Multimedia Signal Processing - 2nd module

Prof. Marco Tagliasacchi

September, 21 2010

1 Question - 10 pts

A discrete source X emits symbols from a finite alphabet of $M = 6$ symbols

- Compute the maximum entropy that the source can achieve. What is the condition that need to be verified for this to hold?
- By observing a large number of symbols (i.e. $N \to \infty$), the estimated entropy is $\hat{H}(X) = 1.6$ bps. Compute the lower and upper bounds on the average code length of Shannon's coding.
- The estimated entropy rate is $\hat{H}_\infty(X) = 0.9$ bps. Based on this observation, list lossless coding methods that can be used to compress such a source. Select one of them and discuss it in detail.

Solution

- The maximum value of the entropy is achieved when all symbols are equally likely. That is, $p(a_i))1/6$, $i = 1, \dots, 6$. In this case, the entropy is equal to $H(X) = \log_2 6 = 2.585$ bits/symbol
- Let L_S denote the average code length of a Shannon's code for a source X, expressed in bits/symbol. The following condition holds

$$H(X) \leq L_S < H(X) + 1 \quad (1)$$

  In this case,

$$1.6 \leq L_S < 2.6 \quad (2)$$

- Since $\hat{H}_\infty(X) < H(X)$, we can conclude that the source has memory. In order to lossless encode a source with memory, we can employ one of the following methods
  - Predictive coding (without the quantization step). We expect that the entropy of the prediction residuals to be smaller than $H(X)$ (but not necessarily as small as $\hat{H}_\infty(X) < H(X)$).
  - Huffman coding, grouping symbols in blocks of length N
  - Elias/Arithmetic coding, in which the coding algorithm is driven by conditional probabilities, rather than marginals.

  Then, you should select one of them, and discuss it in detail.

2 Question - 5 pts

Illustrate the properties of speech signals. Describe the basic ideas and building blocks of the source-system model for speech synthesis. In particular, explain the difference between the short-term and the long-term predictor.

Solution

See slide set `MMSP_2.1_Speech coding`.

3 Question - 12 pts [Matlab]

Consider a speech segment (load the sample signal `matlab`).

- Compute the prediction coefficients for each 20ms time window (use $p = 10$ coefficients) [Hint: you can use `lpc` or `levinson`]
- Design a suitable quantizer for the prediction coefficients allocating $R = 80$ bits per window [Hint: you can use `ac2rc` and `rc2ac`].
- Synthesize speech using the quantized prediction coefficients and the ideal excitation signal obtained at the encoder
- Plot the original and synthesized speech and compute the SNR

Solution

```matlab
clear all
close all
clc

%load matlab speech signal
load mtlb
x = mtlb;

Twin = 20*10^-3; %20 ms window

P = 10; %prediction order

Swin = floor(Fs*Twin); %window length in samples

%crop the signal so that we have an integer number of windows
x = x(1:floor(length(x)/Swin)*Swin);
Nwin = length(x)/Swin;

%output
y = NaN*ones(size(x));

%we're going to extract prediction coefficients, and quantize the
%corresponding reflection coefficients. We want to use 80 bits per windows,
%so we will use 8 bits per coefficient. Remember that the reflection
%coefficients k_i are whitin the range -1 < k_i < 1.
R = 8;
STEP = (1 - (-1))/2^R;

for iw = 0:Nwin-1

    %->ENCODER
    %window signals
    xwin = x(Swin*iw+1:Swin*(iw+1));

    %estimate coefficients
    [r lags] = xcorr(xwin);
    r = r(lags>=0);
    coeffs = levinson(r,P);

    %compute residuals
    res = xwin - filter([0 -coeffs(2:end)],1,xwin);

    %find reflection coefficients
    k = poly2rc(coeffs);
```

```
    %quantize reflection coefficients
    k_q = STEP*floor(k./STEP)+STEP/2;

    %->DECODER
    %come back to LPC coefficients
    coeffs_q = rc2poly(k_q);

    %reconstruct input using ideal excitation (residuals)
    y(Swin*iw+1:Swin*(iw+1)) = filter(1,coeffs_q,res);
end

SNR = 10*log10(var(x)/var(x-y));


figure();
subplot(211)
plot(x);
hold on
plot(y,'or');
title(['Original (blue) and reconstructed (red) signals. SNR: ' num2str(SNR) ' dB']);
subplot(212)
plot(x-y);
title('Error signal');
disp(['SNR: ' num2str(SNR) ' dB']);
```

4 Questions - 5 pts (each answer can be either TRUE or FALSE)

1. Let X denote a memoryless source with an infinite alphabet, e.g. $x \in \mathbb{R}$, and $D(R)$ the corresponding rate-distortion curve

   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ $D(0) = \infty$
     FALSE. $D(0) = \sigma_x^2$.
   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ $D(R) \geq D_S(R)$, where $D_S(R)$ is Shannon's bound
     TRUE. For a given rate, the Shannon's bound is a lower bound on the achievable region.
   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ $R(0) = H(X)$
     FALSE. The entropy is undefined for continuous valued sources.

     $$\lim_D \to 0 \quad R(D) = +\infty \tag{3}$$

   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ $D(R_i) \leq D(R_j)$, if $R_i < R_j$
     FALSE. $D(R)$ is a monotonically non-incresing function of R.

2. Consider an image coding system

   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ The choice of the block size affects coding efficiency
     TRUE. The larger the block size, the higher the coding efficiency, since it is possible to address inter-pixel correlations at larger distances.
   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ Zonal quantization addresses the non-stationary behaviour of natural images
     FALSE. Zonal quantization assumes that the source is wide-sense stationary. This is not the case for natural images.
   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ The luminance component is downsampled by a factor of 2 in both directions
     FALSE. The chrominance components are typically downsampled.

3. Consider a video coding system and a GOP structure $IPP...I$

   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ The coding efficiency improves by increasing the GOP size
     TRUE. The longer the GOP size, the smaller is the fraction of I-frames. Note that, for the same quality, $R_I > R_P$
   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ The computational complexity increases by increasing the GOP size
     TRUE. The longer the GOP size, the larger is the fraction of P-frames. P-frames are more complex to encode due to the need of motion estimation.
   - $\boxed{\text{T}}$ $\boxed{\text{F}}$ The adoption of P-frames introduces drift
     FALSE. Drift is prevented if P-frames are encoded computing the residuals with respect to the previously decoded frames, as in DPCM.