Automatic Speech Recognition (ASR)

Why is speech recognition difficult?

Intuitively...

- Meaning is represented by sentences
- A Sentence is a sequences of words
- A Word is a sequences of phonemes
- ...
- This view of speech is based on <u>text</u>
- But speech is NOT just “*Acoustic text*”

Speech is....

- Continuous
  - *“We were away a year ago”*
- Variable
  - *“bread and butter” or “brembudder”*
- Ambiguous
  - *“The grey tape can fix that leak”*
  - *“The great ape can fix that leek”*
  - *“The great ape can fix that league”*
  - *“The great tape can. Fix that’ll eek!”*

"League" or *"Leek"*?

- *"league"* = / l i g /
- *"leek"* = / l i k /
- Difference appears to be in the final consonant:
  - /g/ is *voiced*
  - /k/ is *unvoiced*
- But in natural fluent speech, the *duration* of the vowel /i/ may be a more important cue to recognition!

Approaches to Speech Recocognition

- Many approaches to speech recognition have been tried in the past, including:
  - Artificial Intelligence (AI)
  - Artificial Neural Networks
  - …
- The use of **Artificial Intelligence (AI)** based methods was widespread in the 1970s.
- Researchers believed there was insufficient information in the acoustic data to recognise speech, and that additional sources of **knowledge** were necessary.

Speech Recognition Approaches

- By late 1970s, AI-based systems had been outperformed simple pattern matching techniques.
- Most successful approach to-date is based on **statistical modelling**, and in particular **hidden Markov models (HMMs).**
- HMMs are basis of all state-of-the-art commercial (and most laboratory) speech recognition systems.

Speech Recognition Terminology

- Basic problem in speech recognition is **variability**.
- Early attempts to solve problem by **removing** it.
- **Speaker-dependent** speech recognition systems train on, and subsequently recognise, a **single speaker**
- **Multiple-speaker** systems work for a particular **population** of speakers
- **Speaker Independent** systems work for **any speaker**, with no implicit or explicit training.

Terminology (continued)

- **Speaker adaptive** systems automatically adapt to a new speaker. E.G: begin with a speaker-independent system, and then adapt the system to a particular speaker to obtain a speaker-dependent system.
- Another source of variability is co-articulation between words. **Isolated word** recognition systems require the user to leave **gaps** between words
- **Connected speech recognition** systems recognize isolated phrases or sentences.
- **Continuous speech recognition** systems recognize continuous speech.

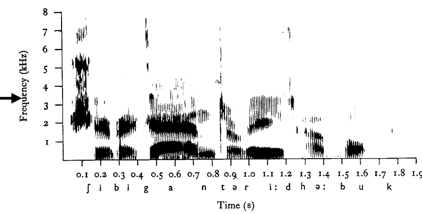
Vocabulary Size

- Another important issue is vocabulary size.
- **Small vocabulary** systems work with vocabularies of *10-100* words.
- **Medium vocabularies** comprise around *100* to *5,000* words.
- **Large Vocabulary Continuous Speech Recognition (LVCSR)** systems can cope with 60,000 words, while
- **Unlimited vocabulary** systems have no vocabulary size limitation.

Phoneme-HMM Speech Recognizer

Speech signal

Front-end signal processing



Phoneme HMM store

N-gram language model

Pronunciation dictionary

Application Compiler

Viterbi Decoder

Optimal word sequence

Or N-Best List,

Hidden Markov Models (HMM) Review

What is Covered

- Observable Markov Model
- Hidden Markov Model
- Evaluation problem
- Decoding Problem

Markov Models

- Set of states: $\{s_1, s_2, \ldots, s_N\}$
- Process moves from one state to another generating a sequence of states : $s_{i1}, s_{i2}, \ldots, s_{ik}, \ldots$
- Markov chain property: probability of each subsequent state depends only on what was the previous state:

$$P(s_{ik} \mid s_{i1}, s_{i2}, \ldots, s_{ik-1}) = P(s_{ik} \mid s_{ik-1})$$

- To define Markov model, the following probabilities have to be specified: transition probabilities $a_{ij} = P(s_i \mid s_j)$ and initial probabilities $\pi_i = P(s_i)$
- The output of the process is the set of states at each instant of time

Calculation of sequence probability

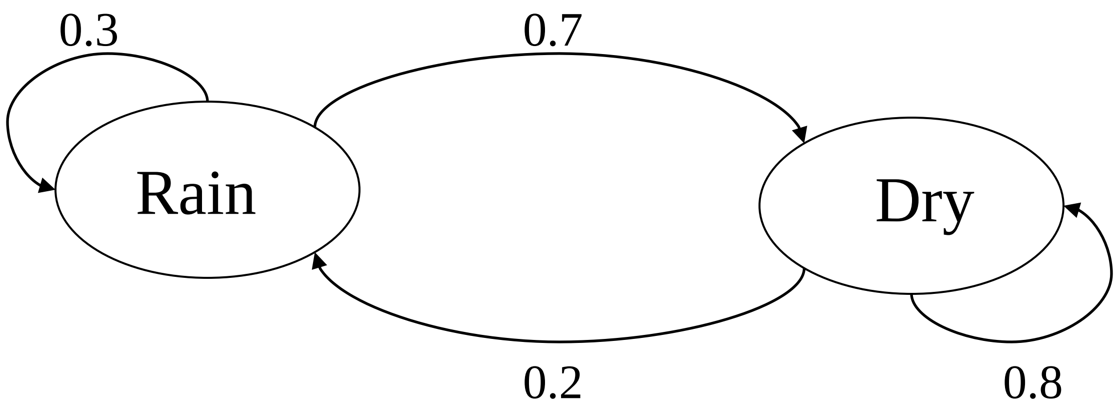
- By Markov chain property, probability of state sequence can be found by the formula:

$$P(s_{i1}, s_{i2}, \ldots, s_{ik}) = P(s_{ik} \mid s_{i1}, s_{i2}, \ldots, s_{ik-1}) P(s_{i1}, s_{i2}, \ldots, s_{ik-1})$$

$$= P(s_{ik} \mid s_{ik-1}) P(s_{i1}, s_{i2}, \ldots, s_{ik-1}) = \ldots$$

$$= P(s_{ik} \mid s_{ik-1}) P(s_{ik-1} \mid s_{ik-2}) \ldots P(s_{i2} \mid s_{i1}) P(s_{i1})$$

Example of Markov Model



- Two states : ‘Rain’ and ‘Dry’.
- Initial probabilities: say P(‘Rain’)=0.4 , P(‘Dry’)=0.6
- Suppose we want to calculate a probability of a sequence of states in our example, {‘Dry’,’Dry’,’Rain’,Rain’}.

P({‘Dry’,’Dry’,’Rain’,Rain’}) = ??.

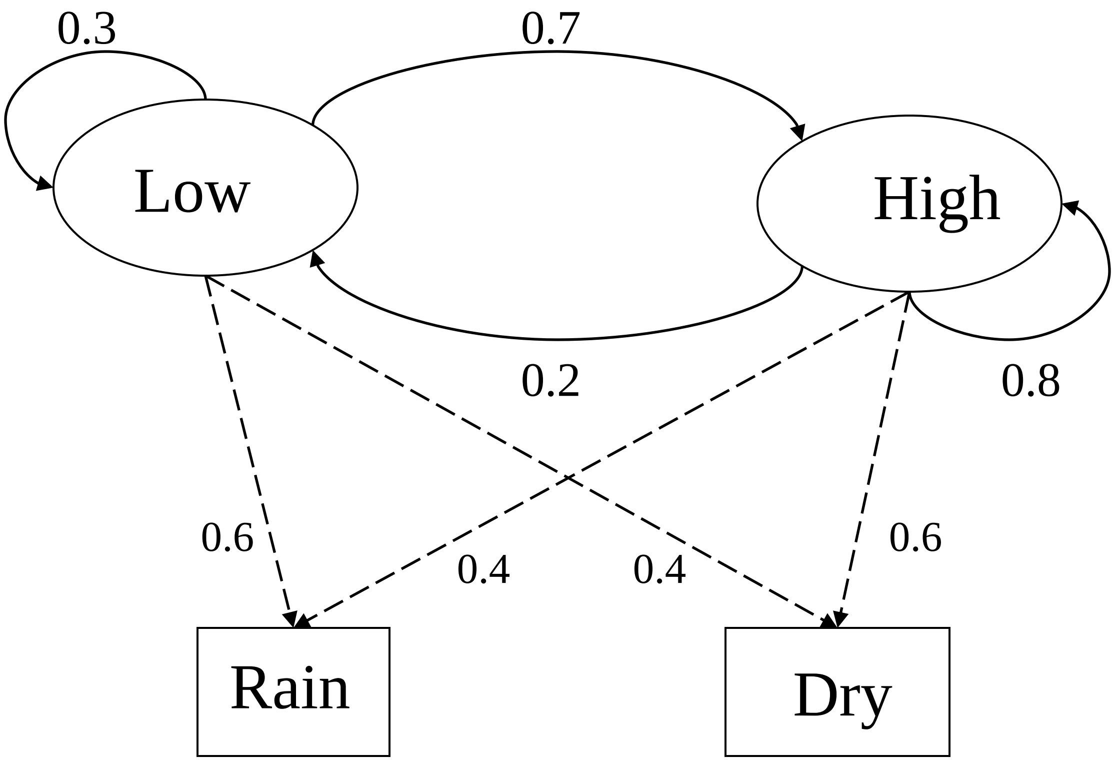
Hidden Markov models.

- The observation is turned to be a probabilistic function (discrete or continuous) of a state instead of an one-to-one correspondence of a state
- Each state randomly generates one of M observations (or visible states) $\{v_1, v_2, \ldots, v_M\}$
- To define hidden Markov model, the following probabilities have to be specified: matrix of transition probabilities $A=(a_{ij})$, $a_{ij}= P(s_i | s_j)$, matrix of observation probabilities $B=(b_i (v_m))$, $b_i(v_m) = P(v_m | s_i)$ and a vector of initial probabilities $\pi=(\pi_i)$, $\pi_i = P(s_i)$. Model is represented by $M=(A, B, \pi)$.

HMM Assumptions

- **Markov assumption**: the state transition depends only on the origin and destination
- **Output-independent assumption**: all observation frames are dependent on the state that generated them, not on neighbouring observation frames

Example of Hidden Markov Model



Example of Hidden Markov Model

- Two states : 'Low' and 'High' atmospheric pressure.
- Two observations : 'Rain' and 'Dry'.
- Transition probabilities: $P(\text{'Low'}|\text{'Low'})=0.3$, $P(\text{'High'}|\text{'Low'})=0.7$, $P(\text{'Low'}|\text{'High'})=0.2$, $P(\text{'High'}|\text{'High'})=0.8$
- Observation probabilities : $P(\text{'Rain'}|\text{'Low'})=0.6$, $P(\text{'Dry'}|\text{'Low'})=0.4$, $P(\text{'Rain'}|\text{'High'})=0.4$, $P(\text{'Dry'}|\text{'High'})=0.3$.
- Initial probabilities: say $P(\text{'Low'})=0.4$, $P(\text{'High'})=0.6$.

Calculation of observation sequence probability

•Suppose we want to calculate a probability of a sequence of observations in our example, {'Dry','Rain'}.

•Consider all possible hidden state sequences:

P({'Dry','Rain'}) = P({'Dry','Rain'} , {'Low','Low'}) + P({'Dry','Rain'} , {'Low','High'}) + P({'Dry','Rain'} , {'High','Low'}) + P({'Dry','Rain'} , {'High','High'})

where first term is :

P({'Dry','Rain'} , {'Low','Low'})=

P({'Dry','Rain'} | {'Low','Low'}) P({'Low','Low'}) = ??

Hidden Markov Models

A HMM consists of

- A set of states $S = \{s_1,...,s_N\}$
- A state transition probability matrix $A = [a_{ij}]_{i,j=1,...N}$,

  where a_{ij} =Prob(s_j at time t | s_i at time t-1)
- For each state s_i, a PDF b_i defined on the set of possible observations O s.t.

  $b_i(o)$ = Prob$(y_t=o \mid x_t=s_i)$
- b_i is called the **state output PDF** for state i (or the i^{th} **state output PDF**)

HMM Assumptions

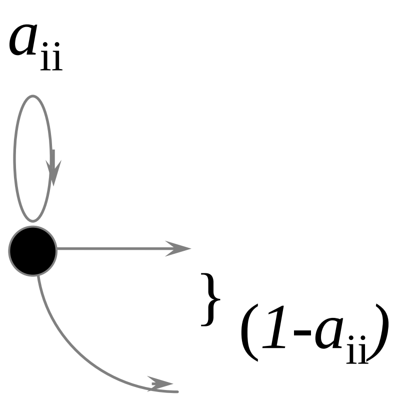
- **Temporal Independence -** the observation y_t depends on the state s_t but is otherwise independent of the rest of the observation sequence $O = \{o_t\}$!

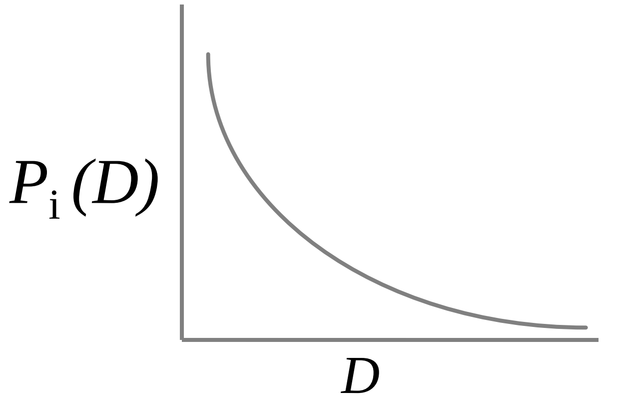
  ... so, the position of the vocal tract at time *t* is independent of its position at time *t-1*!

- **Piecewise stationarity -** the underlying structure of speech is a sequence of stationary segments
- **Random variability -** variations from this underlying structure are random

HMM State Duration Model

- Constant segments correspond to the HMM states





- Probability of state duration D is given by

  $P_i(D) = (1 - a_{ii})a_{ii}^{(D-1)}$

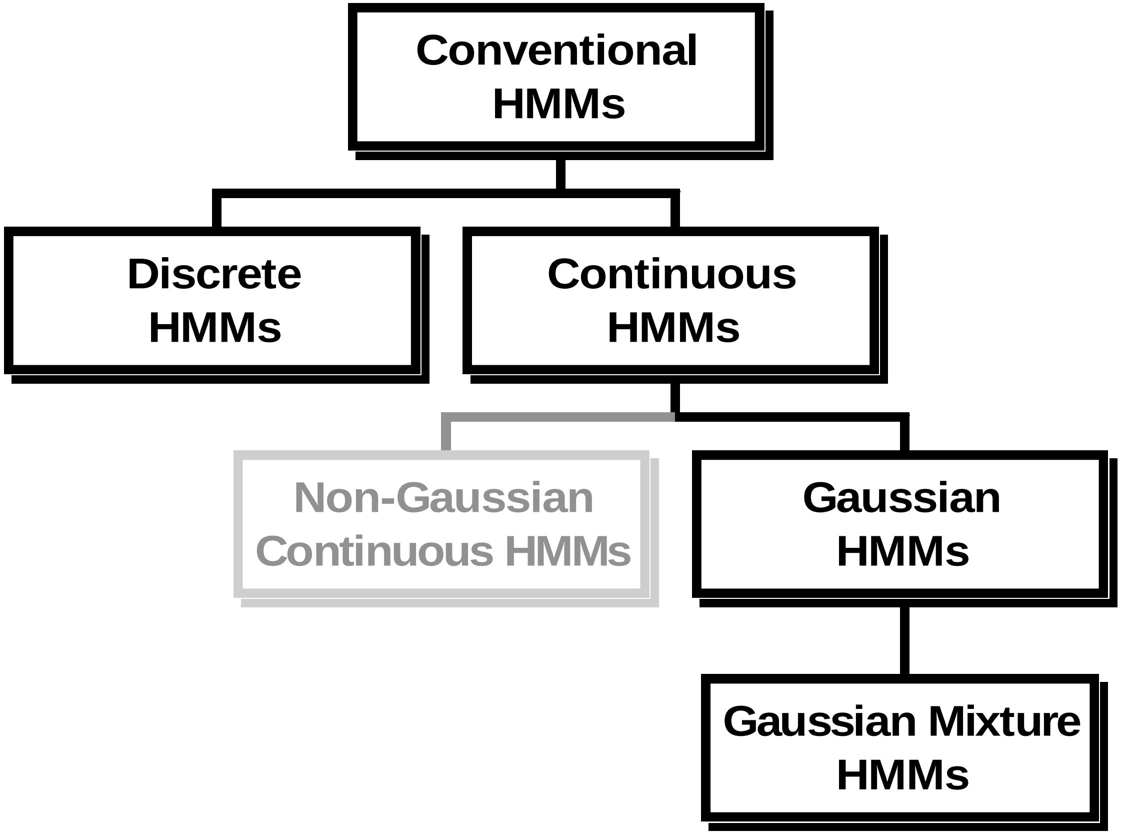
Main issues using HMMs :

Evaluation problem. Given the HMM $M=(A, B, \pi)$ and the observation sequence $O=o_1 o_2 \ldots o_K$, calculate the probability that model M has generated sequence O.

- **Decoding problem.** Given the HMM $M=(A, B, \pi)$ and the observation sequence $O=o_1 o_2 \ldots o_K$, calculate the most likely sequence of hidden states s_i that produced this observation sequence O.
- **Learning problem.** Given some training observation sequences $O=o_1 o_2 \ldots o_K$ and general structure of HMM (numbers of hidden and visible states), adjust $M=(A, B, \pi)$ to maximize the probability.

$O=o_1 \ldots o_K$ denotes a sequence of observations $o_k \in \{v_1, \ldots, v_M\}$.

Types of Conventional HMM

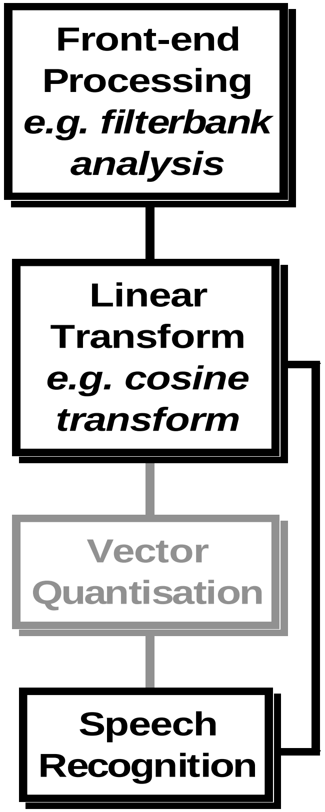


Front-End Processing Re-Visited

Vectors in d-dimensional (continuous) space

Vectors in d-dimensional (continuous) space

Symbols from a **finite** set



Discrete HMMs

- If VQ is used, then a state output PDF b_i is defined by a **list** of probabilities-

  $b_i(m) = \text{Prob}(y_t = z_m \mid x_t = s_i)$
- The resulting HMM is a **discrete HMM**
- Common in mid-1980/ early-1990s
- Computational advantages
- Disadvantages
  - VQ may introduce non-recoverable errors
  - Choice of metric d for VQ?
- Outperformed by Continuous HMM

Continuous HMMs

- Without VQ, $b_i(y)$ must be defined for any y in the (continuous) observation set S
- Hence discrete state output PDFs no longer viable
- Use parametric continuous state output PDFs - **Continuous HMMs**
- Choice of PDF restricted by mathematical tractability and computational usefulness (see “HMM training & recognition” later)
- Most people begin with Gaussian PDFs
- Resulting HMMs called **Gaussian HMMs**

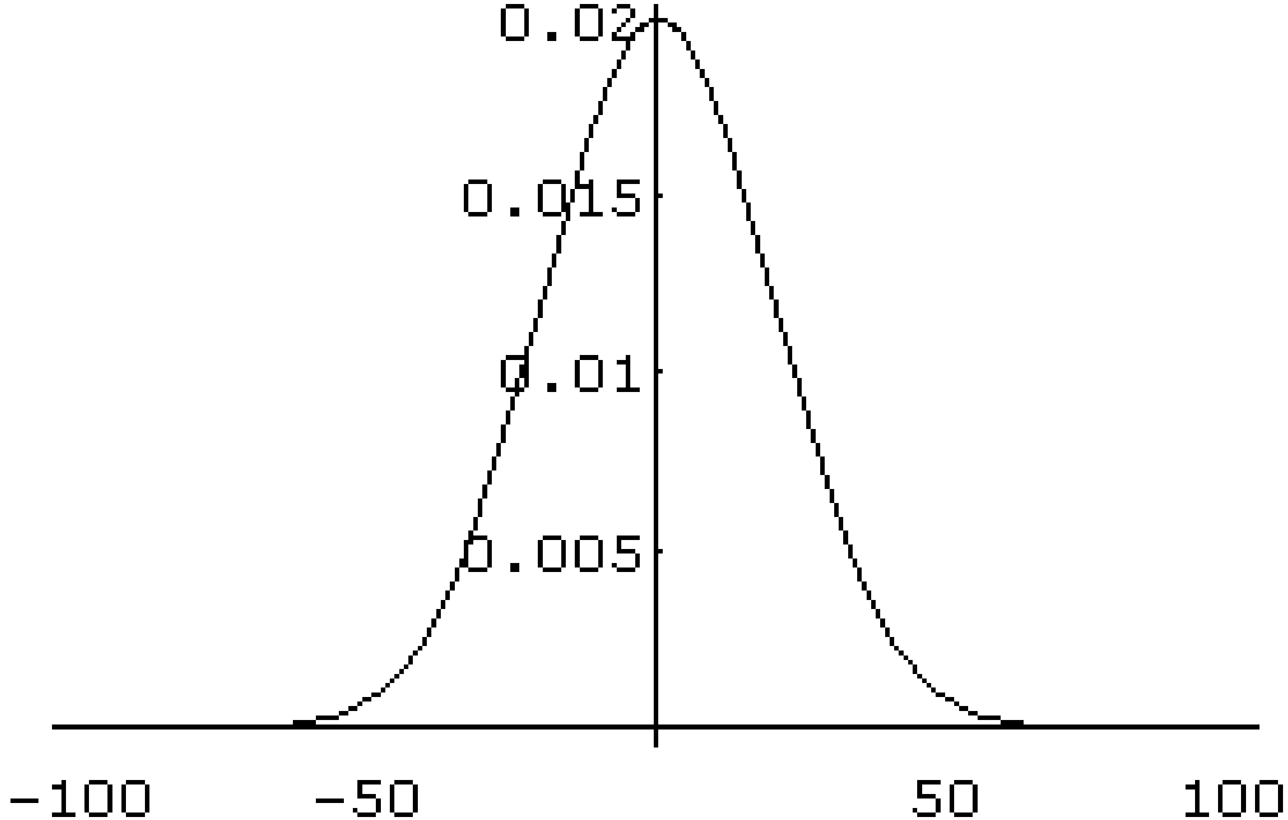
Gaussian HMMs

- State output PDFs are multivariate Gaussian

$$b_i(y) = \frac{1}{\sqrt{(2\pi)^d \left|\Sigma_i\right|}} \exp\left\{-\frac{1}{2}\left(y-\mu_i\right)' \Sigma_i^{-1}\left(y-\mu_i\right)\right\}$$

- μ_i and $\sum_\mathrm{i}$ are the mean vector and covariance matrix which define b_i

Gaussian PDF



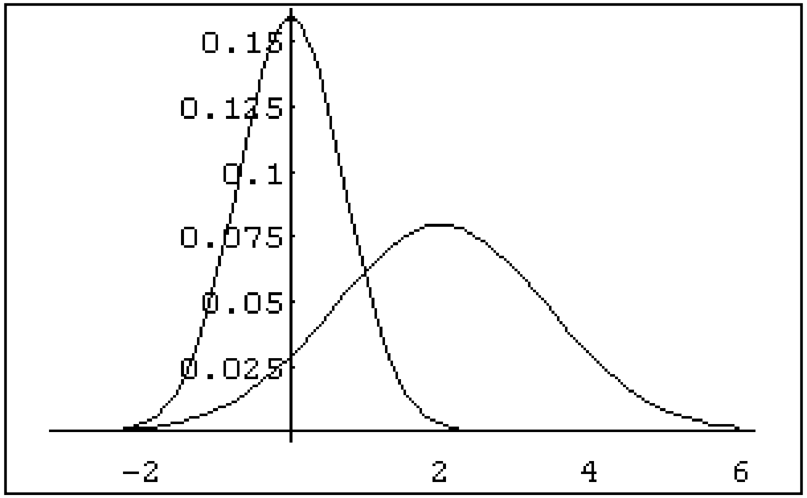
Gaussian PDF with $\mu_i = 0$ and $\Sigma_i = 20$

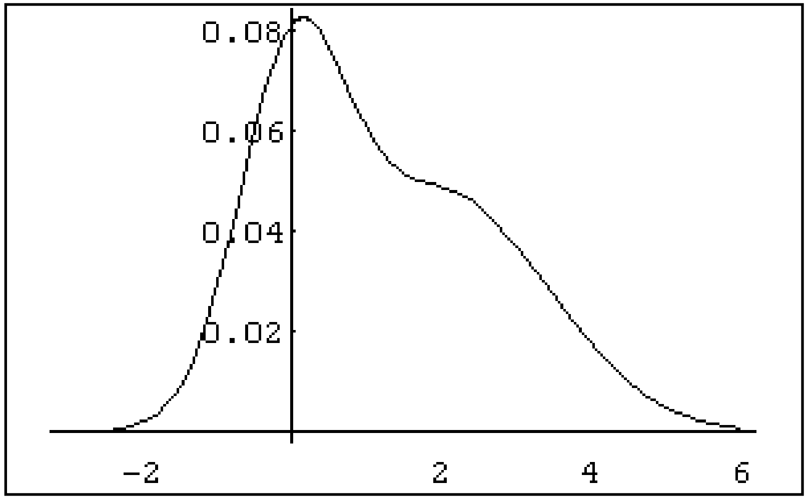
Gaussian HMMs - Issues

- Significant computational savings if covariance matrix can be assumed to be diagonal
- In general, Gaussian PDFs are **not** flexible enough to model speech pattern variability accurately
  - In many applications (e.g. modelling speech from multiple speakers) a unimodal PDF is inadequate
  - Even if unimodal PDF is basically OK there may be more subtle inadequacies

Gaussian Mixture Densities

Example - 2 component Gaussian mixture





$$f(y) = N_{(0,1)}(y)$$

$$g(y) = N_{(2,2)}(y)$$

$$m(y) = w.f(y) + (1-w).g(y)$$

Gaussian Mixture HMMs

- Any PDF can be approximated arbitrarily closely by a Gaussian mixture PDF with sufficient components
- But...
  - More mixture components require more data for robust model parameter estimation
  - Parameter smoothing and sharing needed (e.g. ‘tied mixtures’, ‘grand variance’,...)
- Gaussian mixture HMMs widely used in systems in research laboratories.

Relationship with Neural Networks

- ‘Classical’ HMM training methods focus on fitting state output PDFs to data (modelling), rather than minimizing overlap between PDFs (discrimination).
- NNs are good at discrimination
- **But** NNs poor at coping with time-varying data
- Research interest in ‘hybrid’ systems which use NNs to relate the observations to the states of the underlying Markov model.