Coding with Distortion

\[ X(t) \]

\[ \text{1011010} \]

\[ \text{D/A CONVERTER} \]

\[ \text{SOURCE DECODER} \]

\[ \text{INFO SINK} \]

\[ \text{To Noisy Channel (Using Error Correcting Code)} \]

\[ \text{1011010...} \]

\[ \text{From Noisy Channel (Using Error Correcting Code)} \]

\[ \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} E(X(t) - \hat{X}(t))^2 \, dt = \text{M.S.E.} \]

If signals are bandlimited, one can sample at Nyquist rate and convert continuous-time problem to discrete-time problem. This sampling is part of the A/D CONVERTER.

\[ X_1, X_2, \ldots, X_N \]

\[ \text{SOURCE ENCODER} \]

\[ \text{SOURCE DECODER} \]

\[ \hat{X}_1, \hat{X}_2, \ldots, \hat{X}_N \]

\[ \lim_{m \to \infty} \frac{1}{m} \sum_{k=1}^{m} E(X_k - \hat{X}_k)^2 = E^2 \]
A/D Conversion and D/A Conversion

Assume a random variable $X$ which falls into the range $(X_{min}, X_{max})$ to be converted into $k$ binary digits. Let $M = 2^k$. The usual A/D converter first subdivide the interval $(X_{min}, X_{max})$ into $M$ equal sub-intervals of width $\Delta = (X_{max} - X_{min}) / M$ as shown below.

For the case of $k = 3$ and $M = 8$, we call the $i$th sub-interval, $R_i$.

\[
R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8
\]

\[
X_{min}, X_{max}
\]

Assume that if $X$ falls in the region $R_i$ ($x \in R_i$), then the D/A converter uses as an estimate of $X$, the value $\hat{x} = \gamma_i$ of the center of the $i$th region. Then the mean-squared error between $X$ and $\hat{x}$ is

\[
\varepsilon^2 = \mathbb{E}[(X - \hat{x})^2] = \int_{X_{min}}^{X_{max}} (X - \gamma_i)^2 f_X(x) \, dx
\]

where $f_X(x)$ is the probability density function of the random variable $X$.

Let $f_{X|R_i}(x)$ be the conditional density function of $X$ given that $X$ falls in the region $R_i$. Then

\[
\varepsilon^2 = \sum_{i=1}^{M} P[x \in R_i] \int_{R_i} (x - \gamma_i)^2 f_{X|R_i}(x) \, dx
\]

Note that

\[
\sum_{i=1}^{M} P[x \in R_i] = 1
\]

and

\[
\int_{X_{min}}^{X_{max}} f_{X|R_i}(x) \, dx = 1 \quad \text{for } i = 1, 2, \ldots, M
\]
Now make the further assumption that \( b \) is large enough so that \( f_{X | R_i}(x) \) is a constant over the region \( R_i \). Then \( f_{X | R_i}(x) = \frac{1}{\Delta} \) for all \( i \).

and 
\[
\int_{x \in R_i} (x - y_i)^2 f_{X | R_i}(x) \, dx = \frac{1}{\Delta} \int_{a}^{b} (x - \left(\frac{b-a}{2}\right))^2 \, dx = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} (x - 0)^2 \, dx
\]

\[
= \frac{1}{\Delta} \cdot \frac{2}{3} \left(\frac{\Delta}{2}\right)^3 = \frac{\Delta^2}{12}
\]

Then \( E^2 = \sum_{i=1}^{M} \mathbb{P}(x \in R_i) \cdot \frac{\Delta^2}{12} = \frac{\Delta^2}{12} \).

If \( X \) has variance \( \sigma_x^2 \), the signal-to-noise ratio of the A to D (A/D) converter is often defined as \( \left( \frac{\sigma_x^2}{\Delta^2/12} \right) \).

If \( X_{\min} \) is equal to \(-\infty\) and/or \( X_{\max} = +\infty \), then the least and first
intervals can be infinite in extent. However, \( f_{X | x} \) is usually small
enough in those intervals so that the result is still approximately
the same.
SCALAR QUANTIZATION OF (GAUSSIAN) SAMPLES.

USUAL SCALAR QUANTIZATION (3-BINARY DIGITS/SAMPLE)

**ENCODER**

<table>
<thead>
<tr>
<th>$x = -3b$</th>
<th>000</th>
<th>$0 &lt; x &lt; b$</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2b &lt; x &lt; -b$</td>
<td>001</td>
<td>$b &lt; x &lt; 2b$</td>
<td>101</td>
</tr>
<tr>
<td>$-b &lt; x &lt; 0$</td>
<td>010</td>
<td>$2b &lt; x &lt; 3b$</td>
<td>110</td>
</tr>
<tr>
<td>$0 &lt; x &lt; b$</td>
<td>111</td>
<td>$3b &lt; x$</td>
<td>111</td>
</tr>
</tbody>
</table>

**DECODER**

<table>
<thead>
<tr>
<th>000</th>
<th>-3.5b</th>
<th>100</th>
<th>4.5b</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>-3.5b</td>
<td>101</td>
<td>4.5b</td>
</tr>
<tr>
<td>010</td>
<td>-1.5b</td>
<td>110</td>
<td>3.5b</td>
</tr>
<tr>
<td>011</td>
<td>-0.5b</td>
<td>111</td>
<td>2.5b</td>
</tr>
</tbody>
</table>
\[ b_{i-1} \leq x < b_i \rightarrow x = a_i, \quad i = 1, 2, \ldots, M \]

\[ b_0 = -\infty, \quad b_M = +\infty \]

\[ a_1, \quad a_2, \quad a_3, \quad \ldots, \quad a_M \]

\[ \frac{x}{b_0} - \frac{b_1}{b_1} = \frac{x}{b_0} - \frac{b_1}{b_0} = \frac{b_1}{b_0} \]

\[ b_0 = -\infty \quad b_1 \quad b_2 \quad b_3 \quad b_4 \quad \ldots \quad b_{M-1} \quad b_M = +\infty \]

Optimize \( \{b_i\} \) and \( \{a_i\} \) to minimize \( \varepsilon^2 \)

\[
\varepsilon^2 = \sum_{i=1}^{M} \left( \int_{b_{i-1}}^{b_i} (x-a_i)^2 f_x(x) \, dx \right)
\]

\[
\frac{\partial \varepsilon^2}{\partial a_j} = 0 \quad \frac{\partial \varepsilon^2}{\partial b_j} = 0.
\]

Use Leibnitz's Rule

\[
\frac{\partial}{\partial t} \int_{a(t)}^{b(t)} f(x, \varepsilon) \, dx = \left[ f(b(t), \varepsilon) \frac{\partial b(t)}{\partial \varepsilon} \right]_{a(t)}^{b(t)}
\]

\[
- \left[ f(a(t), \varepsilon) \frac{\partial a(t)}{\partial \varepsilon} \right]_{a(t)}^{b(t)}
\]

\[
+ \int_{a(t)}^{b(t)} \frac{\partial}{\partial \varepsilon} f(x, \varepsilon) \, dt.
\]
\[
\frac{\partial}{\partial b_j} \left( \sum_{i=1}^{m} \int_{b_{j-1}}^{b_j} (x-a_i) f(x) dx \right) = \\
\frac{\partial}{\partial b_j} \int_{b_{j-1}}^{b_j} (x-a_j)^2 f_X(x) dx + \frac{\partial}{\partial b_j} \int_{b_j}^{b_{j+1}} (x-a_{j+1})^2 f_X(x) dx \\
= (b_j - a_j)^2 f_X(x) \bigg|_{x = b_j}^{x = b_j} - (b_j - a_{j+1})^2 f_X(x) \bigg|_{x = b_j}^{x = b_j} \\
= b_j^2 - 2a_jb_j + b_j^2 - b_j^2 - 2a_jb_{j+1} + a_{j+1}^2 \\
2b_j(a_{j+1} - a_j) = a_{j+1}^2 - a_j^2 \\
\left\lbrack b_j = \frac{a_{j+1} + a_j}{2} \right\rbrack \quad (I) \\
\frac{\partial}{\partial a_j} \left( \sum_{i=1}^{m} \int_{b_{j-1}}^{b_j} (x-a_i) f(x) dx \right) = -2 \int_{b_{j-1}}^{b_j} (x-a_j) f_X(x) dx = 0 \\
a_j \int_{b_{j-1}}^{b_j} f_X(x) dx = \int_{b_{j-1}}^{b_j} x f_X(x) dx \\
a_j = \frac{\int_{b_{j-1}}^{b_j} x f_X(x) dx}{\int_{b_{j-1}}^{b_j} f_X(x) dx} \quad (II)
\]
Note that the $b_k$ can be found from (I) once
the $a_{k,l}$ are known. (The $b_k$ are the midpoints
of the $a_{k,l}$.)

And the $a_{k,l}$ can be solved from (II) once the
$b_k$ are known. (The $a_{k,l}$ are the centroids
of the corresponding regions.)

Thus one can use a computer to iteratively solve
for the $a_{k,l}$ and the $b_k$.

1. One starts with an initial guess for the $b_k$.
2. One uses (II) to solve for the $a_{k,l}$.
3. One uses (I) to solve for the $b_k$.
4. One repeats steps 2 and 3 until the $a_{k,l}$
   and the $b_k$ "stop changing."

Comments

1. This works for any $f_X(x)$
2. If $f_X(x)$ only has a finite support one
   adjusts $b_0$ & $b_n$ to be the limits of the
   support.
3. For a Gaussian, one needs to know
   \[ \int_{-\infty}^{b_0} f_X(x) \, dx \text{ and } \int_{b_n}^{\infty} f_X(x) \, dx. \]
\[ \int_{-\infty}^{\rho} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} x^2} \, dx = Q(\rho) - Q(\alpha) \]
\[ \int_{\alpha}^{\rho} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} x^2} \, dx = \ldots \quad (\text{integrate by parts}) \]
(or let \( y = x^2 \))

4. If \( M = 2^a \) one could use "a" binary digits to represent the quantized value. However, since the quantized values are not necessarily equally likely, one could use a Huffman code to use fewer binary digits (on the average).

5. After the \( \{a_i\} \) and \( \{b_n\} \) are known, one computes \( \varepsilon^2 \) from
\[ \varepsilon^2 = \sum_{i=1}^{N} \int_{b_{i-1}}^{b_i} (x-a_i)^2 f_X(x) \, dx \]

6. For \( M = 2 \) and \( f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2} \frac{x^2}{\sigma^2}} \) one can easily show that:
\( b_0 = -\infty \), \( b_1 = 0 \), \( b_2 = +\infty \),
\( a_2 = -a_1 = \sqrt{\frac{2}{\pi} \sigma^2} \)
\[ \varepsilon^2 = (1 - \frac{2}{\pi}) \sigma^2 = 0.369 \sigma^2 \]
VECTOR QUANTIZATION

One can achieve a smaller $e^2$ by quantizing several samples at a time.

We would then use regions in an $n$-dimensional space.

The rate-distortion formula tells us how small $e^2$ can be as $n \to \infty$.

For a Gaussian with one binary digit per sample, $E^2 \geq \frac{5^2}{4} = (0.25)\sigma^2$

This follows from the result on the next page.
**Discrete-Time Gaussian Source**

Let source produce i.i.d. Gaussian samples $x_1, x_2, \ldots$

where $f_X(x) = \frac{1}{\sqrt{2\pi}\sigma^2} e^{-\frac{x^2}{2\sigma^2}}$

Let source encoder produce a sequence of binary digits at a rate of $R$ binary digits/symbol.

In our previous terminology $R = k$

Let the source decoder produce the sequence $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_k$ such that the mean-squared error between $\{x_n\}$ and $\{\hat{x}_n\}$ is $\epsilon^2$.

Then one can prove that for any such system

$$R \geq \frac{1}{2} \log_2 \left( \frac{\sigma^2}{\epsilon^2} \right) \text{ for } \epsilon^2 \leq \sigma^2.$$  

($R = 0$ for $\epsilon^2 > \sigma^2$)

This is an example of "Rate-Distortion Theory."

Noted that for $R = k = 1$,

$$1 \geq \frac{1}{2} \log_2 \left( \frac{\sigma^2}{\epsilon^2} \right)$$

$$2 \geq \log_2 \left( \frac{\sigma^2}{\epsilon^2} \right)$$

$$4 \geq \frac{\sigma^2}{\epsilon^2}$$

or $\epsilon^2 \leq (4/9)\sigma^2$. 

$\epsilon^2 \leq (4/9)\sigma^2$.
Reduced Fidelity Audio Compression

MP3 players use a form of audio compression called MPEG-1 Audio Layer 3. It takes advantage of psycho-acoustic phenomena whereby a loud tone at one frequency "masks" the presence of softer tones at neighboring frequencies. Thus, these softer neighboring tones need not be stored (or transmitted).

Compression efficiency of a audio compression scheme is normally described by the encoded bit rate (prior to the introduction of coding bits.) The CD has a bit rate of \((44.1 \times 10^3 \times 2 \times 16) = 1.41 \times 10^6\) bits/second. The term \("44.1 \times 10^3\)" is the sampling rate which is approximately the Nyquist frequency of the audio to be compressed. The term "2" comes from the fact that there are two channels in a stereo audio system. The term "16" comes from the 16-bit (or \(2^{16} = 65,536\) level) A to D converter. (A slightly higher sampling rate \(48 \times 10^3\) samples/second is used for a DAT recorder.)

Different standards are used in MP3 players. Several bit rates are specified in the MPEG-1, Layer 3 standard. These are 32, 40, 48, 56, 64, 80, 96, 112, 128, 144, 160, 192, 224, 256 and 320 kilobits/second. The sampling rates allowed are 32, 44.1, and 48 kilohertz but the sampling rate of \("44.1 \times 10^3\) Hz is almost always used.
The basic idea behind the scheme is as follows. A block of 576 time domain samples are converted into 576 frequency domain samples using a DFT. The coefficients are then modified using psycho-acoustic principles. The processed coefficients are then converted into a bit stream using various schemes including Huffman encoding. The process is reversed at the receiver: bits $\rightarrow$ frequency domain coefficients $\rightarrow$ time-domain samples.