Cryptography: Handin #2

Søren Løbner
lobner@cs.au.dk

September 9, 2010

Handin 2

The assignment is about a theoretic instance of Wheel of Fortune where you have to guess which collection of letters are hidden behind a row of cards in front of you. Each of the letters are chosen independently from a distribution $[p_0, p_1, \ldots, p_{25}]$. At each round you get to choose one letter, the cards with that letter behind is then turned. So you must choose the one which reveals the most information, on all the letters combined.

First question

If your guess is the letter i, how many bits of information will you learn on average from playing the game (as a function of p_i and N)?

At each guess we either learn that the letter is at the position(s) or that it is not there at all. We can model this as the random variable $\mathbf{X}$ which is defined on the set $X = \{T, F\}$ for $true$, the letter is there at one or more positions or $false$, the letter is not there.

The distribution function for this is then defined as:

$$Pr[T] = p_i$$

$$Pr[F] = (1 - p_i)$$

Then by the following equation we calculate the entropy for a random variable

$$H(X) = \sum_{x \in X} (Pr[x] \times log_2 \frac{1}{Pr[x]})$$

This for T and F gives us:

$$H(X) = p_i \times log_2 \frac{1}{p_i} + (1 - p_i) \times log_2 \frac{1}{(1 - p_i)} \tag{1}$$

Now from the above we have the result for each card, so for all cards we multiply by N and get:

$$T(p_i, N) = N \times (p_i \times log_2 \frac{1}{p_i} + (1 - p_i) \times log_2 \frac{1}{(1 - p_i)}) \tag{2}$$

Second question

What strategy does your result suggest for choosing your guess, given frequencies $[p_0, \ldots, p_{25}]$ as in English?

If we use the frequencies $[p_0, \ldots, p_{25}]$ as in English and use (2) to calculate the average bits of information for E which is the most frequent letter and Z which is one of the less frequent letters in English, we get

$$E: \quad T(0.127, N) = N \times (0.127 \times log_2 \frac{1}{0.127} + (1 - 0.127) \times log_2 \frac{1}{(1 - 0.127)})$$

$$Z: \quad T(0.001, N) = N \times (0.001 \times log_2 \frac{1}{0.001} + (1 - 0.001) \times log_2 \frac{1}{(1 - 0.001)})$$

This suggests that choosing the most frequent letters gets you the most bits of information.

Third question

based on this, does it make sense that players in real life choose the most frequent letter(s)?

Yes it makes sense since they get the most bits of information on average.

Fourth question

Would this be a good strategy no matter what the frequencies were?

No, if the most frequent letter is too frequent, that is if a letter has p_i close to 1, then you should not choose it, because the average bits of information decreases after i peaks at $p = \frac{1}{2}$ .So you should choose the letters that has a frequency as close to $p = \frac{1}{2}$ to get the highest average bits of information.