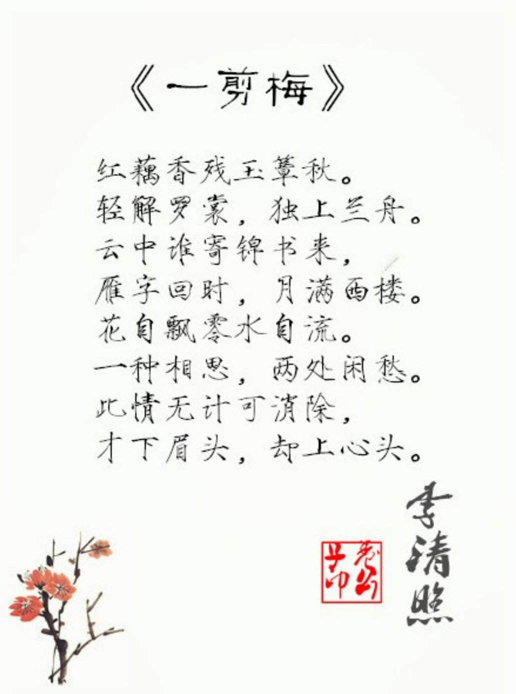
Introduction to Speech Data

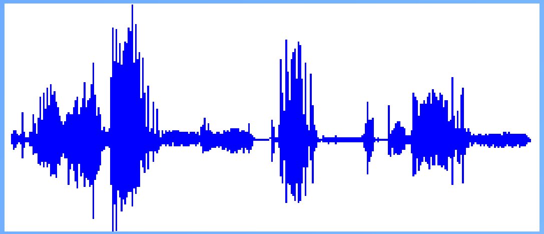
Ville Hautamäki and Tomi Kinnunen



Content

- Text-to-speech
- Speech-to-text
- 'High-level' speaker id

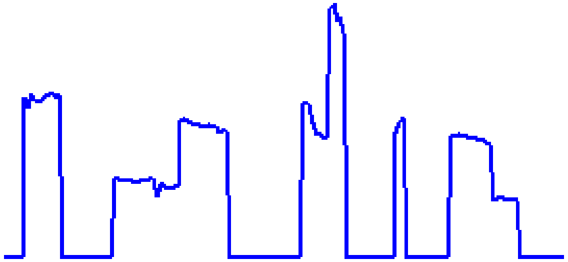
Speech



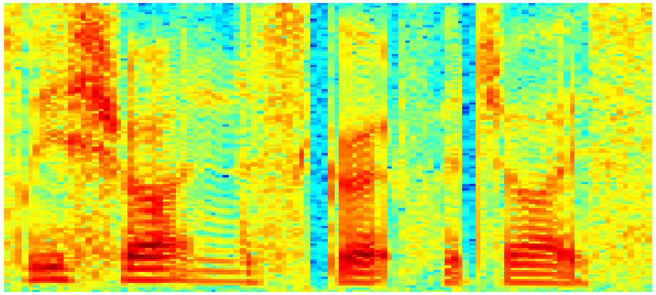
- Expressive TTS
- Singing synthesis
- Speaker verification

- Text-to-speech
- Voice conversion
- Automatic speaker verification

Prosody



Timbre



Slide adapted from prof. Haizhou Li's keynote at APSIPA'13

Speech production system

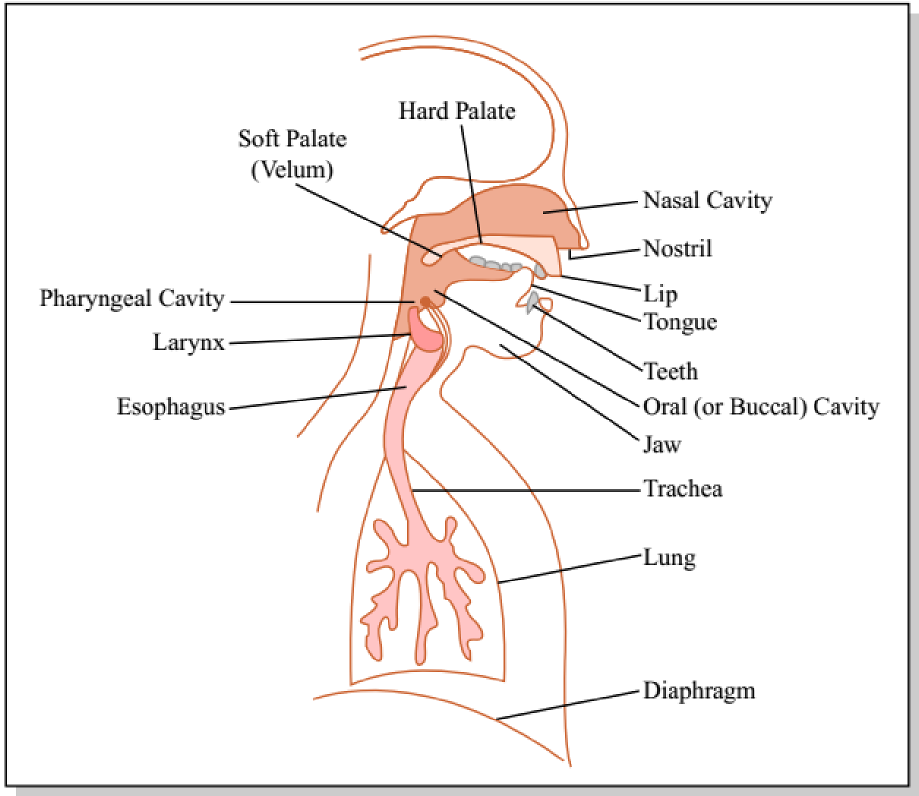
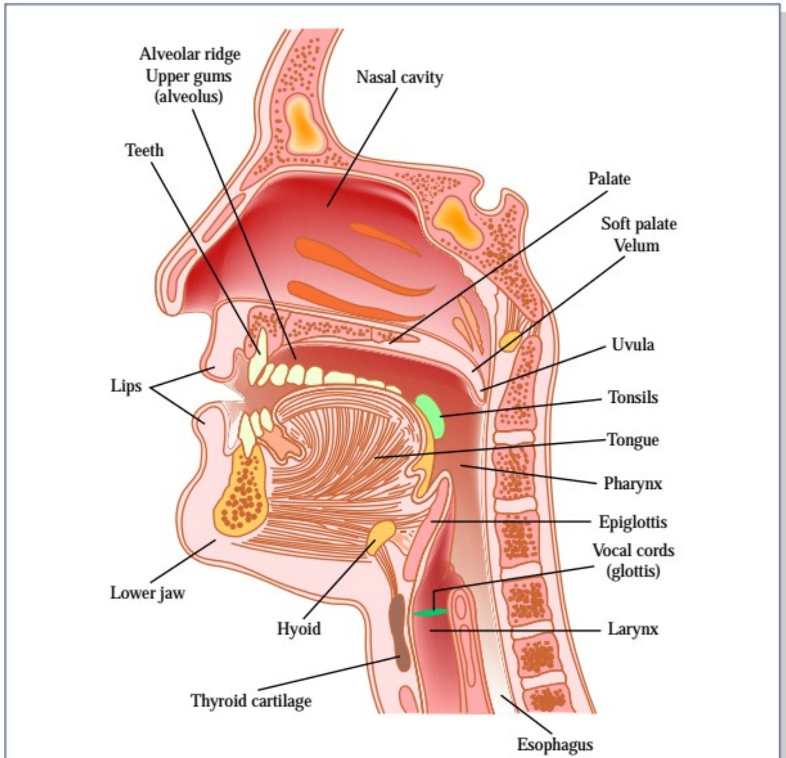


Image by MIT OpenCourseWare.



Vocal tract configuration with raised soft palate for articulating non-nasal sounds.

Image by MIT OpenCourseWare.

Content

Phone, word or N-gram occupation counts

Prosody

'Stylized' F0 / energy contours, Legendre polynomials, multinomial subspace models

Timbre

Short-term spectrum: MFCC, LFCC, LPCC, PNCC … with $\Delta+\Delta^2$

Categorical, sparse

+ Explicit/interpretable
- High error rate
- Difficult to extract

Most practical (today)

+ High accuracy
+ Easy & fast to compute
- Sensitive to disturbances

Continuous, dense

Tomi Kinnunen and Haizhou Li, "An Overview of Text-Independent Speaker Recognition: from Features to Supervectors", Speech Communication 52(1): 12--40, January 2010

Feature extraction at two levels

Level 1: short-term feature extraction

– Based on acoustic-phonetic knowledge of human speech production at millisecond time-scale
– Example: mel-frequency cepstral coefficients (MFCCs)

DIGITAL SIGNAL PROCESSING

Level 2: recording-level feature extraction

– Extraction of higher-level features that are <u>shared</u> across all the frames in a recording (examples: speaker, language, recording device, age, gender, smoking habit - you name it)
– Example: identity vector (i-vector)

MACHINE LEARNING

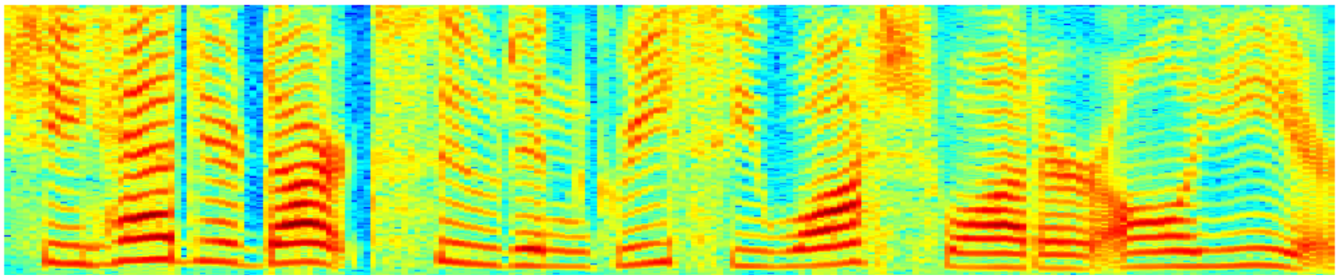
Short-term feature extraction

- Pretend that speech is a piecewise constant function over a short-time period of speech, known as a **frame**
- A feature extractor *f*(.) spits out a 'snapshot' of the short-term power spectrum ⇒ a sequence of **feature vectors** $\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_T\}$
  - Typical frame duration ~25 ms (1 ms = 1/1000th of a second)
  - Typical frame hop ~50% of the frame size
  - Typical dimension of the feature vector ~20 to 60
- Assume the observations to be **independent and identically distributed** (i.i.d.) samples from a recording-level statistical model, $\mathbf{x}_t \sim p(\mathbf{x}|\Theta_{\text{recording}})$
  - Usually, a Gaussian mixture model (GMM)

Bag-of-frames illustrated

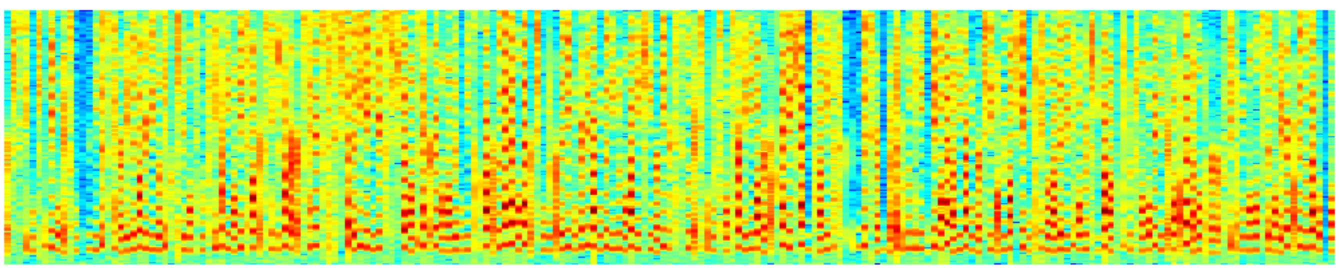
Ordering (temporal) information gets destroyed

Original



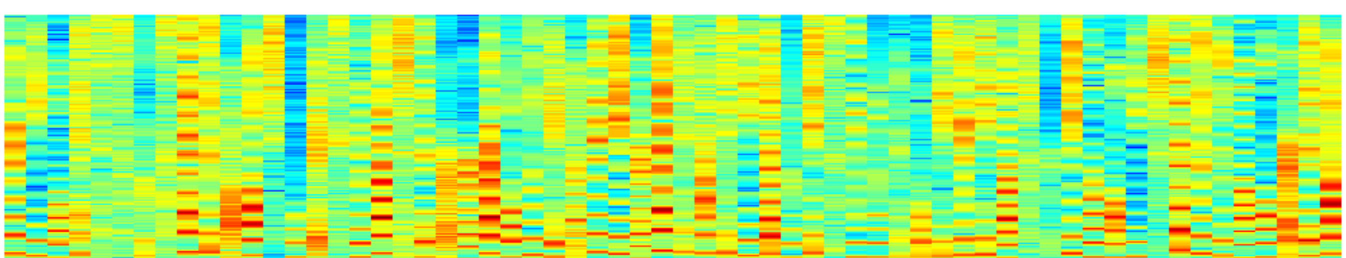
Original

Shuffled frames



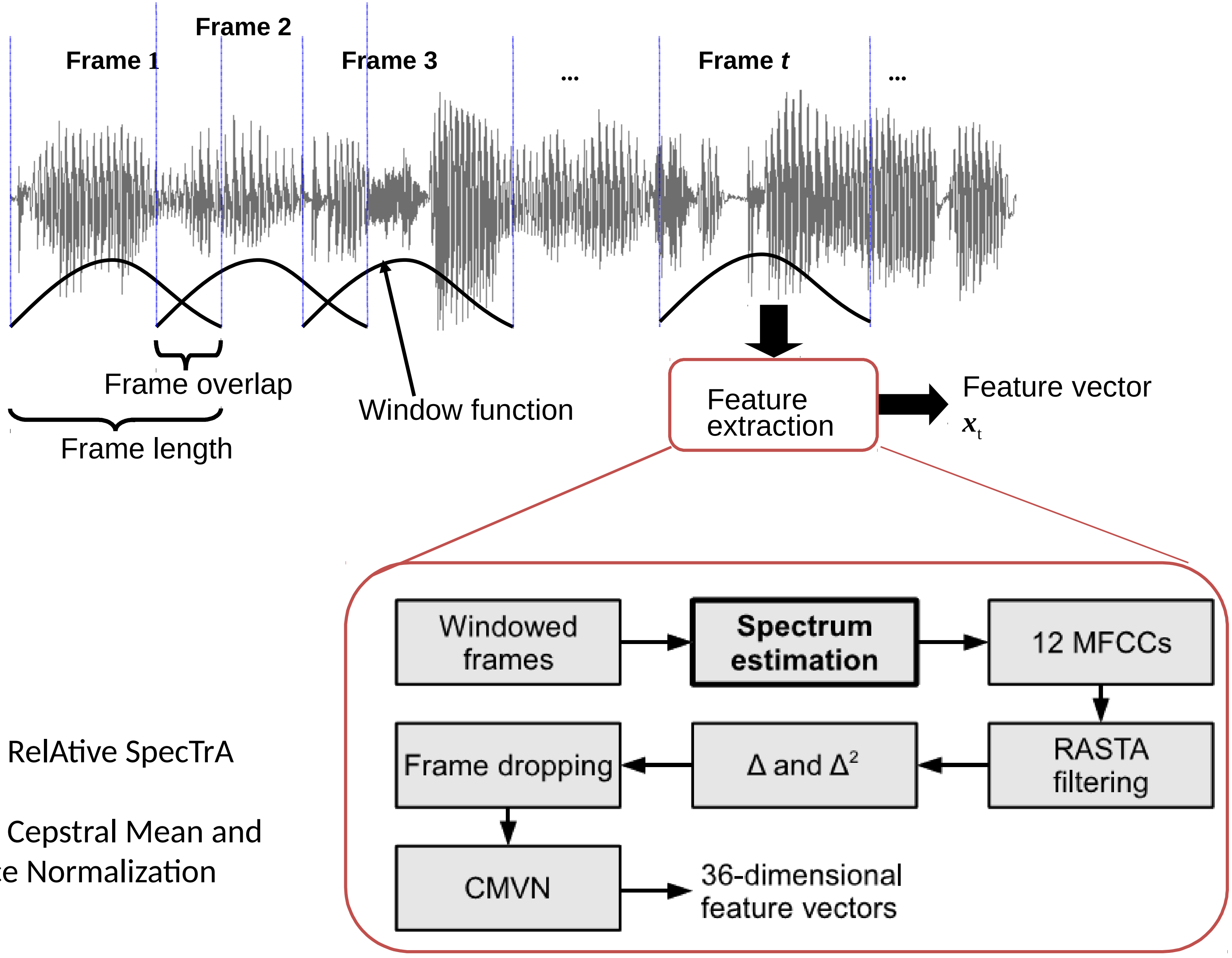
Frames shuffled, 30 ms frame

Shuffled frames



Frames shuffled, 100 ms frame

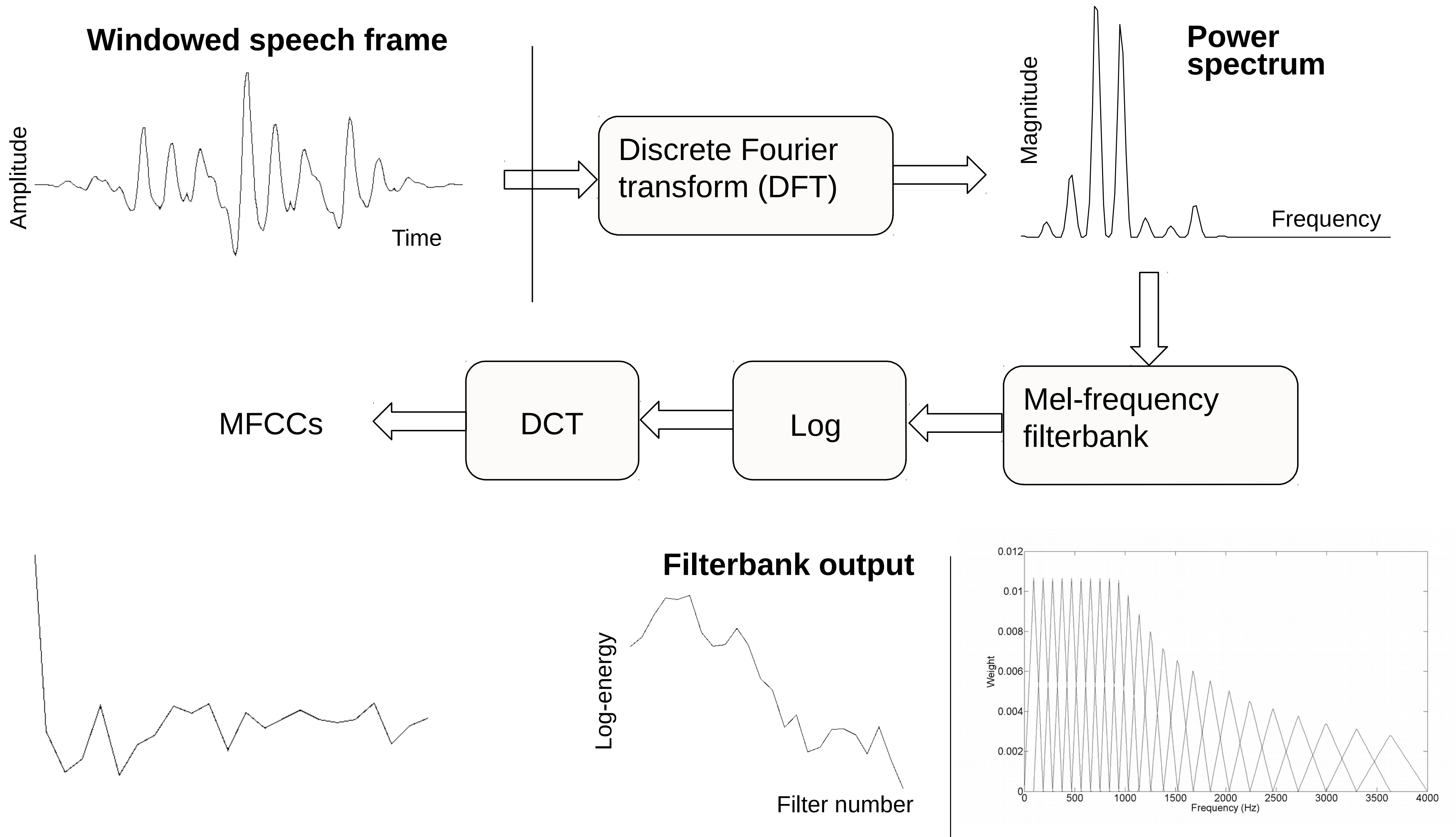
MFCC extraction



RASTA: RelAtive SpecTrA

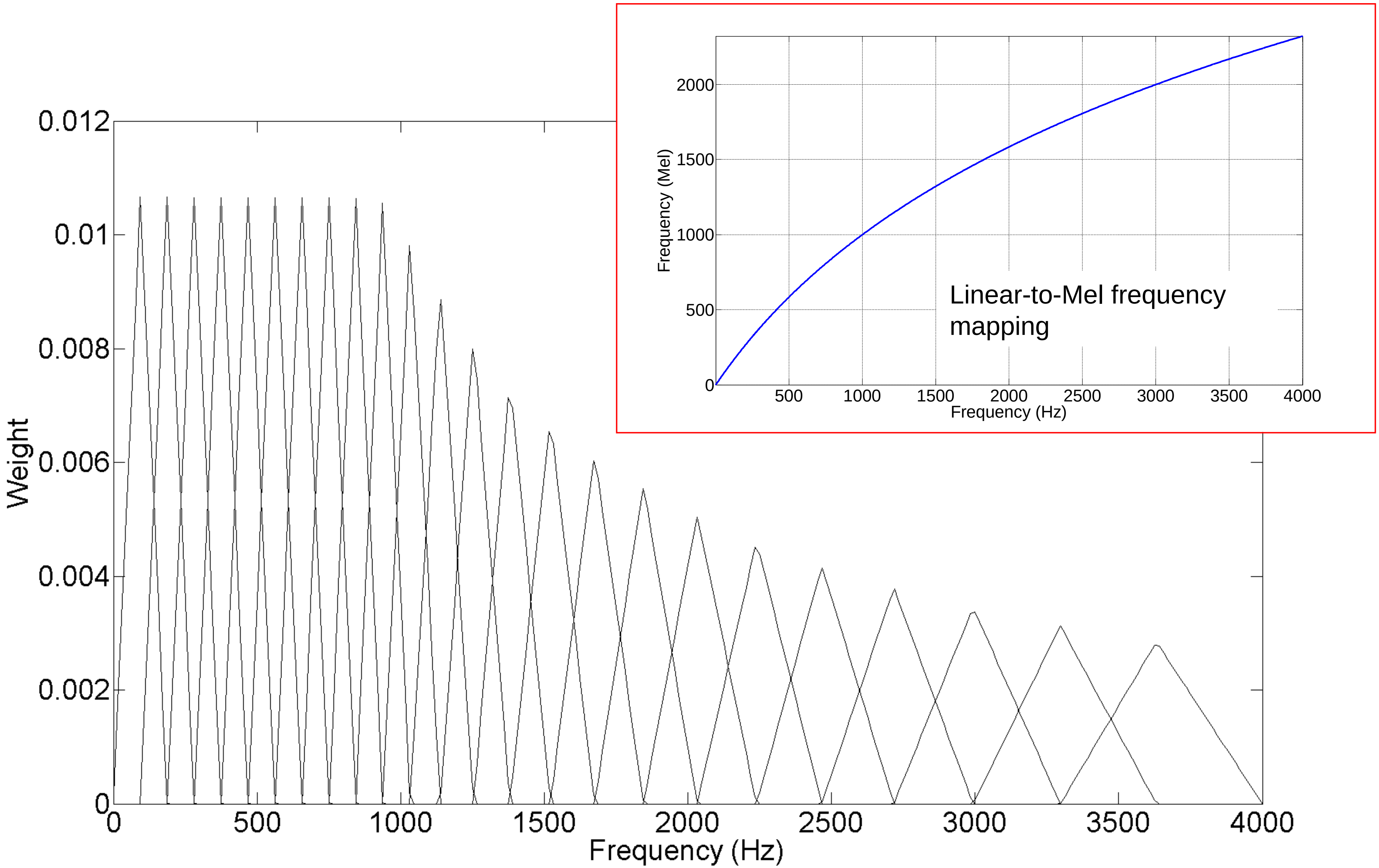
CMVN: Cepstral Mean and Variance Normalization

Mel-frequency cepstral coefficient (MFCC) features



Mel-frequency filterbank

[Generated using 'RASTAmat' package of Dan Ellis]

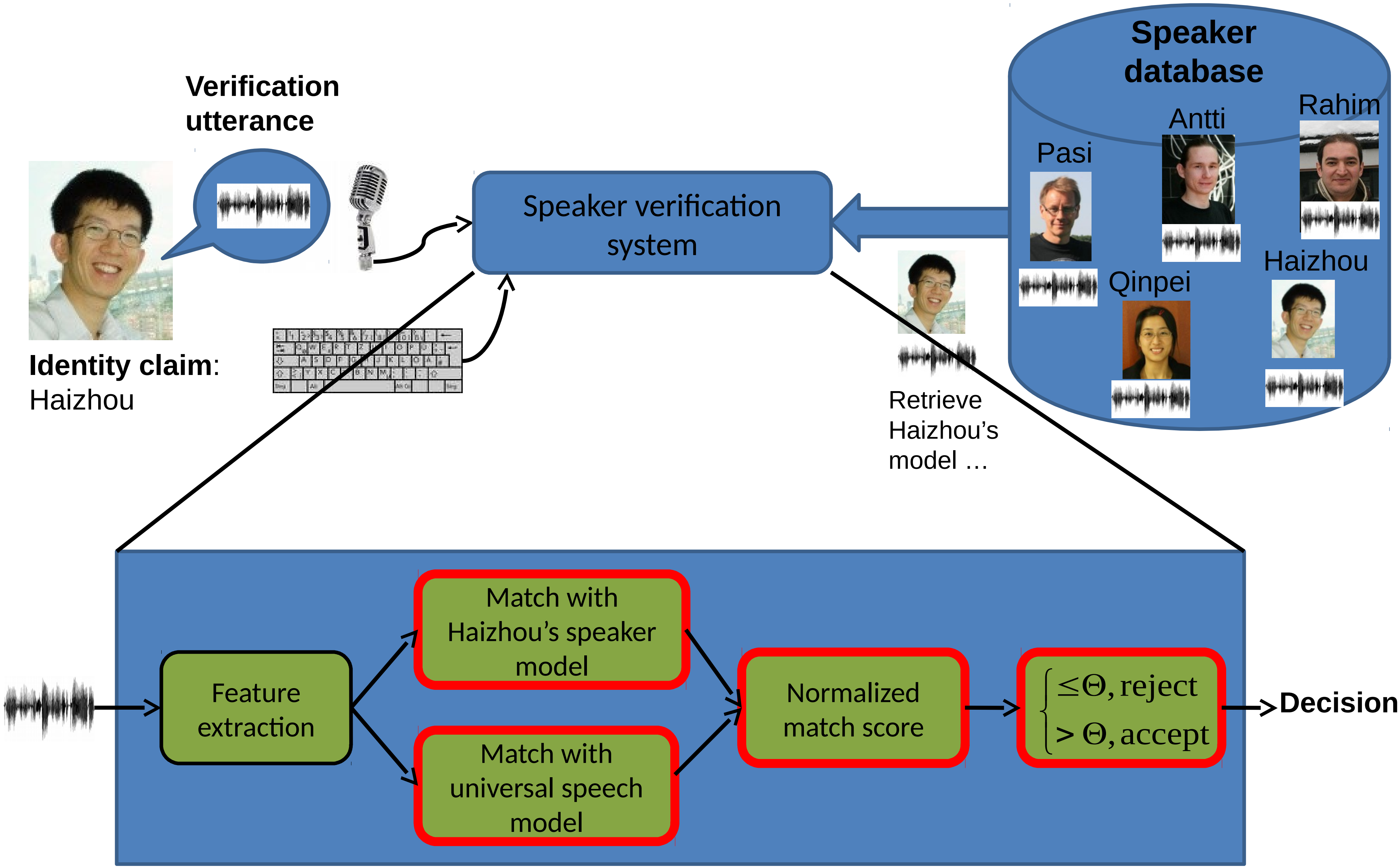


Speaker verification: science and art of data-driven modeling

"... from the speaker-recognition research trend in the last decade, it seems that improving feature robustness beyond a certain level (for a variety of degradations) is extremely difficult—or, in other words, **data-driven modeling techniques have been more successful in improving robustness compared to new features**"

[John H.L. Hansen and Taufiq Hasan, Speaker Recognition by Machines and Humans: A Tutorial Review, IEEE Signal Processing Magazine, Nov 2015]

Speaker modeling and comparison



History of speaker modeling

1970s and before

- Long-term feature averaging

1980s and 1990s

- Dynamic time warping (DTW), vector quantiz. (VQ)
- Hidden Markov Models
- Early neural net models

~1995 to ~2005

- Gaussian mixture models (GMMs)
- Universal background model

2005—today

- GMM supervectors
- Joint factor analysis (JFA)
- i-vectors
- Probabilistic linear discr. analysis (PLDA) scoring
- Deep neural nets

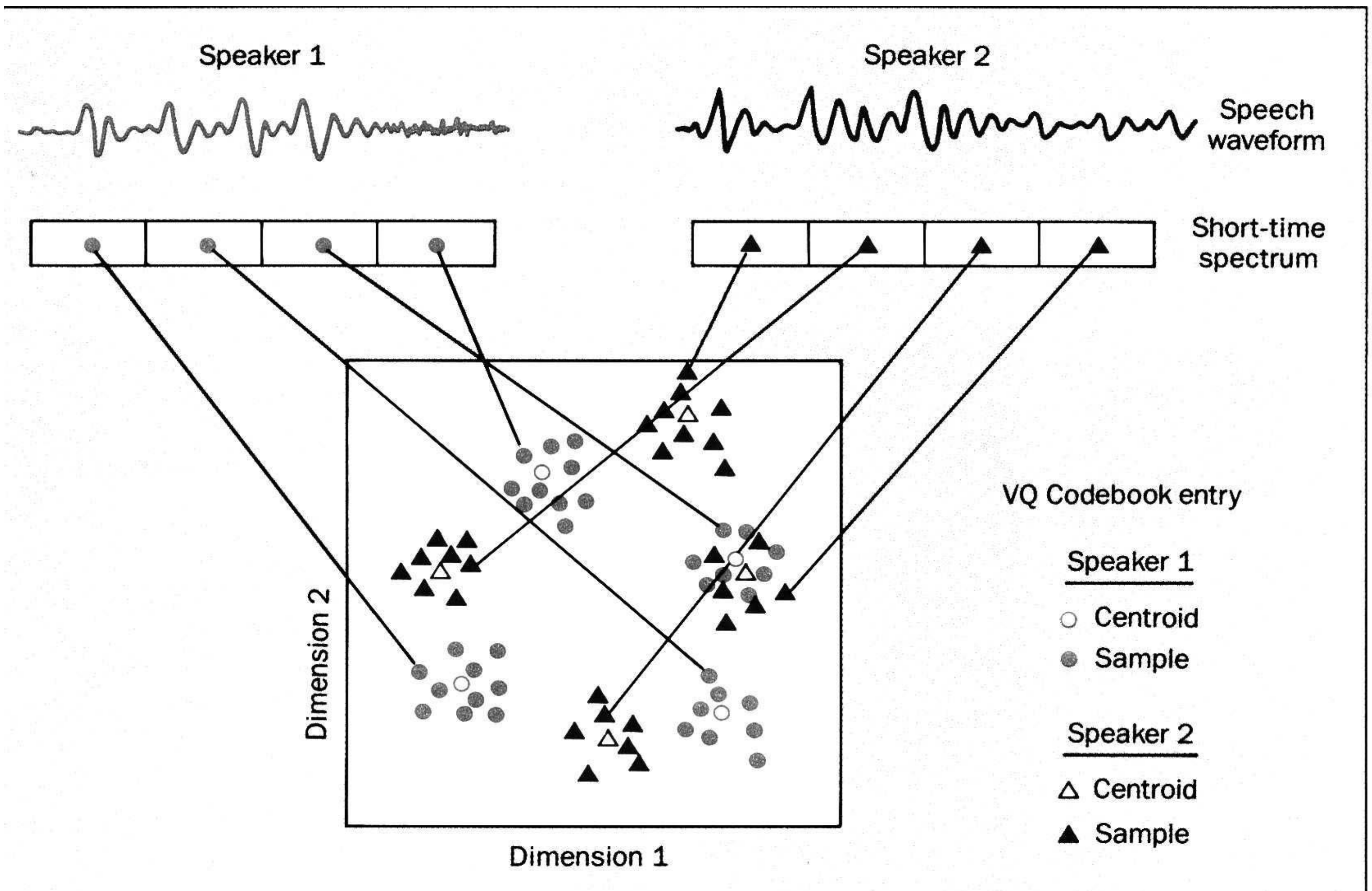
- All rooted on GMMs
- 1996 onwards: NIST SREs
- Focus on text-independent models

Pre-history

Modern era

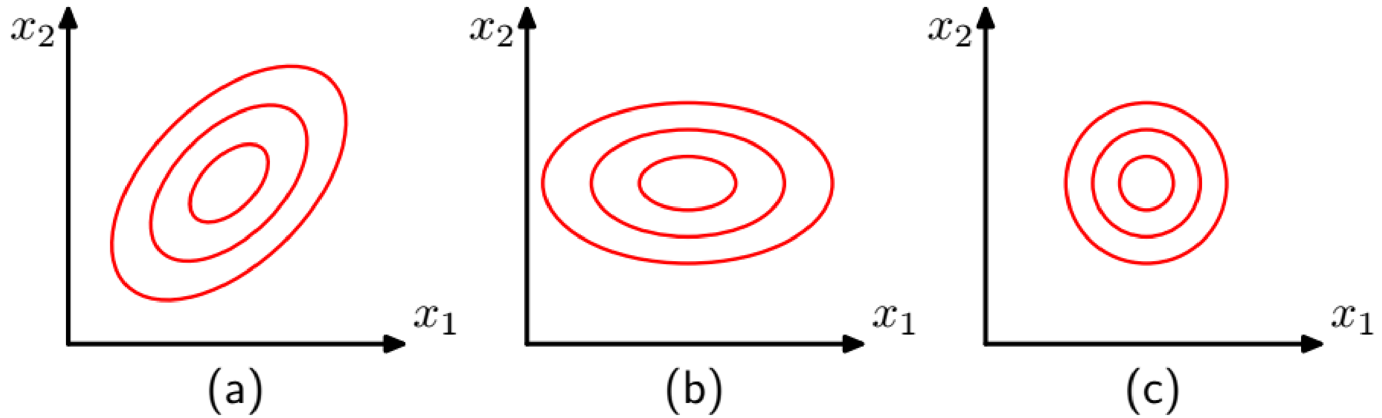
Speaker / language recognition is about modeling the”feature clouds”

[F. K. Soong, A. E. Rosenberg, L. R. Rabiner and B. H. Juang, “A Vector Quantization Approach to Speaker Recognition,” *AT&T Technical Journal*, Vol. 66, pp. 14-26, Mar/Apr 1987]



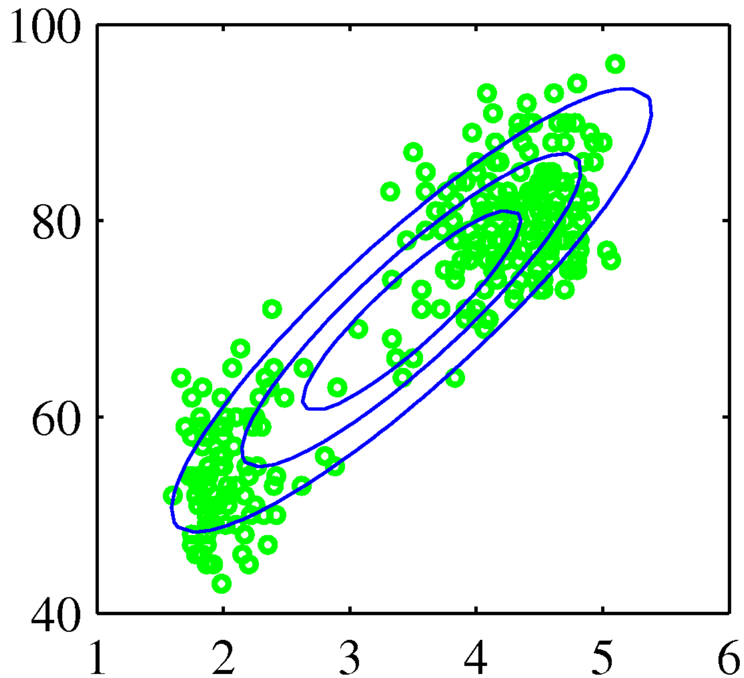
Single Gaussian

Bi-variate normal distribution

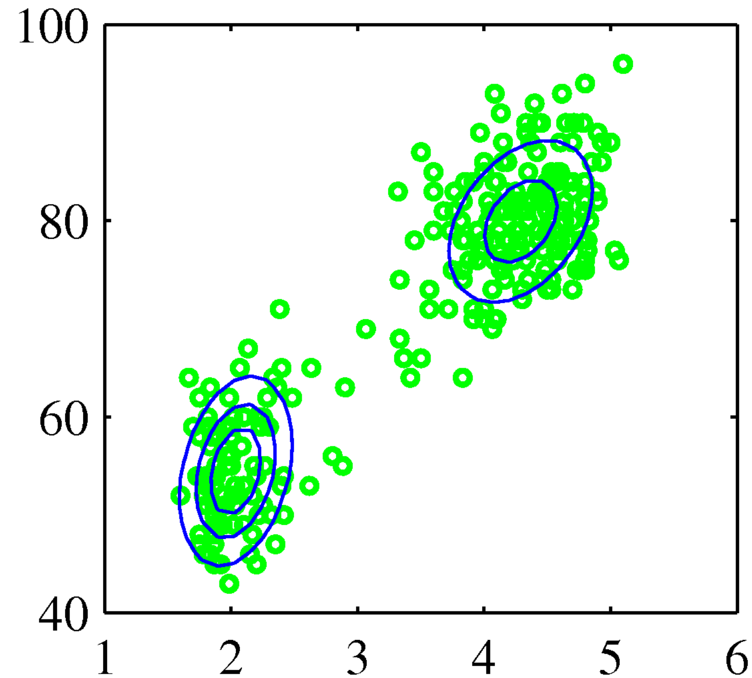


- (a) Covariance matrix $\mathbf{\Sigma}$ is full, rank $\mathbf{\Sigma} = 2$, (b) diag $(\mathbf{\Sigma}) = (a, b)^t$ and (c) $\mathbf{\Sigma} = a * I$.
- How many parameters we need to estimate in (a), (b) , (c) ?

Gaussian mixture model (GMM)

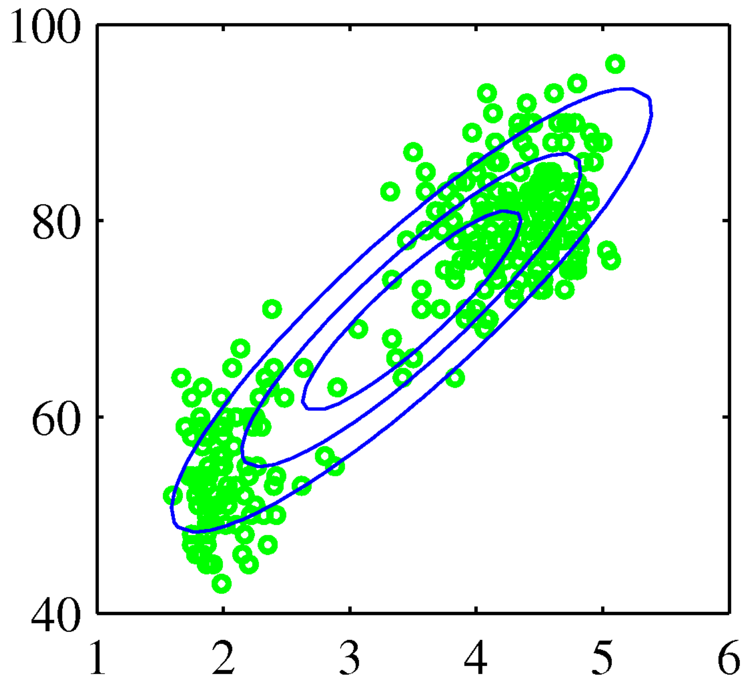


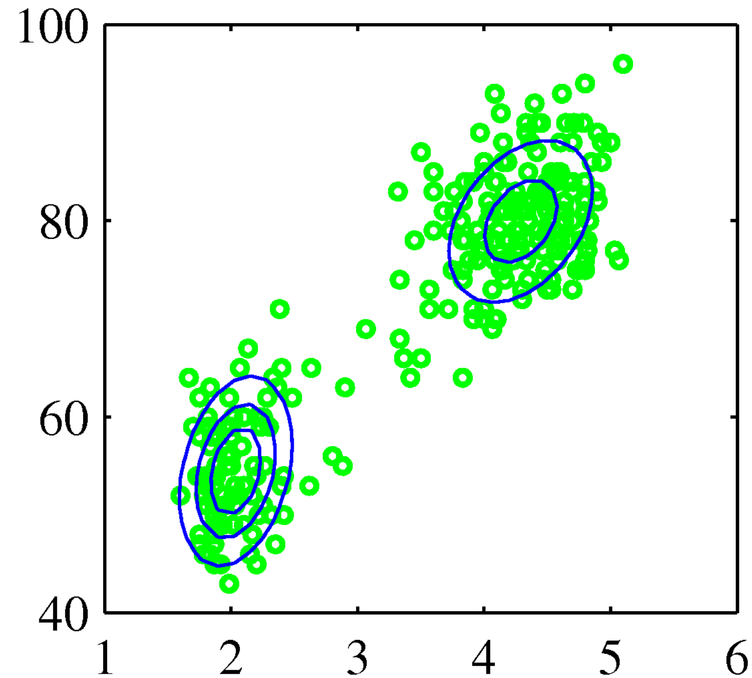
1 Gaussian not good enough ?



Looks better, yes? ^^

Gaussian mixture model (GMM)



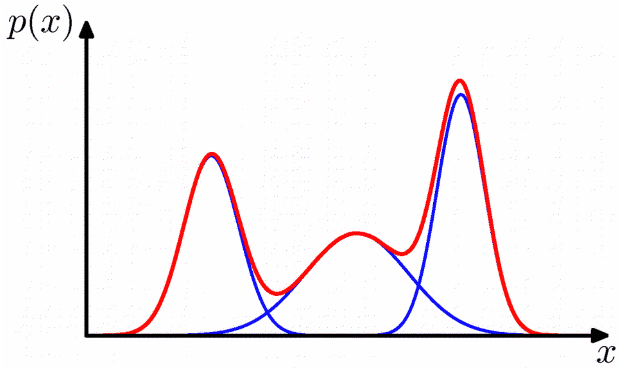


1 Gaussian not good enough ?

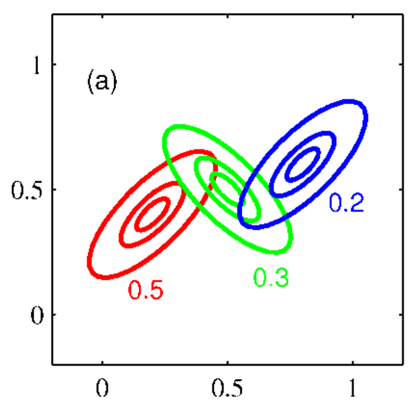
Looks better, yes? ^^

GMM as a density estimator

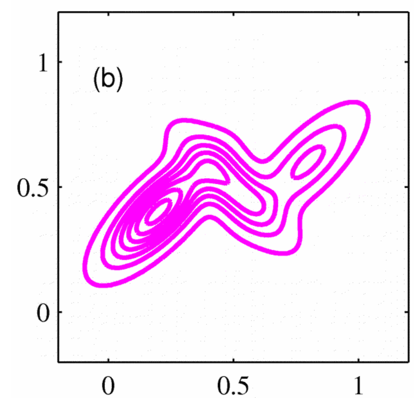
1-d data



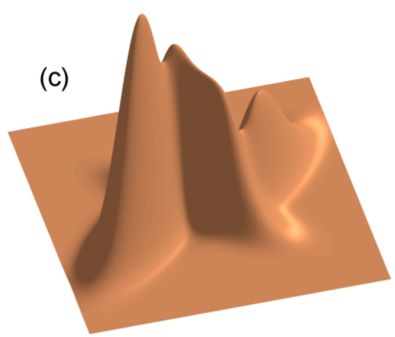
2-d data



Individual components

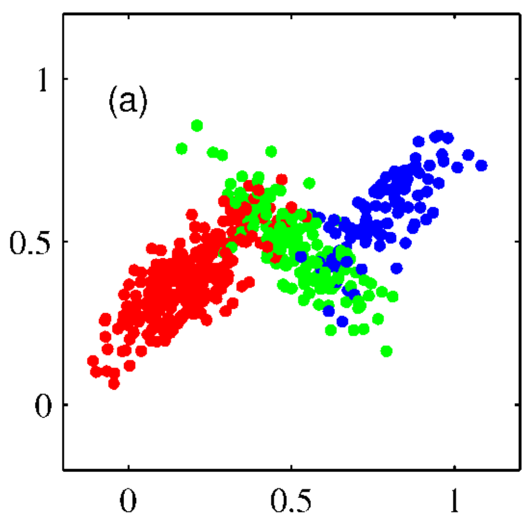


Density contour

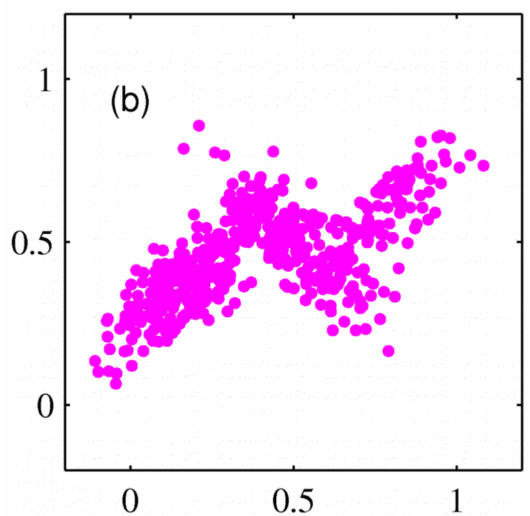


... another visualization

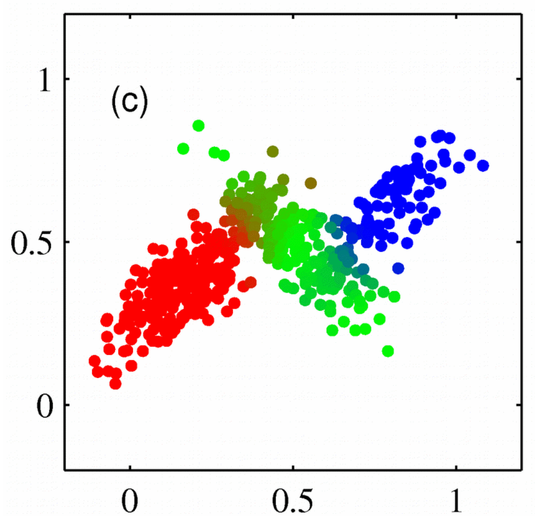
GMM as a latent (hidden) variable model



”Oracle”: Assignment of data points to components is known (red, green or blue)

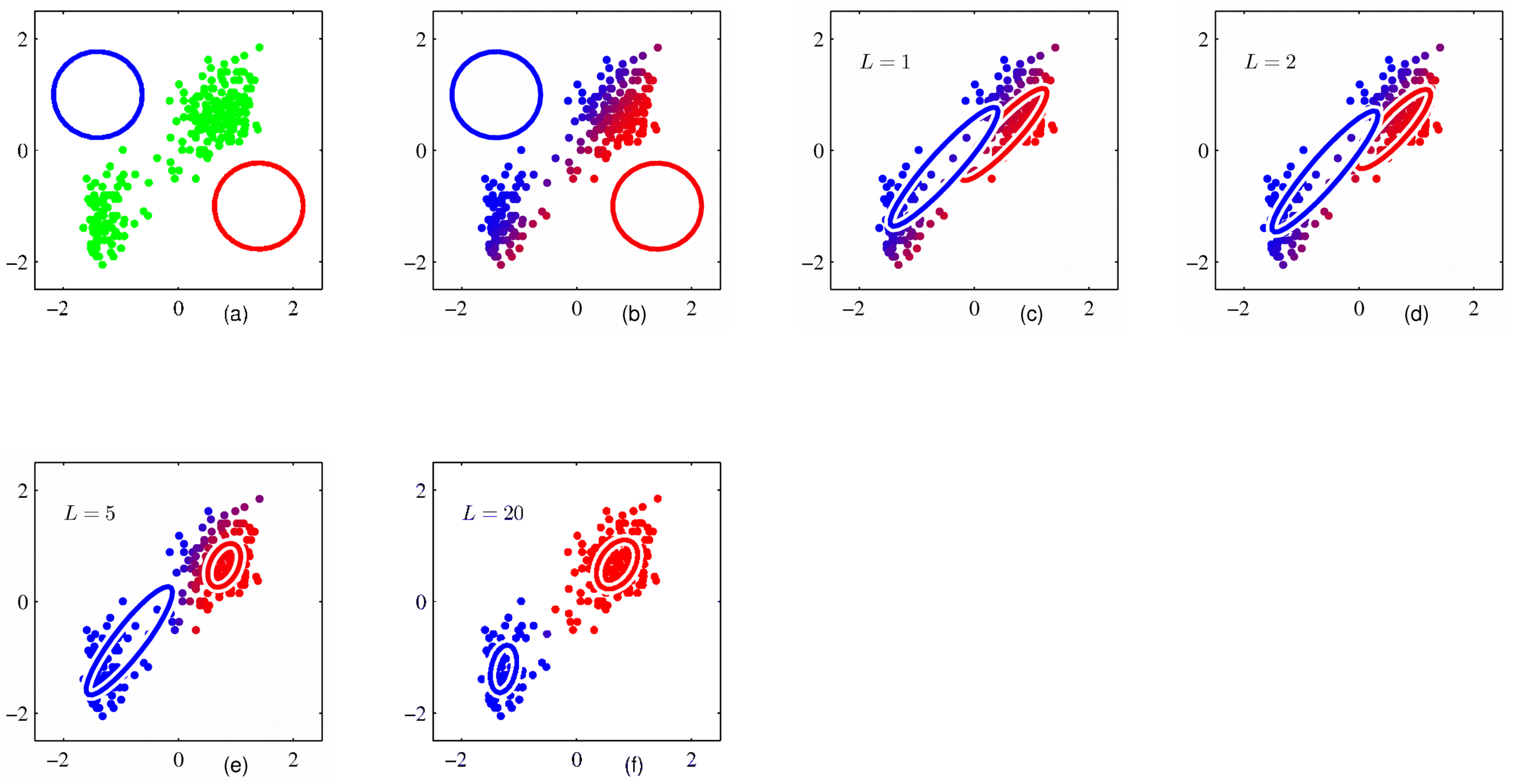


... but this is how our training sample is given: no labels!



strength of each color = posterior probability that the given observation was generated by the corresponding component

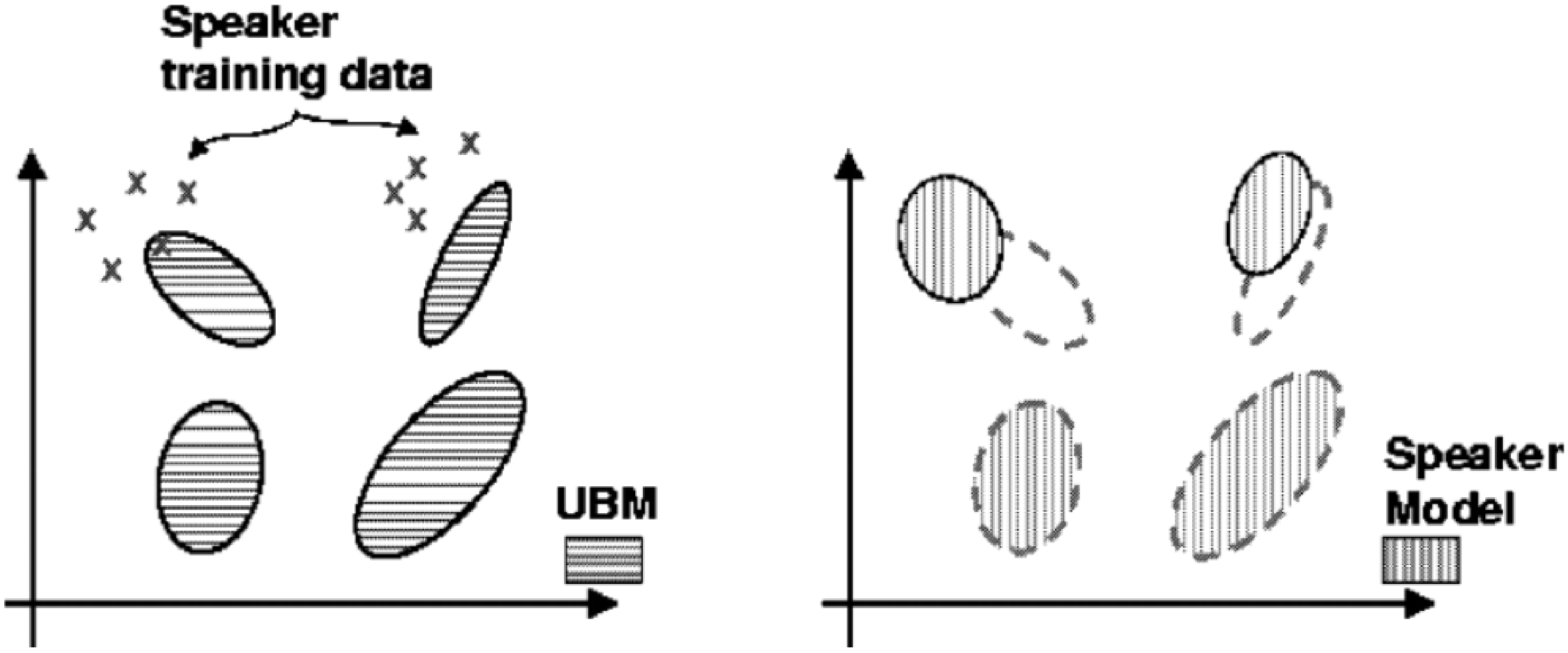
Progress of EM algorithm



GMM as a 'data compressor'

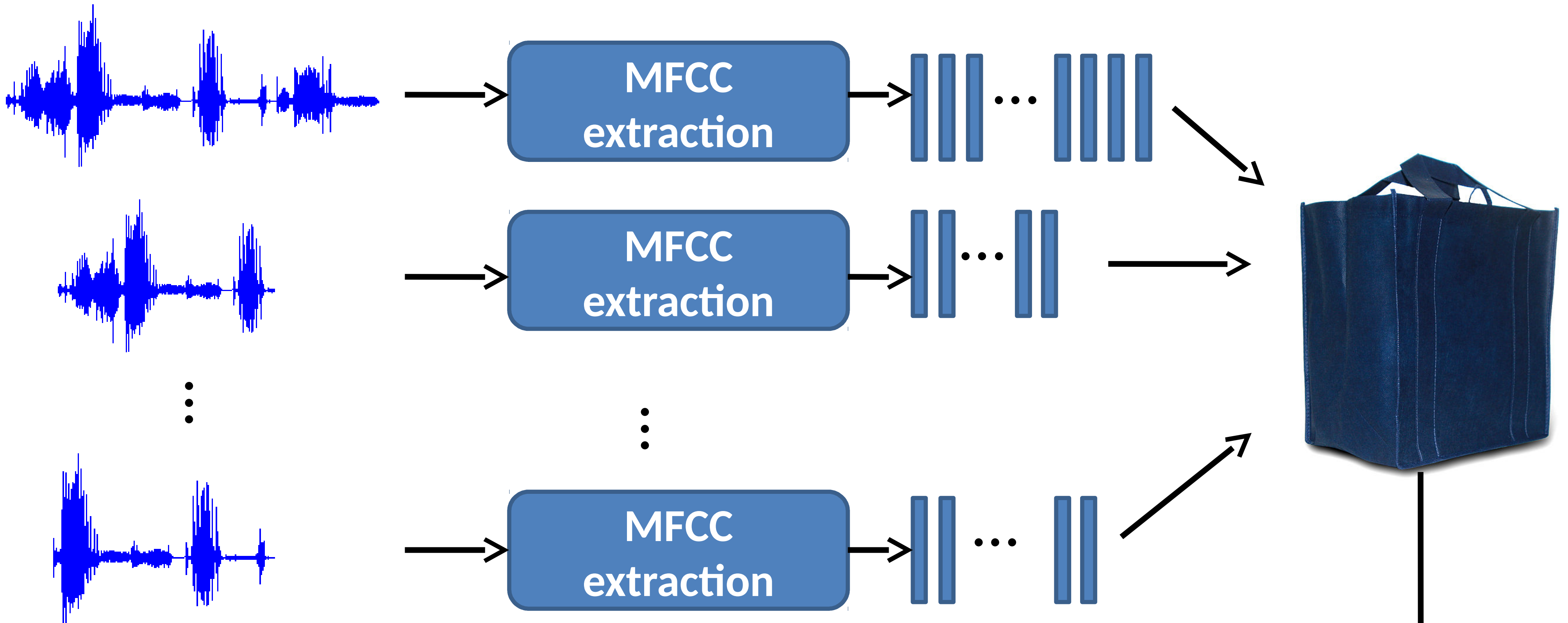
- Assume 1 minute recording of audio sampled with 8 kHz. Raw data has 8000 x 60 = **480,000 samples**
- Assume 12 MFCCs extracted every 10 ms ⇒ 12 coeffs x 100 frames/sec x 60 sec = **72,000 feature values**
- A GMM with 128 Gaussians & diagonal covariance matrices has
  - 128 mean vectors (dimensionality = 12)
  - 128 variance vectors (dimensionality = 12)
  - 128 mixing weights
  - Total 128*12 + 128*12 + 128 = **3200 parameters**

Gaussian mixture model – universal background model (GMM-UBM)



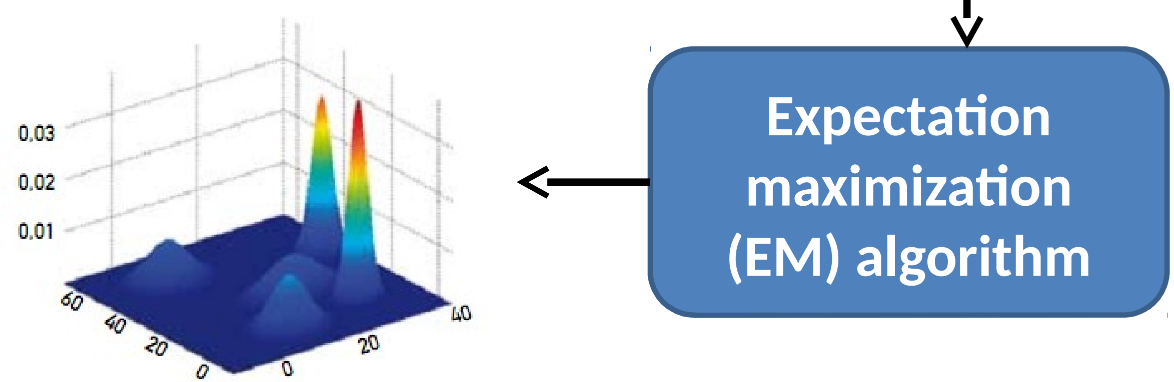
Douglas A. Reynolds and Thomas F. Quatieri and Robert B. Dunn, ”Speaker Verification Using Adapted Gaussian Mixture Models”, *Digital Signal Processing*, 2000.

Step 1: training the UBM



Utterances from a large number of different speakers

- No need for speaker identity, transcripts or other metadata
- Sometimes, gender information can be useful



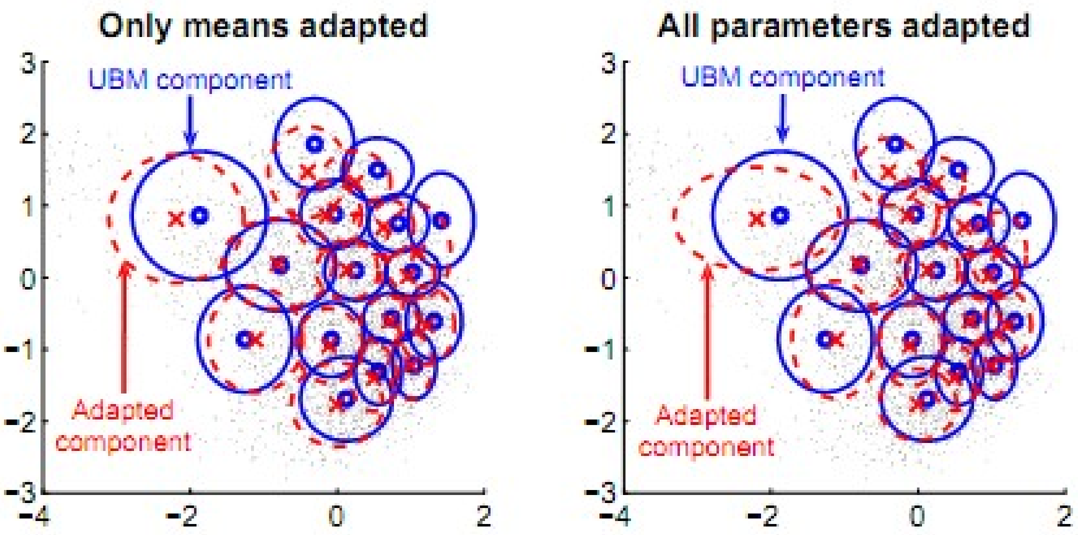
Step 2: speaker enrollment via maximum a posteriori (MAP) adaptation

$$\boxed{\boldsymbol{\mu}'_k = \alpha_k \tilde{\mathbf{x}}_k + (1 - \alpha_k)\boldsymbol{\mu}_k}$$

Adapted mean vector

Mean of training data

Prior mean from univ. background model (UBM)



$$\alpha_k = \frac{n_k}{n_k + r}$$

Adaptation coefficient (r = relevance factor)

$$\tilde{\mathbf{x}}_k = \frac{1}{n_k}\sum_{t=1}^{T} P(k|\mathbf{x}_t)\mathbf{x}_t$$

Mean of training data assigned to kth Gaussian

$$n_k = \sum_{t=1}^{T} P(k|\mathbf{x}_t)$$

Soft count of vectors assigned to kth Gaussian

$$P(k|\mathbf{x}_t) = \frac{P_k \mathcal{N}(\mathbf{x}_t|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{m=1}^{K} P_m \mathcal{N}(\mathbf{x}_t|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)}$$

Posterior probability of the kth Gaussian for one feature vector

Step 3: speaker comparison

- $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ is a sequence of test utterance feature vectors and $\boldsymbol{\theta}_s$ is a GMM of speaker *s*, claimed to have ’generated’ **X**, adapted from the UBM as explained above.

- We compute log-likelihood ratio

$$LLR(\mathbf{X}) = \log p(X|\boldsymbol{\theta}_s) - \log p(X|\boldsymbol{\theta}_{ubm})$$

- The larger the LLR value, the stronger the evidence for the ”same speaker” hypothesis. Binary decision is made by comparing LLR(**X**) against a pre-selected threshold (such as 0).

- Intuition: $\log p(X|\boldsymbol{\theta}_{ubm})$ represents the uninformative part common to all speakers (such as speech content)

GMM supervectors

[W. M. Campbell, D. E. Sturim, D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification", *IEEE Signal Proc Lett* 2006]