- Read the "Adaptive Quantization" from book.

Vector Quantization (VQ)

- Quantization of vector sequences.
- From pure quantization perspective, the best performing case is a VQ. In fact, VQ with high dimension is good.
- Speech is a scalar source. But we can buffer some L scalars together and get a vector source.
- However, if there is two/Multiple speakers. Then anyway, if multiple microphones are there, the ~~cs~~ collected signal is a vector signal.
- What VQ does?
  - best correlation exploitation
    (The fundamental requirement of a source coding method)

$$\underline{x} = (x_1, x_2, \cdots, x_L)^T$$

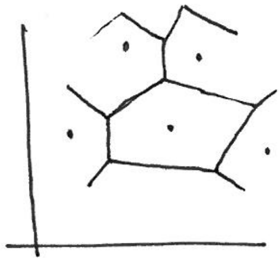
$\underline{x}$ is an L-dimensional source vector.
$\underline{\hat{x}}$ is its coded value (quantized value).

We now skip $\_$ notation.

- x has to quantized to a representation vector by comparing with all possible representation vectors.
- The set of representation vectors are called a codebook.

Quantization region :

- Voronoi region
- Boundary.



See Figure 7.14 of the book.

- Let the set of ~~code book~~ representation vectors (kept in the codebook) $\{\hat{\underline{x}}_i\}_{i=1}^{K}$, $K = 2^{\omega}$

  ω bits/vector

- Quantization task : Replace an input vector $\underline{x}$ by most similar vector $\hat{\underline{x}} = \hat{\underline{x}}_{i_{opt}}$ from the codebook.

  The choice is made by a distance comparison.

$$d(x, \hat{x}_{i_{opt}}) = \min_i d(x, \hat{x}_i).$$

- By this distance comparison, boundary between two Voronoi cells are implicitly defined.

- Through channel, only i_{opt} is sent.

- ω bits / vector ; K representation vectors (or codes/quantization values) in the codebook ; L dimensional vector ; $\bar{\omega}$ bits/scalar $= \frac{K}{L} = \frac{2^{\omega}}{L}$.

  A typical example : $K = 2^{10} = 1024$ and $L = 40$ (typical dimension for error/residual signal coding in LP speech coder).

  So, $\frac{10}{40} = \frac{1}{4}$ bits/scalar.

- Important issue : fractional bits/scalar is possible. SQ can not do that.

Choice of Distance Measure.

quantization ~~signal~~ error (vector) $e = \hat{x} - x$.

e decides the perceptual speech quality.

- Squared distance

$$d(\underline{x}, \hat{\underline{x}}_i) = \frac{1}{L} (\underline{x} - \hat{\underline{x}}_i)^T (\underline{x} - \hat{\underline{x}}_i).$$

- Weighted Square distance

$$d(\underline{x}, \hat{\underline{x}}_i) = \frac{1}{L} (\underline{x} - \hat{\underline{x}}_i)^T \underline{W}_i (\underline{x} - \hat{\underline{x}}_i)$$

  - $\underline{W}_i$ is a positive-definite matrix of $L \times L$ size.
  - In general $\underline{W}$ is $\underline{W}(\underline{x}) \longrightarrow$ depends on the input signal vector.

- An example of $d(\underline{x}, \hat{\underline{x}})$ in speech LPCs quantization.

Itakura - Saito Measure

$$d(\underline{x}, \hat{\underline{x}}_i) = \frac{(\underline{x} - \hat{\underline{x}}_i)^T R^{(n+1)} (\underline{x} - \hat{\underline{x}}_i)}{\underline{x}^T R^{(n+1)} \underline{x}}$$

$$\underline{x} = (1, -a_1, -a_2 \cdots, -a_n)^T$$

$$\hat{\underline{x}}_i = (1, -\hat{a}_{i1}, -\hat{a}_{i2} \cdots, -\hat{a}_{in})^T$$

$\{a_j\}_{j=1}^{n}$ are n LPCs.

Complexity problem:

- Exponential search and storage complexity.
(memory)

Example: $\bar{\omega}$ bits/scalar.

$$\omega = \bar{\omega} . L \quad \text{(bits/vector)}$$

$$K = 2^{\omega} \quad \text{(codebook size)}$$

$$= 2^{\bar{\omega} . L}$$

For a fixed $\bar{\omega}$, we will always get an coding performance improvement by increasing L.

Codebook size increases exponentially with L.

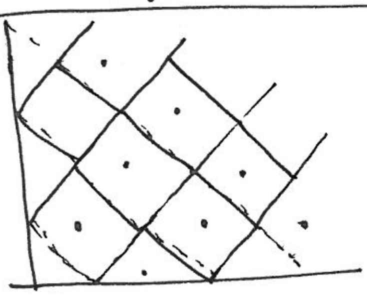
Lets say $\bar{\omega} = 1$. (1 bit/scalar).

$L = 10$, $K = 1024$

$L = 100$ $K = 2^{100}$ (its a huge number).

- So, use of large VQ problem with unstructured codebook is complexity limited.

Lattice Quantization (structured VQ)



D_2 Lattice

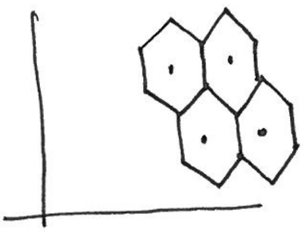
(Figure 7.14 in book)

- Using generating basis vectors, we can always represent a code (the centroid).
- So no need to store all the codes and search complexity is very low.

Q: What is the optimum Lattice in 2D ?

Optimum → least average SED.

Hexagon



Honey bee

- In Mathematics: This is a great theoretical challenge to find the optimal lattice in different dimensions.

Design of optimal vector codebooks.

Linde – Buzo – Gray (LBG)

Algorithm 1

- Codebook (initial) is a random codebook.
- Lot of training vectors $\{\underline{x}\}$.
- Steps

Step 0 : (a) choose a start codebook consisting of K random vectors $\hat{x}_i$ (or uniform lattice vectors) of dimension L.

(b) Set $m = 1$.

Step 1 : (a) Quantize the training sequence $\underline{x}(k)$, $k = 1, 2 \cdots, K_T$, with $K_T \gg K$ and compute the average distortion

$$D_m = \frac{1}{K_T} \sum_{k=1}^{K_T} d\left(\underline{x}(k), \hat{x}_{i\,opt}\right).$$

(b) Terminate if D_m has no significant reduction w.r.t. D_{m-1}.

$$\frac{|D_{m-1} - D_m|}{D_m} < \epsilon.$$

Step 2 : (a) Replace the old code vectors $\{\hat{x}_i\}_{i=1}^{K}$ by the centroids of those training vectors $\underline{x}(k)$ which have been allocated to the old vectors $\hat{\underline{x}}_i$

(b) $m = m+1$. Go to step 1.

Aspects :
- does not allow global minimum of D_m. So, no globally optimum codebook.
- provides locally optimum solution.
- Initial codebook is a big factor.

Algorithm 2 :

By "splitting"

Explain in the class by figure drawing.

Gain - Shape Quantization

- In speech signal, we can have same sound, but with different loudness.
- Same signal shapes can occur with different amplitudes.
- Hence, the codebook should contain representation vectors $\{\hat{x}_i\}$ with the same shape and a different gain.
- So, $\hat{x}$ = Q.gain $\times$ Q. shape.

=

SED (square Euclidean distance)

$$d_i = d(\underline{x}, g_i \hat{\underline{x}}_i) = \frac{1}{L} \| \underline{x} - g_i \hat{\underline{x}}_i \|_2^2$$

what we need to solve? $\{g_i, \hat{x}_i\}$.

- For a given $\hat{x}_i$, choose g_i.

$$\frac{\partial d_i}{\partial g_i} = 0$$

or, $$2(x - g_i \hat{x}_i)^T \hat{x}_i = 0$$

or, $$x^T \hat{x}_i = g_i \hat{x}_i^T \hat{x}_i$$

or, $$g_{i,opt} = \frac{x^T \hat{x}_i}{\|\hat{x}_i\|^2}.$$

- Now we know g_i and then choose $\hat{x}_i$ which minimizes

$$d_i = \frac{1}{L} \| x - g_{i,opt} \hat{x}_i \|^2$$

$$= \frac{1}{L} \left\| x - \frac{x^T \hat{x}_i}{\|\hat{x}_i\|^2} \hat{x}_i \right\|^2$$

$$\hat{x}_{i,opt} = \min_i d_i .$$

- Note a trouble. $g_{i,opt}$ is dependent ~~of~~ on x. So it is analogue and need to be ~~quantized~~ quantized.

We must need to have $\hat{g}_{i,opt}$, $\hat{x}_{i,opt}$ to get $\hat{x}_{opt} = \hat{g}_{i,opt} \cdot \hat{x}_{i,opt}$.