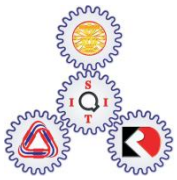


Sirindhorn International Institute of Technology

Thammasat University at Rangsit

School of Information, Computer and Communication Technology

ET 601 Project 2: Entropy Rate Estimation

In this project, we attempt to estimate the entropy rate of non-technical written text in a particular language. From the nationality of students taking the class, the language considered here is Thai or Burmese.

Instructions

1. This project has four parts, with increasing difficulty.
2. Your report should include description of your MATLAB implementation, simulation results, and result discussion.
   Note that the description part is not the same as showing your MATLAB scripts. It should explain the steps that you use in a way that those who may not know MATLAB but know some basic programming (and some Mathematics) can understand.
3. Scripts for all parts should be included as appendices at the end of the report. It should contain some amount of explanation in the form of comments.
4. The report is to be submitted as a PDF file via email on the last day of class. The PDF file name should be ET601_Project2_FIRSTNAME.pdf in which the FIRSTNAME part is replaced by your first name. The MATLAB scripts should be compressed together in a file ET601_Project2_FIRSTNAME.zip and submitted with the report.
5. Do not copy scripts or written materials from other reports. Scores of submitted works that are too similar will be shared from the score of one report. For example, if we find three copied reports, each of them would get 1/3 of the scores.

Mar 8 (Sat)

Due Date: March 8, 2014

Introduction

Entropy

(Shannon) Entropy is a measure of randomness (uncertainty, ambiguity) in a random variable. In particular, it is a measure of the amount of information required on the average to describe the random variable. Let X be a discrete random variable whose support is S_X and probability mass function $p_X(x)$, $x \in S_X$. The entropy $H(X)$ of a discrete random variable X is defined by

$$H(X) \equiv -\sum_x p_X(x) \log_2 p_X(x).$$

The log is to the base 2 which implies entropy is expressed in bits. We will use the convention that

$$0 \log 0 = 0.$$

Entropy rate

Consider a sequence of random variables $X_1, X_2, X_3, \ldots$ For us, we are interested in a sequence of characters in a (thick) book. In which case, X_1 represents the first character in the book, X_2 represents the second character, and so on. In this context, the support of each random variable is called the **alphabet**.

To get some idea of how the calculation works, let's work with a very short sample of text in a language that has only two possible symbols:

#@@@@#@@@@@@@@@@@##@ (1)

By considering individual symbols in the above text, we found that the relative frequency of "#" is 4/20 = 1/5, and the relative frequency of "@" is 16/20 = 4/5. When the length of the sequence is large enough[1], the relative frequency is a good approximation for probability and we can estimate the entropy by

$$H \approx -\frac{1}{5}\log_2\frac{1}{5} + -\frac{4}{5}\log_2\frac{4}{5} \approx 0.7219. \quad (2)$$

Now, to take into account dependency among the symbols in the text, we start by considering pairs of adjacent symbols in the text. In general, we look at $X_1X_2, X_2X_3, X_3X_4, X_4X_5, \ldots$ In our sample sequence above, the relative frequencies are 3/19, 13/19, 2/19, 1/19 for the pairs #@, @@, @#, ##, respectively. So the estimate of the entropy is

$$-\frac{3}{19}\log_2\frac{3}{19} - \frac{13}{19}\log_2\frac{13}{19} - \frac{2}{19}\log_2\frac{2}{19} - \frac{1}{19}\log_2\frac{1}{19} \approx 1.3605. \quad (3)$$

Note that this is the amount of entropy per two symbols. Therefore, the amount of entropy per symbol is

[1] It's <u>not</u> large enough here because the length is only 20. This is why choosing a large text source for this project is important.

$$H \approx \frac{1.3605}{2} \approx 0.6803.$$

Similarly, we can consider three adjacent characters at a time, find the relative frequencies, estimate entropy, and then calculate the amount of entropy per symbol. When we consider *L* adjacent symbols in the sequence, we denote the corresponding amount of entropy per symbol by H_L. In particular,

$$H_L = \frac{1}{L} \times \left(\text{entropy value when } L \text{ adjacent symbols are considered} \right) \tag{4}$$

Hence, in the example above, $H_1 \approx 0.7219$ and $H_2 \approx 0.6803$.

The **entropy rate** is defined as

$$H_r = \lim_{L \to \infty} H_L \,. \tag{5}$$

Theoretically, when we have infinite amount of texts to work with, we can estimate the entropy rate by evaluating H_L when *L* is large. Alternatively, one can also find the entropy rate from

$$H_r = \lim_{L \to \infty} \left(L H_L - \left(L - 1 \right) H_{L-1} \right). \tag{6}$$

Project Description

Part A (25%)

In each of the files `sampleX_1e5t.mat`, `sampleX_1e6t.mat`, and `sampleX_1e7t.mat`, a text is saved in a vector (character array) `X`. The three versions are different in the length (*n*) of the text considered. In all versions, there are only two possible symbols in the alphabet: @ and #.

i) Write a MATLAB script to plot the estimated values of H_L when *L* = 1,2, …, 25. Figure 1 shows the three expected plots.

Remark: The process that generates the provided text data is known to have an entropy rate of about 0.6827 bit per symbol. The plots above show that, when the size of the text is large ($n = 10^7$), the estimated H_L does converge to the entropy rate. However, when the text size is smaller ($n = 10^6$), the estimated entropy rate drops below the true entropy rate. The drop occurs even earlier when the text size decreases further ($n = 10^6$). Again, this is why we need the sample text size to be large.

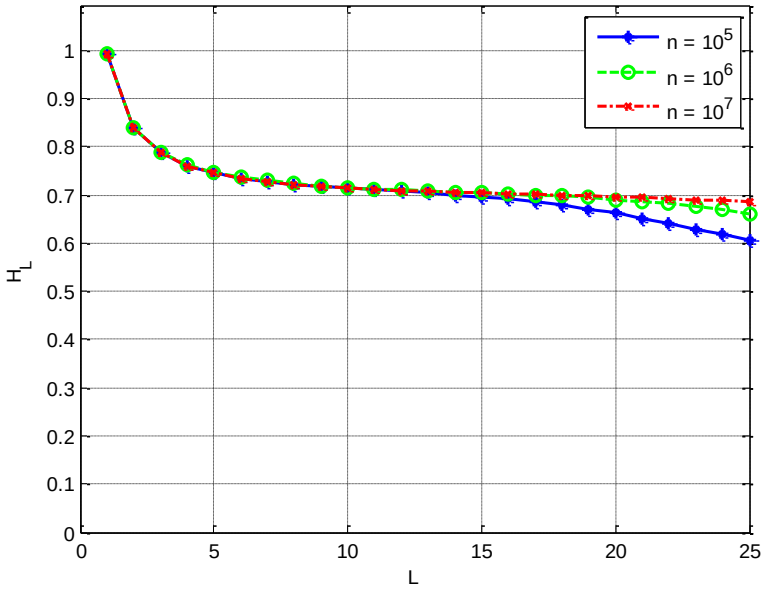
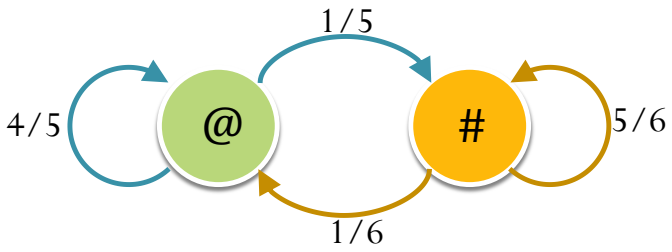


Figure 1: The estimated amount of entropy per symbol for the provided sources.

ii) In the same plot as part (i), plot $LH_L - (L-1)H_{L-1}$.

Part B (25%)

Consider the following text generation process in which there are only two possible symbols in its alphabet: @ and #. Suppose that $X_1 = @$. The rest of the text is generated by the process described in the diagram below:



For $k \geq 2$, to generate X_k, the process first look at X_{k-1}. The numbers on the arrows give the following conditional probabilities:

$$P\left[X_k = @ \middle| X_{k-1} = @\right] = \frac{4}{5},$$

$$P\left[X_k = \# \middle| X_{k-1} = @\right] = \frac{1}{5},$$

$$P\left[X_k = @ \middle| X_{k-1} = \#\right] = \frac{1}{6}, \text{ and}$$

$$P\left[X_k = \# \middle| X_{k-1} = \#\right] = \frac{5}{6},$$

i) Write a MATLAB script to generate $X_1, X_2, \ldots, X_{1,000,000}$ according to the process description above.

ii) Plot the estimated values of H_L (using the script that you have constructed in the previous part) when *L* = 1,2, …, 25.

Part C (25%)

Now that you have some experience with the estimation of entropy rate, it's time to estimate the entropy rate of the English language. The text source(s) that you use should be a typical representative of a writing in English language. In your report, there should be some discussion on why you think your selection is a good one.

Note that the number of possibilities grows very fast as we increase the values of *L*. However, try to increase your *L* values to as large a value as possible.

Use the plot of H_L and possibly $LH_L - (L-1)H_{L-1}$.

Also include interesting probability facts about the language. For example, what are the top ten most frequently occur pairs? What are the top ten in terms of conditional probabilities?

Part D (25%)

Finally, you are now ready to estimate the entropy rate of Thai or Burmese language. The new difficulty here is how to put text encoded in non-English symbols into MATLAB.