Vector Quantization

Representations of Speech

- Information rate of raw speech (Time Domain)
  - ✓ Sampling Frequency = 10 KHz = 10000 samples/sec
  - ✓ # bits/sample = 16 bits
  - ✓ # bits for 1-sec speech = 10000 × 16 = 1,60,000 samples/sec = 160 kbps
- Spectral Representation (LP analysis / Filter Bank Analysis)
  - ✓ Speech -> Frames of 20 ms with 10 ms frame-shift
  - ✓ 1-sec of speech -> 100 frames/sec
  - ✓ 1 speech frame -> 1 spectral vector with 10 coefficients
  - ✓ 1-sec of speech -> 100 frames × 10 coeffs × 16 bits/coeff -> 16000 bits/sec -> 16 kbps
- Raw speech -> Spectral representation : 160 kbps to 16 kbps

Representations of Speech : Vector Quantization

- Information Rate in the context of speech

  - ✓ Speech signal -> 40 basic sound units (approx…)
  - ✓ Based on inherent variability of speech -> 25 variations/sound unit
  - ✓ As spectral vector represents sound unit & by considering variability of sound units
    - ➢ Total number of distinct spectral vectors required -> 40 × 25 = 1000 (approx..)
    - ➢ Encoding 1000 distinct spectral vectors (SVs)requires -> 10 bits ($2^{10} = 1000$)

- 1-sec of speech -> 100 frames -> 100 SVs -> 100 × 10 = 1000 bits/sec = 1kbps

- Raw speech -> 10000 samples × 16 bits/sample = 160000 bits/sec = 160 kbps

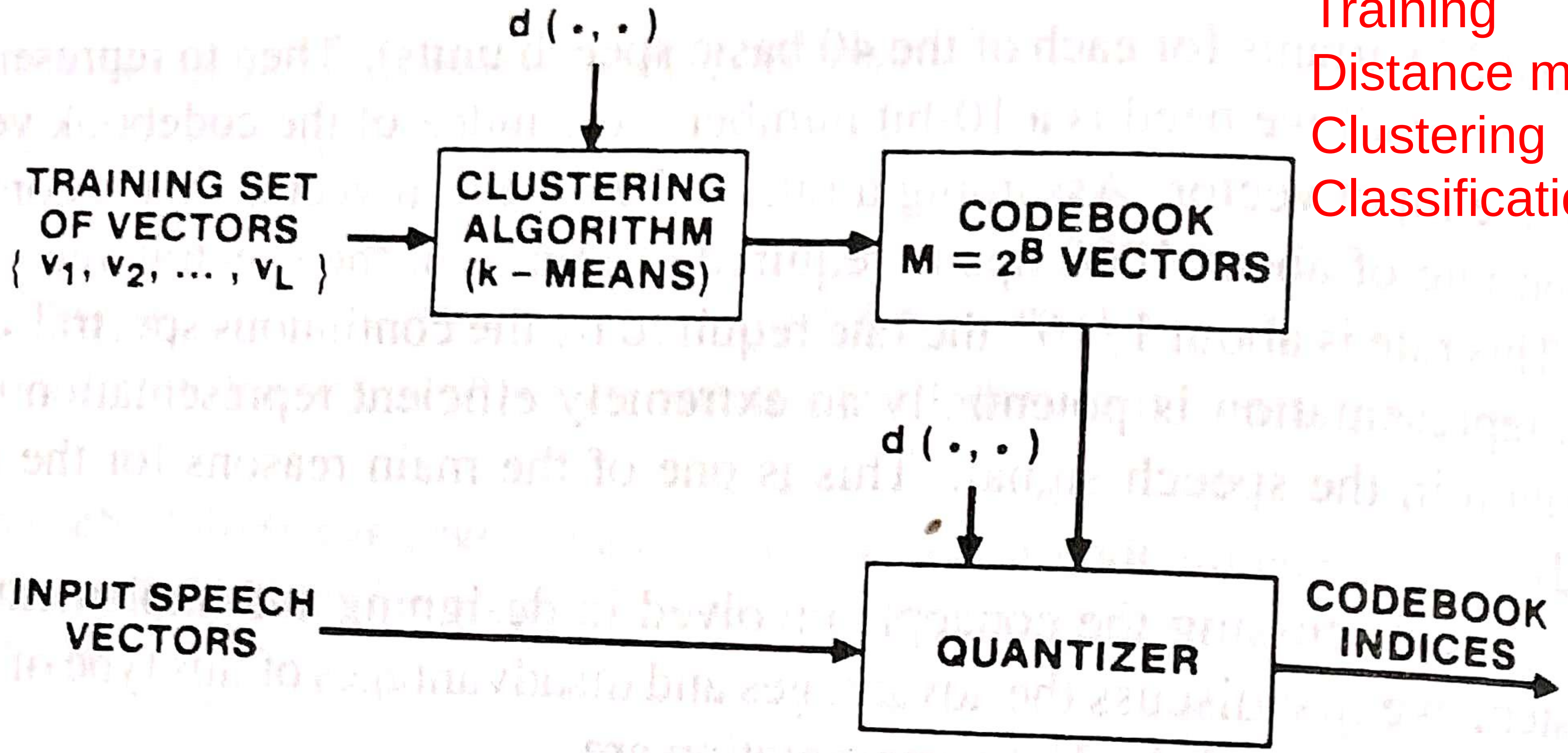
- Spectral representation of speech = 100 frames × 10 coeffs × 16 bits/coeff = 16000 bits/sec = 16 kbps

- Vector quantization = 100 frames -> 100 SVs × 10 bits/sv = 1000 bits/sec = 1 kbps

VQ (Advantages vs Disadvantages)

- Advantages
  - ✓ Reduced storage
  - ✓ Reduced computation for determining similarity
  - ✓ Discrete representation of speech sounds
- Disadvantages
  - ✓ Inherent spectral distortion in representing the actual spectral vector
  - ✓ Storage required for codebook vectors
- Trade-offs
  - ✓ Quantization error
  - ✓ Similarity computation cost for choosing the codebook vector
  - ✓ Storage of codebook vectors

Elements of VQ Implementation



Training
Distance measure
Clustering
Classification

Elements of VQ Implementation

- Training step : A large set of spectral vectors : v1, v2, v3, …. vL

  ✓ Codebook size = M (M = 2^{B}) -> B-bit codebook
  ✓ L >> M (at least L should be 10M)

- Measure of similarity between a pair of SVs : d(vi, vj) = dij (spectral distance)

- Centroid computation

  ✓ L training vectors -> M clusters
  ✓ M codebook vectors -> centroids of M clusters

- Classification procedure

  ✓ Arbitary spectral vector -> closest codebook vector
  ✓ Nearest neighbor labeling

The VQ Training Set & Distance Measure

- Training data set
  - ✓ Speakers : age, accent, gender, speaking rate, energy/emotion levels, etc…
  - ✓ Speaking environments : quite room, automobile, crowded places, babble noise, etc..
  - ✓ Transducers and Transmission systems : wideband microphones, telephone handsets, direct transmission, telephone channel, wideband channels, and other devices etc..
  - ✓ Speech units : Digits, conversational speech, isolated words, etc…
- Similarity/Distance measure
  - ✓ $d(v_i, v_j) = d_{ij} \{ d_{ij} = 0 \,(if\ v_i = v_j); d_{ij} > 0 \,(if\ v_i \neq v_j)\}$

Clustering the Training Vectors

K-Means Clustering (Generalized Lloyd Algorithm) : Training Vectors → M Codebook Vectors

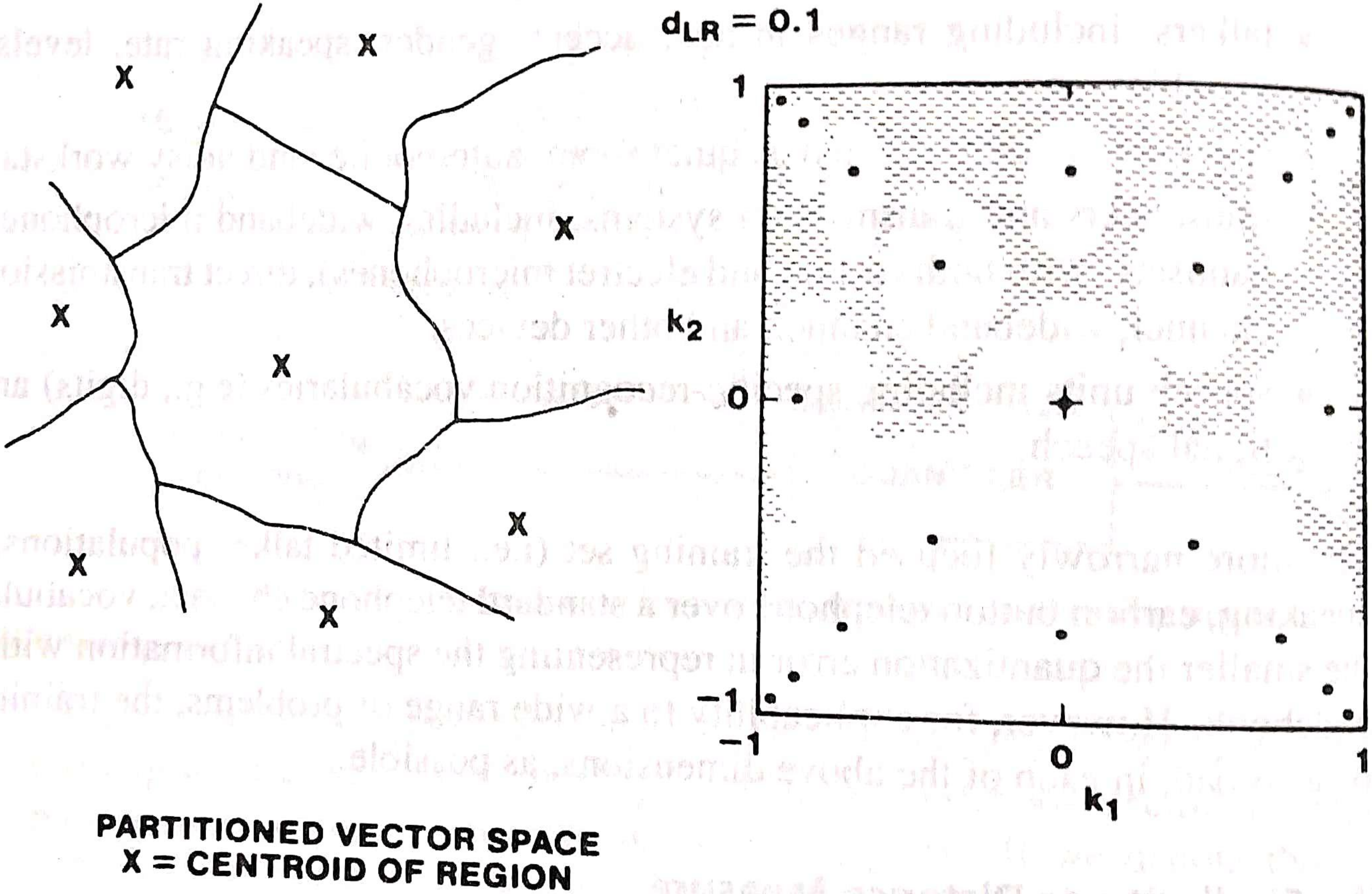
- ✓ Initialization : Arbitrarily choose M vectors as initial set of code words in the codebook

- ✓ Nearest-Neighbor Search : Each SV -> One of the M code words in the current codebook. Based on spectral distance associate each SV to the closest code word.

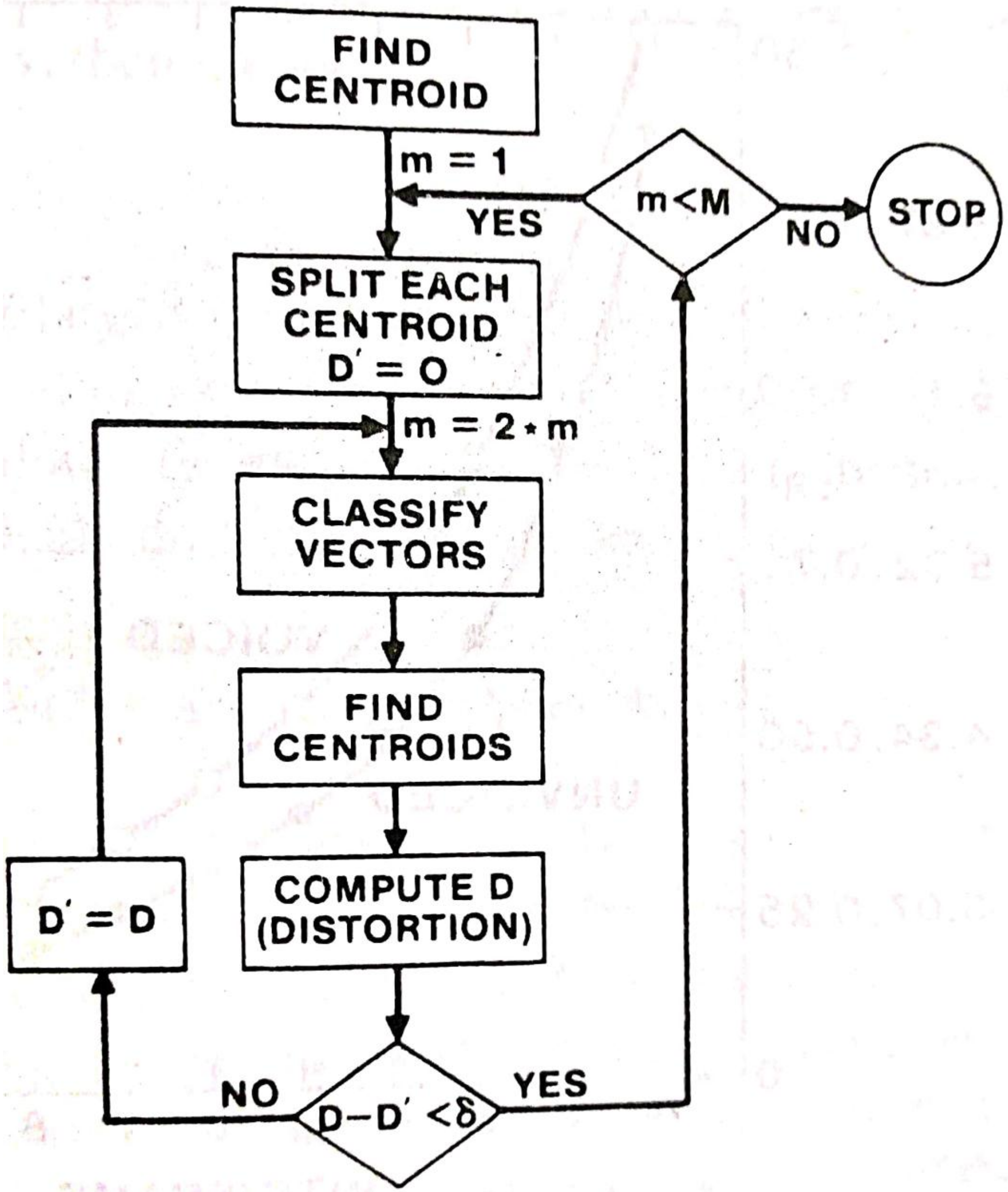
- ✓ Centroid update : Update the code word in each cell using the centroid of the training vectors assigned to that cell.

- ✓ Iteration : Repeat steps 2 and 3 until the average distance falls below the preset threshold.

Clustering the Training Vectors



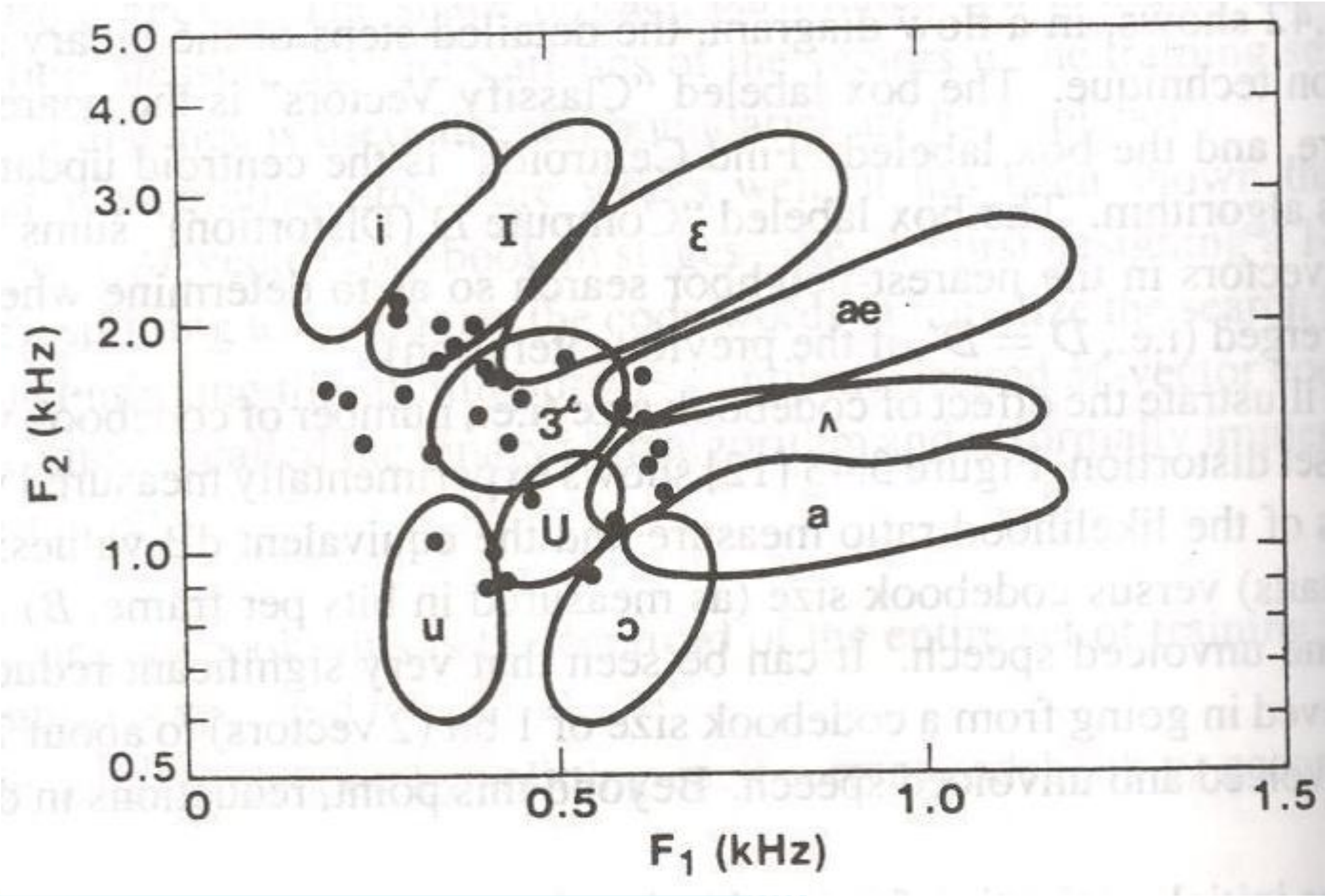
VQ : Binary-Split Algorithm



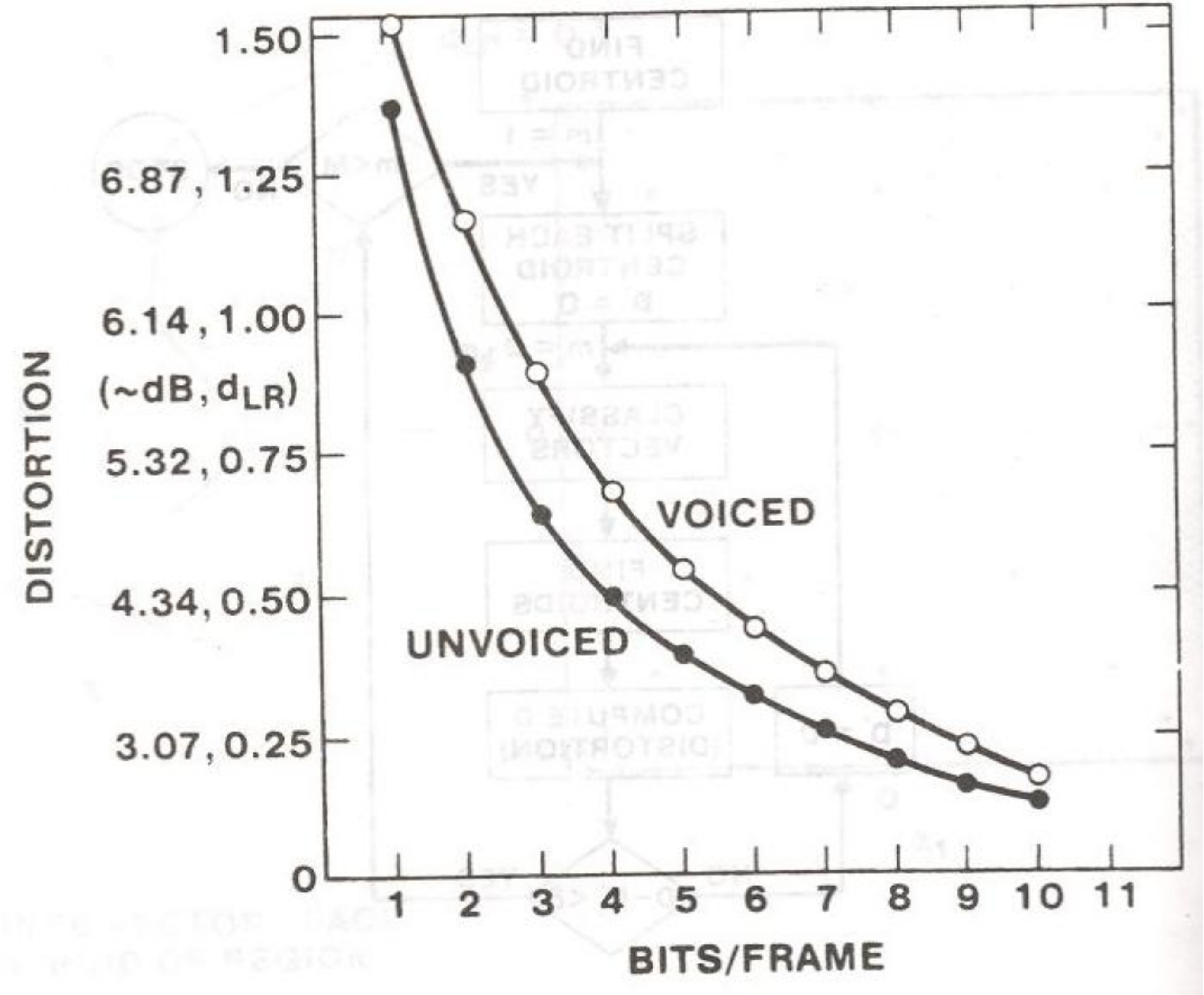
Binary-Split Algorithm

- ✓ Design a 1-vector codebook using all training SVs.
- ✓ Double the size of codebook by splitting each current entry $Y_n = \{Y_n^+ = Y_n(1 + \varepsilon); Y_n^- = (1 - \varepsilon)\}$; *n* = codebook size; ε = (0.01 to 0.05)
- ✓ Use K-means algorithm to get the best set of centroids for the split codebook.
- ✓ Iterate steps 2 & 3 till codebook size reaches to desired size M.

Codebook Vector Locations in F1 – F2 Plane



Codebook Distortion vs Codebook Size



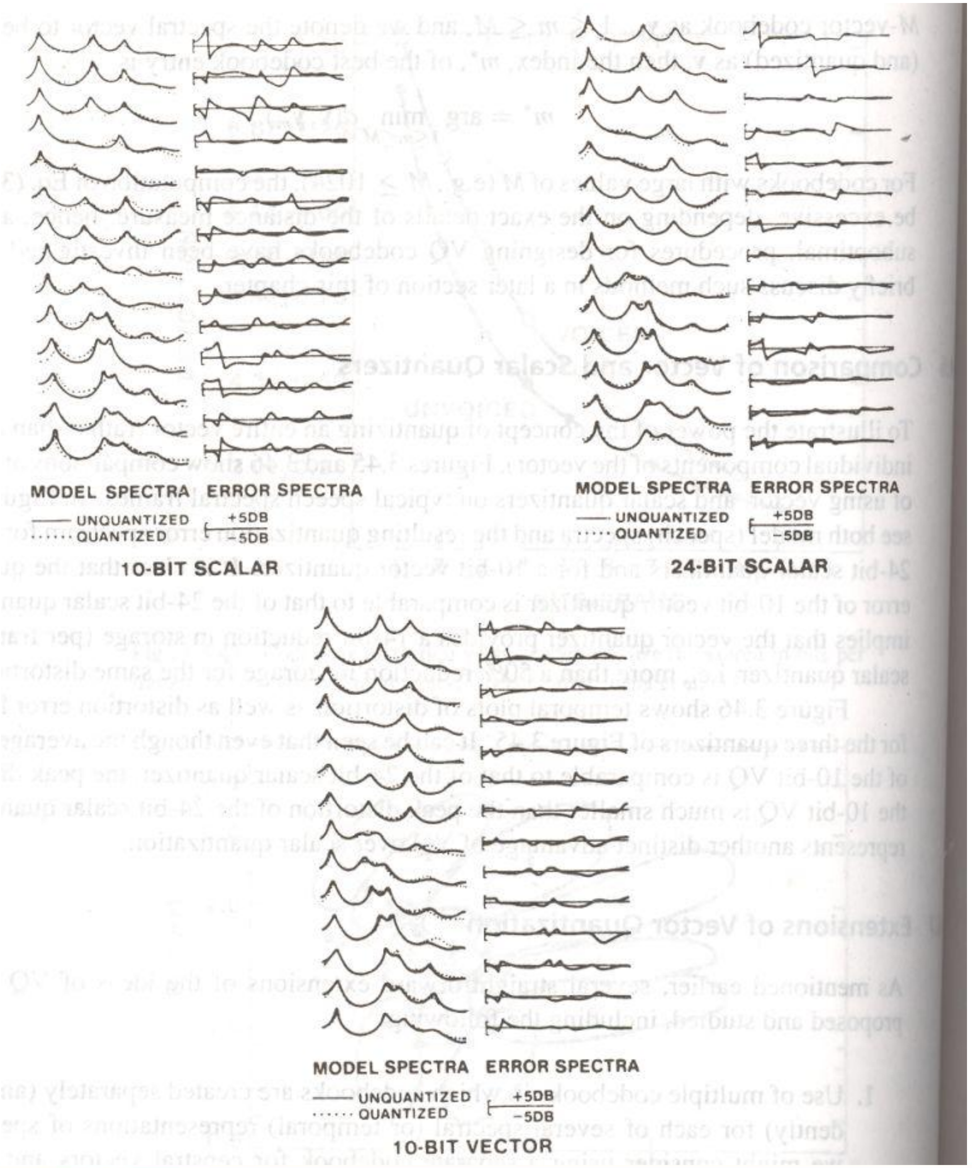
Vector Classification Procedure

$$m^* = arg \min_{1 \le m \le M} d(v, y_m)$$

$$y_m = one\ of\ the\ codebook\ entry\ (code\ word); 1 \le m \le M.$$

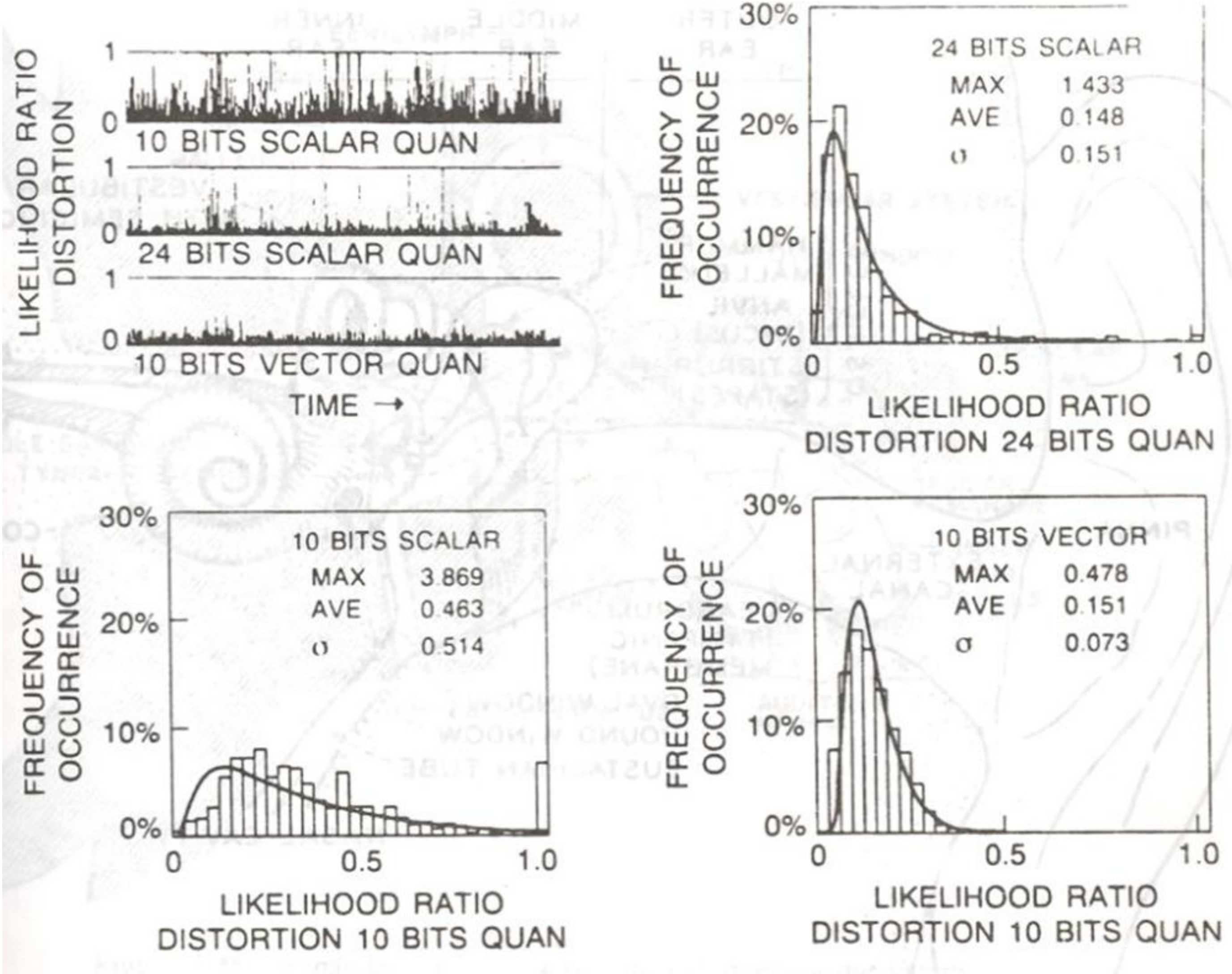
$$v\ - spectral\ vector$$

Comparison of Vector & Scalar Quantizers



- Model & Distortion Error Spectra
  - ✓ 10-bit scalar quantizer
  - ✓ 24-bit scalar quantizer
  - ✓ 10-bit vector quantizer

Comparison of Vector & Scalar Quantizers



Histograms of Temporal Distortion

- ✓ 10-bit scalar quantizer
- ✓ 24-bit scalar quantizer
- ✓ 10-bit vector quantizer

Extensions of Vector Quantization

- Use of Multiple Codebooks : Codebooks developed from different spectral representations of speech (Ex: LPCCs, MFCCs, PLPs, RASTA-PLPs, etc…)

- Binary search trees with sub-optimal VQs (M vs log(M))

- K-tuple quantizers
  - ✓ Adv: vowel-like sounds (exploiting the correlations)
  - ✓ Dis-adv: transcient sounds and unvoiced consonants

- Matrix quantization : Codebook of sounds/words of variable sequence length is created.
  - ✓ Word Recognition Systems

- Trellis codes : Time-sequential dependencies among codebook entries are explicitly determined ($\mathrm{v_n} \rightarrow y_l$ $then$ $v_{n+1} \rightarrow subset\ of\ codebook\ entries\ related\ to\ y_l$)

- Hidden Markov Model : Time and Spectral constraints are used to quantize the entire speech utterance