, 2008. ISBN-10: 0471201707

- *The EM Algorithm and Extensions*, Geoffrey J. McLachlan and Thriyambakam Krishnan, Wiley-Interscience; 2nd edition, 2008. ISBN-10: 0471201707

- *Fundamentals of Speech Recognition*, Lawrence Rabiner & Biing-Hwang Juang, Englewood Cliffs NJ: PTR Prentice Hall (Signal Processing Series), c1993, ISBN 0-13-015157-2

- *Hidden Markov models for speech recognition*, X.D. Huang, Y. Ariki, M.A. Jack. Edinburgh: Edinburgh University Press, c1990.

- *Statistical methods for speech recognition*, F.Jelinek, The MIT Press, Cambridge, MA., 1998.

- *Maximum Likelihood from incomplete data via the em algorithm*, J. Royal Statistical Soc. **39**(1), pp. 1-38, 1977.

- Maximum a *Posteriori* Estimation for Multivariate Guassian Mixture Observation of Markov Chains, J.-L.Gauvain and C.-H.Lee, IEEE Trans. SAP, **2**(2), pp. 291-298, 1994.

- *A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models*, J.A.Bilmes, ISCI, TR-97-021.

- *Boosting GMM and Its Two Applications*, F.Wang, C.Zhang and N.Lu in N.C.Oza et al. (Eds.) LNCS 3541, pp. 12-21, 2005.