Arithmetic Coding

- a nonblock code (also known as tree code)
- a codeword assigned to the entire input sequence s_m of length m symbols
- the codeword length $\approx -\log_2 p(s_m)$

Elias coder for a DMS $S = \{0, 1\}, \{p, 1-p\}$

- $I = [0, 1)$
- $\Sigma_{p(s_m)} = 1$
  all 2^m possible source
  sequences of length m
- $I = \cup_{l=1}^{2^m} I_l, \;\; I_l \cap I_k = \emptyset \quad (l \neq k)$

  $I_l \leftrightarrow s_m : |I_l| = p(s_m)$

- $m = 1$:
  $s_m = 0 \leftrightarrow [0, p)$
  $s_m = 1 \leftrightarrow [p, 1)$
- Assume
  $s_{m-1} \leftrightarrow [L^{(m-1)}, R^{(m-1)})$
- $s_{m-1}\delta_m \leftrightarrow [L^{(m)}, R^{(m)})$:

$$\begin{aligned} \delta_m = 0: \; L^{(m)} &= L^{(m-1)} \\ R^{(m)} &= L^{(m-1)} \\ &\quad + p(R^{(m-1)} - L^{(m-1)}) \end{aligned}$$

$$\begin{aligned} \delta_m = 1: \; L^{(m)} &= L^{(m-1)} \\ &\quad + p(R^{(m-1)} - L^{(m-1)}) \\ R^{(m)} &= R^{(m-1)} \end{aligned}$$

- $|[L^{(m)}, R^{(m)})| = p(s_m)$:
  $m = 1$
  Assume $m = k - 1$
  For $m = k$:

$$\begin{aligned} p(s_k) &= p(s_{k-1}) \cdot p(\delta_k) \\ &= |[L^{(m-1)}, R^{(m-1)})| \cdot \\ &\quad [(1-\delta_k)p + \delta_k(1-p)] \end{aligned}$$

$$\begin{aligned} |[L^{(m)}, R^{(m)})| &= |[L^{(m-1)}, R^{(m-1)}| \cdot \\ &\quad [(1-\delta_k)p + \delta_k(1-p)] \end{aligned}$$

- The subintervals produced by all the possible sequences of length m are non-overlapping and their union equals I.
- If $[L_l, R_l) \leftrightarrow s_m$, a codeword for s_m can be constructed:
  - expanding L_l in binary form
  - retaining the $\underline{\underline{n_l = \lceil -\log_2 p(s_m) \rceil}}$ bits after the decimal point
  - denoting the n_l bits expansion of L_l by E_l:

$$E_l \le L_l < E_l + 2^{-n_l}$$

$$\leq L_l + 2^{-n_l} \leq L_l + p(s_m) = R_l$$

– if $E_l = L_l$, then encode the I_l or s_m by n_l bits codeword E_l

– If $E_l < L_l$, then
$L_l < E_l + 2^{-n_l} < R_l$
and encode the I_l or s_m by
the n_l bits codeword $E_l + 2^{-n_l}$

  - Therefore, only $n_l = [-\log_2 p(s_m)]$ bits are needed to uniquely specify the I_l or s_m.
  - Given n_l bits codeword for s_m, how to find out s_m?

- The major implementation problem of the Elias code: the precision required to compute I_l
  $m \uparrow \implies |I_l| \searrow \implies$ more bits needed to precisely identify I_l

Chapter 4
Entropy Estimation and Lossless Compression

- In light of the noiseless source coding theorem, the bit rate can be made arbitrarily close to the entropy of the source that generated the image.
- How to determine the entropy?
- Characterize the source using a certain model and find the entropy w.r.t. that model.
- Accurate source modeling is essential.

- The effectiveness of a model is determined by how accurately it predicts the symbol probabilities.
- For natural information-generating sources such as speech and images, using the more complex models will result in lower entropies and higher compression if they are capable of accounting for the structure.
- The real challenge: approximating the source structure as close as possible while keeping the complexity of the model to a minimum.

- Another approach:
  - segmenting the image into blocks of size N
  - using the frequency of occurrence of each block as a measure of its probability
  - The entropy per original source symbol of the so-formed adjoint source → the entropy of the original source
    - the convergence is slow
    - large block sizes
    - for an 8-bit image and N-pixel block, there are 256^N possible symbols

- Structure and Entropy of English Language
  - $S = \{$26 English letters and a space symbol "*"$\}$
  - The simplest model: a DMS with equiprobable symbols
    - $p(s_i) = \frac{1}{27}, i = 1, \ldots, 27$
    - $H(S) = \log_2(27) = 4.75$ bits/symbol
    - This model does not reflect any of the structure contained in the English Language
    - Its entropy (uncertainty) is high and any coding scheme based on the model cannot reduce the redundancy.

- A better DMS model uses the actual probabilities of symbols:
    - $H(S) = \Sigma_S \, p(s_i) \log_2 \frac{1}{p(s_i)} = 4.03$ bits/symbol
    - The model does not provide good English, but it reflects some of the structure of the language.
    - The words generated by this model contain a more realistic proportion of vowels and consonants.
    - The model does not take into account any dependence among different letters.

- A first-order Markov source with the appropriate conditional probabilities

  27 probability tables, one for each state of the Markov source

  $$H(S) = \Sigma_s \Sigma_s P(s_i, s_j) \cdot \log_2 \frac{1}{p(s_i|s_j)} = 3.32 \text{ bits/symbol}$$

- A second-order Markov source with 729 (27 × 27) probability tables

  $$H(S) = \Sigma_s \Sigma_s \Sigma_s p(s_i, s_j, s_k) \log_2 \frac{1}{p(s_i|s_j,s_k)} = 3.1 \text{ bits/symbol}$$

  - the order of the Markov source $\nearrow$ $\Rightarrow$ the number of model parameters $\nearrow$
  - convergence to the entropy of the English language

- Shannon's estimate of the entropy of the English language is between 0.6 and 1.3 bits/symbol

- Predictability and Entropy of English Language
  - Shannon exploited the fact:
    - anyone speaking a language implicitly possesses an enormous knowledge of the language
- He found upper and lower bounds to the entropy of printed English by eliciting knowledge of the conditional probability distribution of the symbols from a subject through using a guessing game.

- A subject was shown $N-1$ consecutive symbols of an unfamiliar text, and was asked to guess the next letter in the passage.
- Guesses continued until the correct letter was selected.
- This guessing process ranks the selected letters in decreasing order of conditional probability according to the subject's knowledge of the English language.
- The experiment was repeated n times.

- $q_i^N = \#$(the subject required i guesses to discover the correct letter given the previous $N-1$ letters)
- The entropy or uncertainty of the text is bounded by

  $$H(S) \leq -\sum_{i=1}^{27} \frac{q_i^N}{n} \log_2 \left(\frac{q_i^N}{n} \right)$$

  and

  $$H(S) \geq \sum_{i=1}^{27} i \left(\frac{q_i^N}{n} - \frac{q_{i+1}^N}{n} \right) \log_2 i$$

- One hundred samples of English text selected at random from a book, each contains a hundred letters in length ($N = 100$)
- upper bound 1.3 bits/symbol
  lower bound 0.6 bits/symbol

Chapter 5

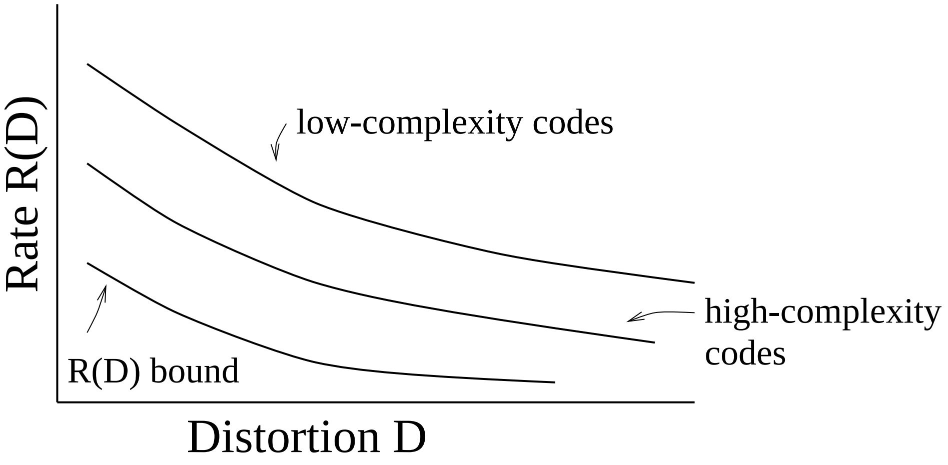
Rate-Distortion Theory and Lossy Compression

- What is the minimum bit rate required to encode a source while keeping the resulting degradation below a certain level?
- Addressed by Rate-Distortion Theory
  - establishes theoretical performance bounds for lossy data compression according to a fidelity criterion.
  - provides a rate-distortion function $R(D)$ for a broad class of distortion measures and source models.

- For any given level of distortion D, $\exists$ a coding scheme:
  - rate $\approx R(D)$
  - average distortion $\approx D$

  $\nexists$ a code:
  - rate $< R(D)$
  - achieves reproduction with distortion $\leq D$
- It can be shown:

$$R(D): \quad \begin{array}{l} \text{convex,} \\ \text{continuous,} \\ \text{strictly decreasing} \end{array}$$



- $R(0) \geq H(S)$
- To characterize $R(D)$, one needs a source model and a distortion criterion.
- For a DMS source model and a context-free distortion measure, the closed-form solutions of $R(D)$ for some source distributions have been found.

- The results are of little utility to images.
- Natural imagery is highly correlated and cannot be adequately modeled by a DMS.
- For sources with memory, one important case where a solution exists is for a Gaussian source with a weighted quadratic-error distortion measure.
- source models accurately describing natural imagery
- distortion measures correlating well with visual criteria

Part III
Lossless Compression Techniques

- In medical imaging, compressing digital radiographs with a lossy scheme may compromise diagnostic accuracy.
- The price paid for an error-free reconstruction is a much lower compression ratio as compared to lossy schemes.
  - bit plane encoding
  - lossless predictive coding
  - lossy plus lossless residual encoding

Chapter 6
Bit Plane Encoding

- $N \times N$ image,
  each pixel represented by k bits
- Original image decomposed into a set of k $N \times N$ bit planes
  - numbered 0 for the least significant bit (LSB) plane through $k-1$ for the most significant bit (MSB) plane
  - each bit plane can be encoded efficiently.

  - The more significant bit planes generally contain major structural information and are highly compressible, progressively, reconstructing an image using the bit planes can be a viable technique.

- Gray Code
  - Bit plane encoding algorithms typically encode the bit planes independently and take advantage of the existence of large uniform areas (coherency) in each plane to achieve high compression
  - It is desirable to form the bit planes so as to increase their coherence and minimize their complexity.

- The Gray code:
  - mapping a set of numbers into a binary alphabet:
    - successive numerical changes result in a change of only one bit in the binary representation

Step 1.	Starting with the MSB of the binary code representation, all 0's are left intact until a 1 is encountered.
Step 2.	The 1 is left intact, but all the following bits are complemented until a 0 is encountered.

Step 3. The 0 is complemented, but all the following bits are left intact until a 1 is encountered.
Step 4. Go to Step 2.

The inverse mapping:

Step 1. Starting from the MSB of the Gray code representation, all 0's are left intact until a 1 is encountered.
Step 2. The 1 is left intact, but all the following bits are complemented until another 1 is encountered.

Step 3. The 1 is complemented, but all the following bits are left intact until another 1 is encountered.
Step 4. Go to Step 2.

- The complexity of the bit planes $\nearrow$ as one moves to the lower planes
- The complexity of the Gray code bit planes $<$ the complexity of the corresponding binary bit planes