Ch. 10 Vector Quantization

Overview
Motivation

Recall in Lossless Methods:

- Block Huffman is better than single-symbol Huffman
  - Blocks allow to exploit correlation between symbols (assuming source symbols are not independent!)
- Shannon proved that “blocking taken to the limit” achieves optimal compression… exploits correlation

Recall in Scalar Quantization:

- It is the lossy version of a single-symbol method
- Shannon also proved that for lossy we can achieve the theoretical bound on compression (R-D curve) via “blocking taken to the limit”

This blocking idea motivates Vector Quantization
Main Idea of VQ

That info theory says to consider “blocking” to exploit correlation

Group into vectors (non-overlapping) and “quantize” each vector $x[n]$

ACF of Speech

Samples in vector are highly correlated!

No Need to Limit to 2-D Vectors
Illustration of Gain from VQ: Real Speech Data

- 100 cells smaller than SQ
  - Same Rate, Lower Distortion
- Fewer than 100 cells of same size as SQ
  - Lower rate, Same Distortion
Going to higher dimension: vectors concentrated in smaller % of whole space ➔ Improved Performance

Scatter plot for 3-Dimensional vectors

Same speech signal…
3-D VQ
Forming Vectors

For time signals… we usually form vectors from temporally-sequential samples.

For images… we usually form vectors from spatially-sequential samples.

But… you should let the ACF structure guide your choice…

Note: Book uses upper-case italic to indicate a vector… I use the more standard lower-case bold non-italic…

\[
\begin{align*}
\mathbf{x}_1 &= [x[0] \ x[N] \ x[2N] \ x[3N]] \\
\mathbf{x}_2 &= [x[1] \ x[N+1] \ x[2N+1] \ x[3N+1]] \\
\mathbf{x}_3 &= [x[2] \ x[N+2] \ x[2N+2] \ x[3N+2]]
\end{align*}
\]
Designing VQ

Q: What does it mean to design a VQ?
A: Similar to SQ… Specify Decision Boundaries and Reconstruction Values… except now those are in \(N\)-D space… Goal is to minimize MSQE for a given rate (or vice versa)

**Example of a VQ Design:** For 2-D vectors taken from a sequence of independent Gaussian samples…

Somehow we need to design the DBs and RLs…
However, we really only need to specify the RLs…
Then the \(i^{th}\) Decision Region consists of all points closer to the \(i^{th}\) RL than they are to any other RL
We’ll see how to do this later…

Figure from book *Vector Quantization and Signal Compression* by Gersho & Gray
Structure of a VQ

Binary Codes
\[ \lceil \log_2 M \rceil \text{ bits} \]
1 code per code vector

*ML-Dim Code Vectors*

i.e. Reconstruction Points

L-Dim. Vectors

Fig. 10.1 in Textbook
Example

\[ x[n]: \ldots 0.75 \ 1.27 \ 1.78 \ 2.11 \ldots \]
Encoder & Decoder Operation

**Encoder:** *Search* codebook for code vector $y_j$ that is closest to input vector $x$.

$$\text{Codebook } \{y_i\}_{i=1}^{M} = C \quad \text{where each } y_i \in \mathbb{R}^L$$

$x$ is closest to $y_j$ if:

$$\|x - y_j\| \leq \|x - y_i\| \quad \forall y_i \in C$$

where

$$\|z\| = \sqrt{\sum_{i=1}^{L} z_i^2} \quad \text{(Euclidian Norm)}$$

for $z = \begin{bmatrix} z_1 & z_2 & \cdots & z_L \end{bmatrix}^T$

**Decoder:** Use received binary codeword (the “index”) as an address into the codebook (i.e., Table Look-Up)

**Notation:** If $x$ is closest to $y_j$ we write $Q(x) = y_j$

**Note:** VQ cells are defined by the RLs & these equations rather than explicit boundaries!!!
Complexity of VQ

Encoder is Computationally Complex

Must check all $M y_i$ for closeness… must compute $M$ norms

$M$ can be quite large: 256, 512, 1024, etc.

Decoder is Easy & Fast

VQ Complexity is Asymmetrical

- May not work well for Real-Time Encoding
- But Real-Time Decoding is very easy
VQ Rate & Performance vs. Rate

If $L$-D vectors are quantized using a VQ having $M$ reconstruction points we need binary codewords having

$$\text{codeword length} = \left\lceil \log_2 M \right\rceil \text{ bits/vector}$$

$$\frac{\text{# bits/sample}}{L} = \left\lceil \log_2 M \right\rceil \text{ bits/sample}$$

“Typical” $L$ values:
- Images: $L = 16$ (4x4)
- Speech: $L = 3, 4, 5, 6$

**Info Theory Says:** Increasing $L$ improves the VQ performance

**Practice Says:** … only up to a point!
- Improvement decreases w/ increasing $L$
- Design gets harder w/ increasing $L$
- Encoder complexity grows w/ increasing $L$
Comparison of VQ Results with Published Results (Speech)

Solid Line: Published Results

- o: Signal S1
- x: Signal S2
- *: Signal S3

VQ Dimension: \( L \)

Post-Quantization SNR (dB)

Design was not achieved

L = 5

2 bits/sample