The Lempel-Ziv algorithm matches the longest string of yet unencoded symbols with strings starting in the window.

Window size w is large power of 2, maybe 2^{17}.

$\log w$ **bits to encode** u, $2\lfloor \log n \rfloor + 1$ **bits for** n

$$w = \textbf{window} \qquad P$$

$$\underline{\textbf{Match}} \qquad n = 3$$

$$b \ \ c \ \ d \ \ a \ \ c \ \ b \ \ a \ \ b \ \ a \ \ c \ \ d \ \ b \ \ c \ | \ a \ \ b \ \ a \ \ b \ \ d \cdots$$

$$u = 7$$
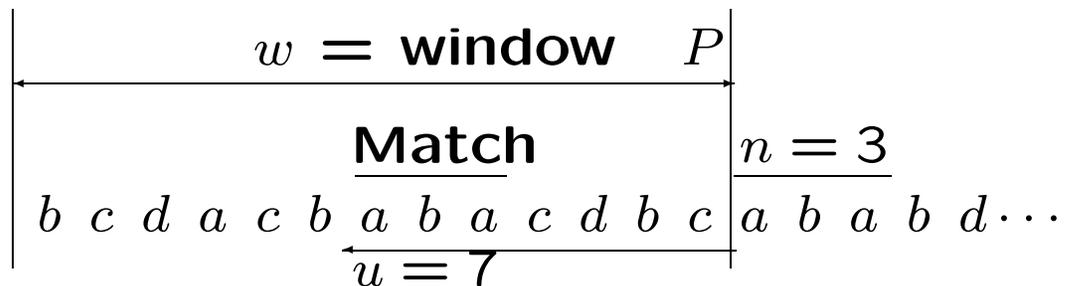
Slightly flaky argument about optimality for ergodic Markov source:

From WLLN, relative frequency of x_{P+1}^{P+n} in very long window is $\approx w \Pr[x_{P+1}^{P+n}]$.

For typical x_{P+1}^{P+n}, this is $\approx w 2^{-nH[X|S]}$.

We expect many appearances of x_{P+1}^{P+n} in the window (high probablity of match of size n or more) if $n < (\log w)/H[X|S]$

Bits per step is

$$(l(n) + \log w)/n \approx \frac{2 \log n}{n} + H[X|S]$$

QUANTIZATION

input waveform → **sampler** → **quantizer** → **discrete encoder** → **reliable binary channel**

analog sequence

symbol sequence

output waveform ← **analog filter** ← **table lookup** ← **discrete decoder** ←
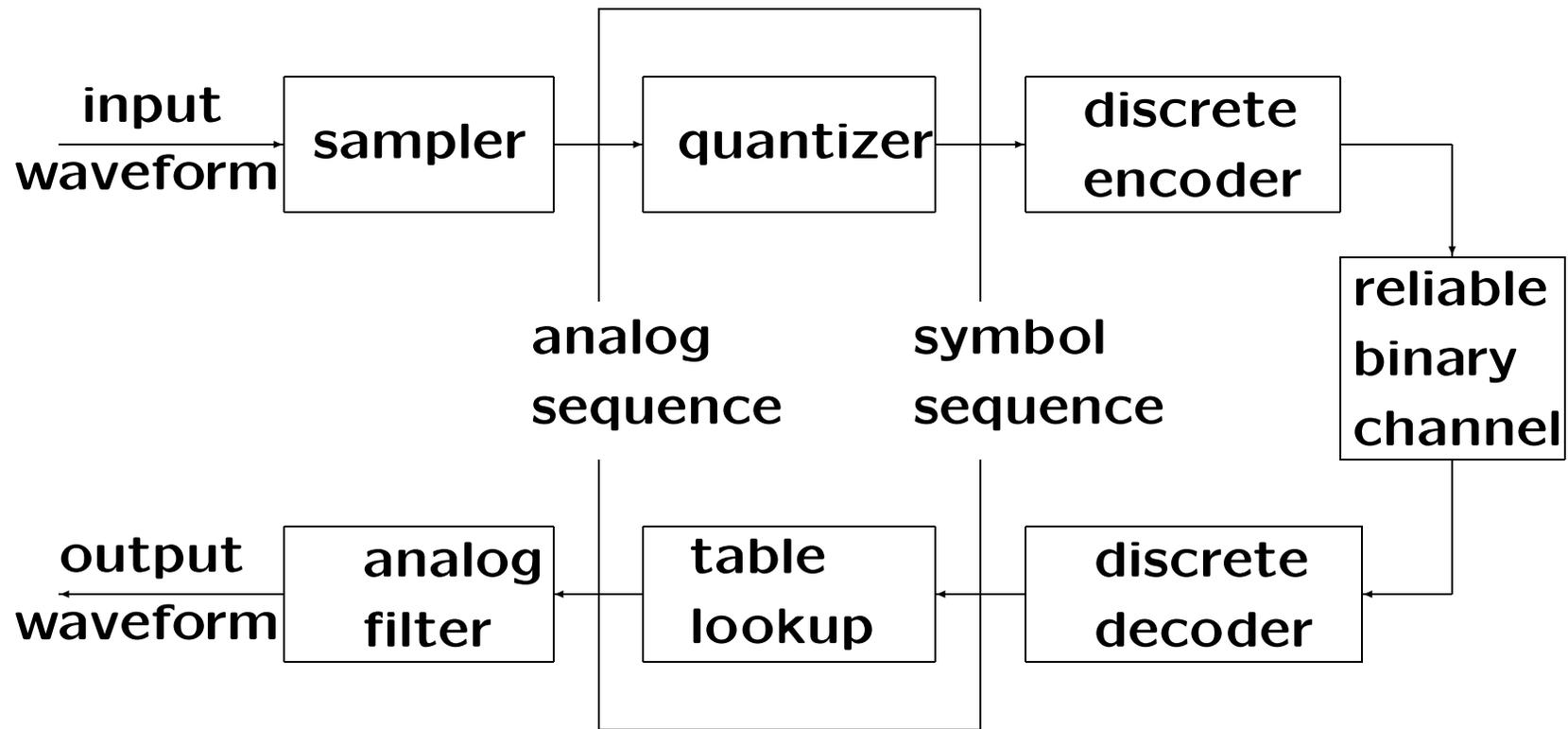
Converting real numbers to binary strings requires a mapping from $\mathbb{R}$ to a discrete alphabet.

This is called <u>scalar quantization</u>.

Converting real n-tuples to binary strings requires mapping $\mathbb{R}^n$ to a discrete alphabet.

This is called <u>vector quantization</u>.

Scalar quantization encodes each term of the source sequence separately.

Vector quantization segments source sequence into n-blocks which are quantized together.

A scalar quantizer partitions $\mathbb{R}$ into M regions $\mathcal{R}_1, \dots, \mathcal{R}_M$.

Each region $\mathcal{R}_j$ is mapped to a symbol a_j called the representation point for $\mathcal{R}_j$.

$$\begin{array}{c}
\overset{b_1}{} \quad \overset{b_2}{} \quad \overset{b_3}{} \quad \overset{b_4}{} \quad \overset{b_5}{} \\
\longleftarrow \mathcal{R}_1 \dashv\vdash \mathcal{R}_2 \dashv\vdash \mathcal{R}_3 \dashv\vdash \mathcal{R}_4 \dashv\vdash \mathcal{R}_5 \dashv\vdash \mathcal{R}_6 \longrightarrow \\
a_1 \qquad a_2 \qquad a_3 \qquad a_4 \qquad a_5 \qquad a_6
\end{array}$$
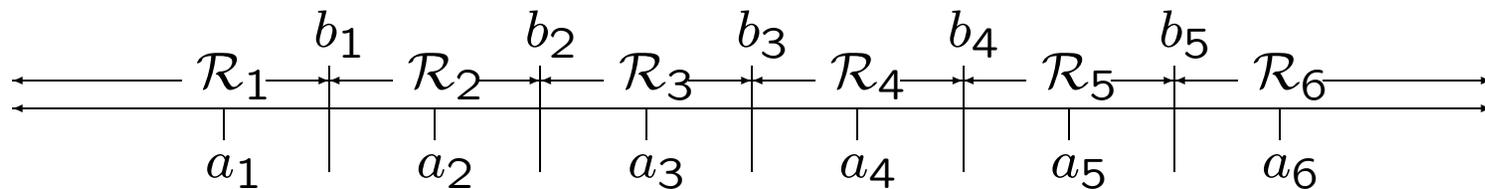
Each source value $u \in \mathcal{R}_j$ is mapped into the same representation point a_j.

After discrete coding and channel transmission, the receiver sees a_j and the distortion is $u - a_j$.

View the source value u as a sample value of a random variable U.

The representation a_j is a sample value of the rv V where V is the quantization of U.

That is, if $U \in \mathbb{R}_j$, then $V = a_j$.

The source sequence is $U_1, U_2, \ldots$. The representation is $V_1, V_2, \ldots$ where if $U_k \in \mathbb{R}_j$, then $V_k = a_j$.

Assume that $U_1, U_2, \ldots$ is a memoryless source which means that $U_1, U_2, \ldots$ is iid.

For a scalar quantizer, we can look at just a single U and a single V.

We are almost always interested in the mean square distortion of a scalar quantizer

$$\mathsf{MSE} = \mathsf{E}[(U - V)^2]$$

Interesting problem:

For given probability density $f_U(u)$ and given alphabet size M, choose $\{\mathbb{R}_j, 1 \leq j \leq M\}$ and $\{a_j, 1 \leq j \leq M\}$ to minimize MSE.

Subproblem 1: Given representation points $\{a_j\}$, choose the regions $\{\mathcal{R}_j\}$ to minimize MSE.

This is easy: for source output u, squared error to a_j is $|u - a_j|^2$.

Minimize by choosing closest a_j.

$$\mathcal{R}_1 \quad \overset{b_1}{|} \quad \mathcal{R}_2 \quad \overset{b_2}{|} \quad \mathcal{R}_3 \quad \overset{b_3}{|} \quad \mathcal{R}_4 \quad \overset{b_4}{|} \quad \mathcal{R}_5 \quad \overset{b_5}{|} \quad \mathcal{R}_6$$

$$a_1 \qquad a_2 \qquad a_3 \qquad a_4 \qquad a_5 \qquad a_6$$
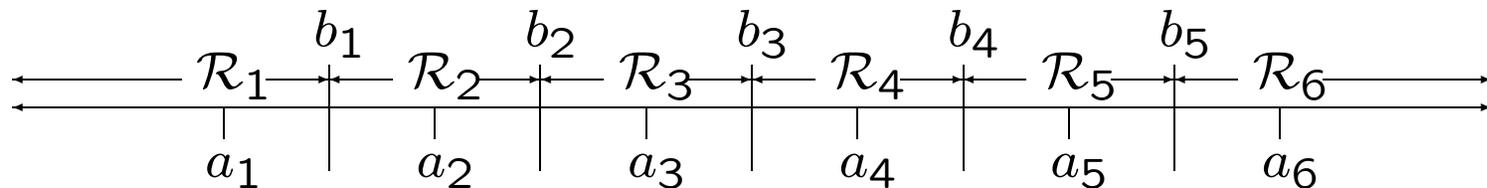
Thus $\mathcal{R}_j$ is region closer to a_j than any $a_{j'}$.
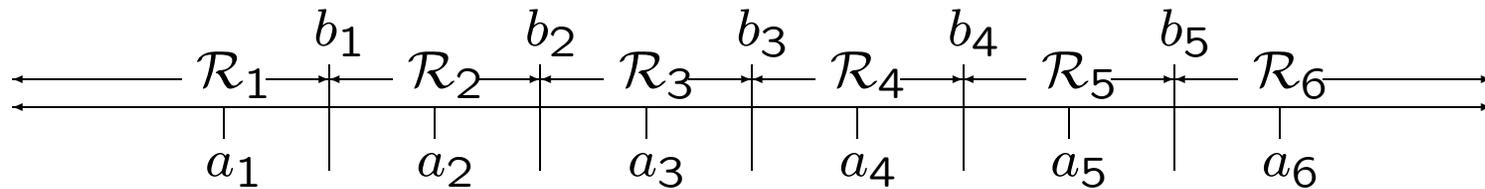
$\mathcal{R}_j$ **is bounded by**

$$
\begin{aligned}
b_{j-1} &= \frac{a_j + a_{j-1}}{2} \\
b_j &= \frac{a_j + a_{j+1}}{2}
\end{aligned}
$$

MSE regions must be intervals and region separators must lie midway between representation points in any minimum MSE scalar quantizer.

Subproblem 2: Given interval regions $\{\mathcal{R}_j\}$, choose the representation points $\{a_j\}$ to minimize MSE.

$$\mathsf{MSE} = \int_{-\infty}^{\infty} f_U(u)(u - v(u))^2 \, du$$

$$= \sum_{j=1}^{M} \int_{\mathcal{R}_j} f_U(u) \left(u - a_j\right)^2 \, du.$$

Given $U \in \mathcal{R}_j$, the conditional density of U is $f_j(u) = f_U(u)/Q_j$ for $u \in \mathcal{R}_j$ where $Q_j = \Pr(U \in \mathcal{R}_j)$.

Let $U(j)$ be rv with density $f_j(u)$.

$$\mathsf{E}[|U(j) - a_j|^2] = \sigma_{U(j)}^2 + |\mathsf{E}[U(j)] - a_j|^2$$

Choose $a_j = \mathsf{E}[u(j)]$.

$$\mathcal{R}_1 \quad b_1 \quad \mathcal{R}_2 \quad b_2 \quad \mathcal{R}_3 \quad b_3 \quad \mathcal{R}_4 \quad b_4 \quad \mathcal{R}_5 \quad b_5 \quad \mathcal{R}_6$$

$$a_1 \qquad a_2 \qquad a_3 \qquad a_4 \qquad a_5 \qquad a_6$$
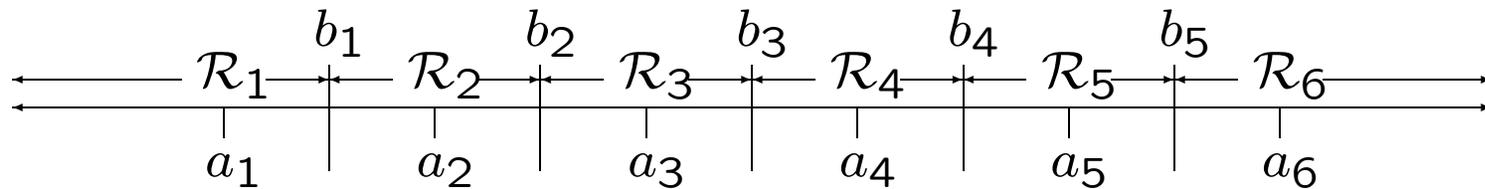
Given $\mathcal{R}_j$, a_j **must be chosen as the conditional mean of** U **within** $\mathcal{R}_j$**.**

This is another condition that must be satisfied over each interval in order to minimize MSE.

An optimal scalar quantizer must satisfy both $b_j = (a_j + a_{j+1})/2$ and $a_j = \mathsf{E}[U(j)]$.

The Lloyd-Max algorithm:

1. **choose** $a_1 < a_2 < \cdots < a_m$.

2. **Set** $b_j = (a_j + a_{j+1})/2$ **for** $1 \le j \le M - 1$.

3. **Set** $a_j = \mathsf{E}[U(j)]$ **where** $\mathcal{R}_j = (b_j - 1, b_j]$ **for** $1 \le j \le M - 1$.

4. Iterate on 2 and 3 until improvement is negligible.

The MSE is non-negative and non-increasing with iterations, so it reaches a limit.

The Lloyd-Max conditions are necessary but not sufficient for minimum MSE.

It finds local min, not necessarily global min.

$$f_U(u)$$

$$b_1$$

$$\mathcal{R}_1 \qquad \mathcal{R}_2$$

$$a_1 \qquad\qquad a_2$$
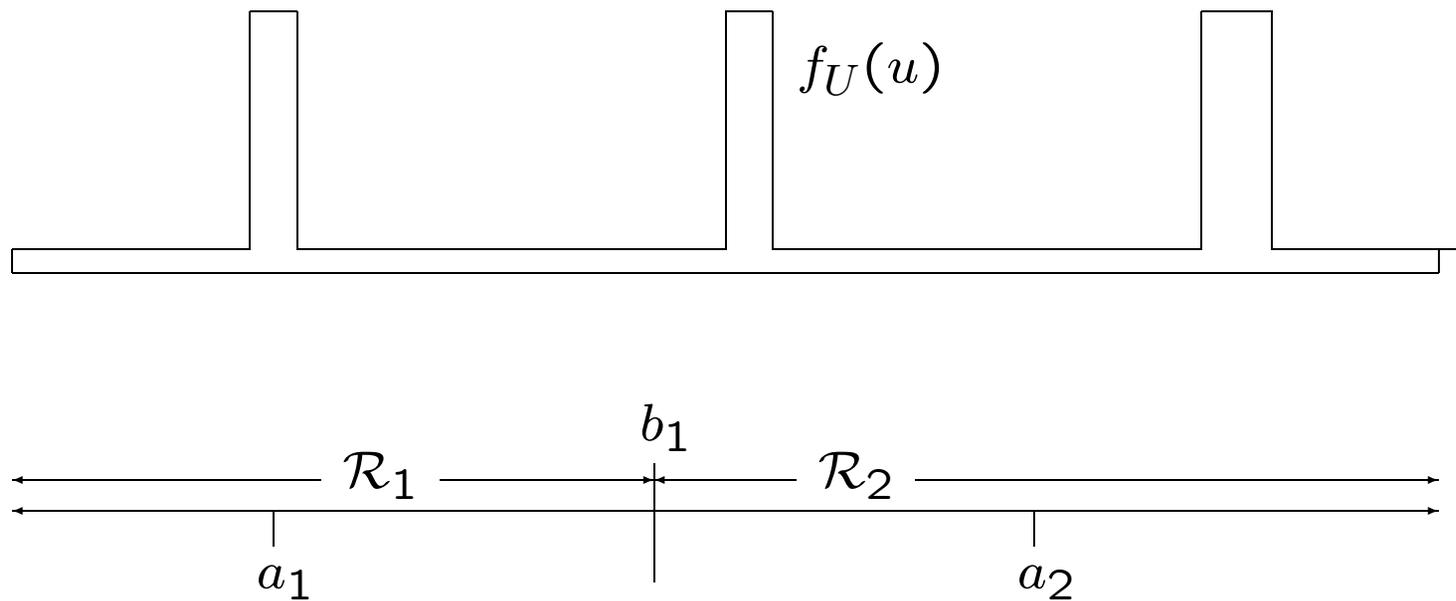
Moving b_1 to right of peak 2 reduces MSE.

VECTOR QUANTIZATION

Is scalar quantization the right approach?

Look at quantizing two samples jointly and draw pictures.

Possible approach: use rectangular grid of quantization regions.

This is really just two scalar quantizers.
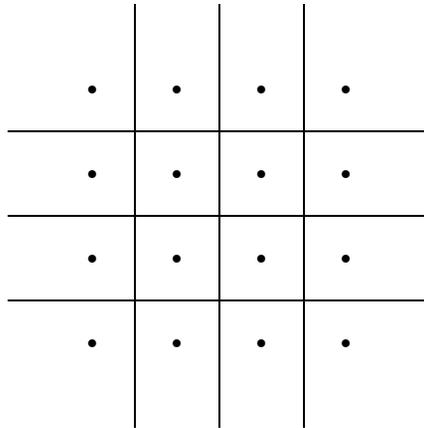
This is 2D picture of 2 uses of scalar quantizer

MSE per dimension of best vector quantizer is at least as small as best scalar quantizer.

Note that 2D region separators are perpendicular bisectors. These are called Voronoi regions.
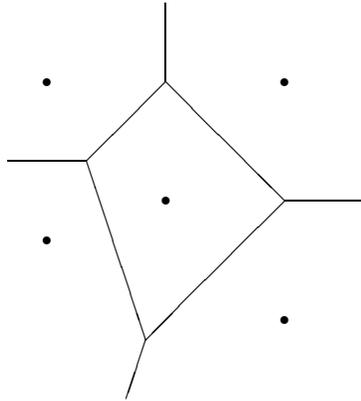
For 2D (and for n-D), MSE is minimized for given points by Voronoi regions.

For given regions, MSE minimized by conditional means.

Lloyd-Max still finds local min.

ENTROPY-CODED QUANTIZATION

Finding minimum MSE quantizer for fixed M is often not the right problem.

With quantization followed by discrete coding,

quantizer should minimize MSE for fixed representation point entropy.

Given regions, the representation points should be the conditional means.

Given the representation points, the Voronoi regions are not necessarily best.

The exact minimization here is difficult, but a simple approximation works for high rate (large entropy).

This is best explained by defining an entropy like quantity for random variables with a probability density.

Definition: The differential entropy of a continuous-valued real random variable U with pdf $f_U(u)$ is

$$\mathsf{h}[U] = \int_{-\infty}^{\infty} -f_U(u) \log f_U(u) \ du.$$

$$\mathbf{h}[U] = \int_{-\infty}^{\infty} -f_U(u) \log f_U(u) \ du.$$

Similarities between $\mathbf{h}[U]$ and discrete $\mathbf{H}(V)$:

$$\begin{aligned}
\mathbf{h}[U] &= \mathbf{E}[-\log f_U(u)] \\
\mathbf{h}[U_1 U_2] &= \mathbf{h}[U_1] + \mathbf{h}[U_2] \qquad \textbf{for } U_1, U_2 \textbf{ IID} \\
\mathbf{h}(U + a) &= \mathbf{h}(U) \qquad \textbf{(shift invariance)}
\end{aligned}$$

Differences:

- $\mathbf{h}[U]$ **not scale invariant. If** $f_U(u)$ **stretched by** a, $\mathbf{h}[U]$ **increases by** $\log a$, **i.e.,**

$$\mathbf{h}(aU) = \mathbf{h}(U) + \log a$$

- $\mathbf{h}[U]$ **can be negative.**

Uniform high-rate scalar quantizers

$$\cdots \quad \mathcal{R}_{-1} \quad \mathcal{R}_0 \quad \mathcal{R}_1 \quad \mathcal{R}_2 \quad \mathcal{R}_3 \quad \mathcal{R}_4 \quad \cdots$$

$$\cdots \quad a_{-1} \quad a_0 \quad a_1 \quad a_2 \quad a_3 \quad a_4 \quad \cdots$$
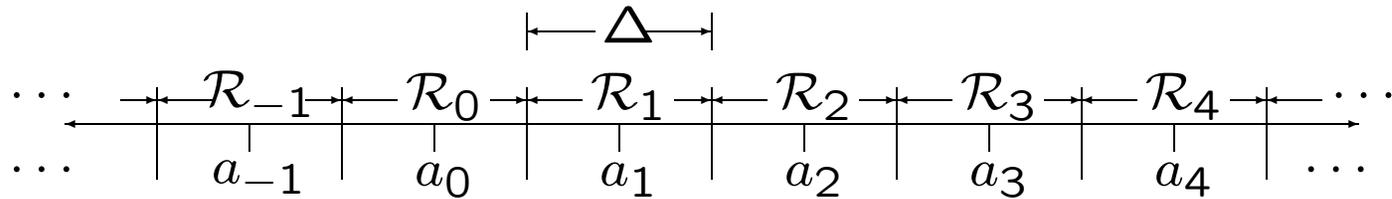
Assume $\triangle$ is small - $f_U(u)$ almost constant within each region. Define average f as

$$\overline{f}(u) = \frac{\int_{\mathcal{R}_j} f_U(u)\, du}{\triangle} = \frac{\Pr(\mathcal{R}_j)}{\triangle} \qquad \text{for} \quad u \in \mathcal{R}_j$$

$\overline{f}(u)$ $f_U(u)$

High-rate approximation: $\overline{f}(u) \approx f_U(u)$ **for all** u.

Conditional on $u \in \mathcal{R}_j$, $\overline{f}_{U|\mathcal{R}_j}(u) = 1/\Delta, \quad u \in \mathcal{R}_j$.

Conditional on $u \in \mathcal{R}_j$, $f_{U|\mathcal{R}_j}(u) \approx 1/\Delta, \quad u \in \mathcal{R}_j$.

For $\overline{f}$,

$$\text{MSE} \approx \int_{-\Delta/2}^{\Delta/2} \frac{1}{\Delta} u^2 du = \frac{\Delta^2}{12}$$

Entropy of quantizer output V. V has alphabet $\{a_j\}$ and pmf given by

$$p_j = \int_{\mathcal{R}_j} f_U(u) \ du \qquad \text{and} \qquad p_j = \overline{f}(u)\Delta.$$

$$
\begin{aligned}
\mathsf{H}[V] \ &= \ \sum_j -p_j \log p_j \\
&= \ \sum_j \int_{\mathcal{R}_j} -f_U(u) \log[\overline{f}(u)\Delta] \ du \\
&= \ \int_{-\infty}^{\infty} -f_U(u) \log[\overline{f}(u)\Delta] \ du \\
&= \ \int_{-\infty}^{\infty} -f_U(u) \log[\overline{f}(u)] \ du - \log \Delta \\
&\approx \ \int_{-\infty}^{\infty} -f_U(u) \log[f_U(u)] \ du - \log \Delta \\
&= \ \mathsf{h}[U] - \log \Delta
\end{aligned}
$$

Summary: uniform scalar high-rate quantizer

- **Efficient discrete coding achieves $\overline{L} \approx \mathsf{H}[V]$.**

- $\overline{L} \approx \mathsf{H}[V]$ **depends only on** $\triangle$ **and** $\mathsf{h}(U)$.

- $\mathsf{h}[U] \approx \mathsf{H}[V] + \log \triangle$ **explains diff. entropy.**

- $\qquad \overline{L} \approx \mathsf{h}(U) - \log \triangle; \qquad \mathsf{MSE} \approx \frac{\triangle^2}{12}$

- **Uniform approaches optimal for small** $\triangle$

$$\overline{L} \approx \mathbf{h}(U) - \log \triangle; \qquad \mathbf{MSE} \approx \frac{\triangle^2}{12}$$

Reducing $\triangle$ by a factor of 2 increases $\overline{L}$ by 1 bit/symbol and reduces MSE by factor of 4 (i.e., by 6 dB).

$$\overline{L} \approx \mathbf{h}[U] - \frac{\log 12}{2} - \frac{\log \mathbf{MSE}}{2}$$

MSE

$$\mathbf{MSE} \approx \frac{2^{2\mathbf{h}[U]-2\overline{L}}}{12}$$

$$\overline{L} \approx \mathbf{H}[V]$$
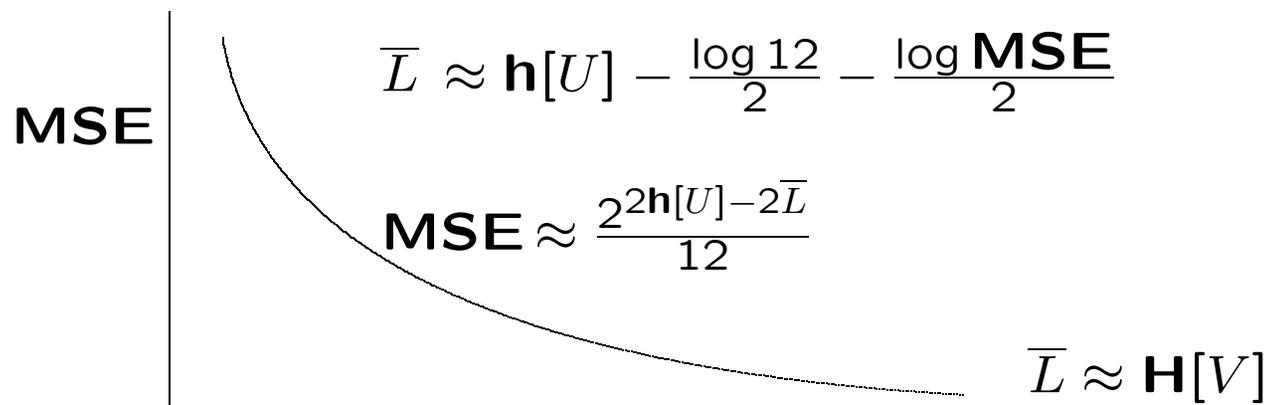
6.450 Principles of Digital Communication I
Fall 2009