Torah Code Cluster Probabilities

Robert M. Haralick
Computer Science
Graduate Center
City University of New York
365 Fifth Avenue
New York, NY 10016
haralick@netscape.net

1 Introduction

In this note we analyze the probability calculation discussed by Roy Reinhold for determining the probability that a cluster of ELSs in a table would happen by chance.

Our first order of business is to say what probability means. For us probability must always be associated with an experiment. The experiment begins with an experimental protocol that has

1. *a priori* specification of the key words
2. a monkey text population
3. an ELS skip specification
4. a resonance specification
5. a procedure by which a compactness score value for a text can be computed

In the experiment, a text is randomly sampled from a specified population of texts, called monkey texts to indicate that whatever effect is thought to be occuring with the Torah text it is certainly not occuring in the texts of the monkey text population. In accordance with a given experimental protocol, a table of ELSs is constructed and a statistic value C measuring the compactness of the table is computed. The probability we are interested in is the probability that a randomly sampled text from the population will have a table whose compactness score C is smaller than (better than) C_0: $Prob(C \leq C_0)$.

This probability can be determined two ways. One way is to determine the probability analytically exactly analogous to the kind of analytic computation of drawing five cards at random in a poker game and having a full house. In the case of small combinatorial situations, this way is tractable. In the general situation, particularly with regard to Torah code tables, it is difficult if not tractable.

The second way is to estimate the probability by a Monte Carlo experiment. In the Monte Carlo experiment a given number N of texts are sampled from the specified population. For each of the randomly sampled texts, a table of ELSs is constructed and a statistic values $C_1, \ldots, c_N$ measuring the compactness of the table is computed. Then $Prob(C \leq C_0)$ is estimated by

$$Prob(C \leq C_0) = \frac{\#\{n \mid C_n < C_0\}}{N}$$

The compactness score value C_0 is typically determined by carrying out the experiment with the Torah text. Hence the probability $Prob(C \leq C_0)$ means the probability that a randomly sampled text from the text population will have a table that is as compact or better than the table obtained from the Torah text.

There is an important internal condition that must be satisfied in the experiment. This is the condition of symmetry or uniformity. Simply stated it is that whatever is done to the first text to establish a value C_0 must be done exactly the same way to compute a value C from each text sampled from the text population.

Reinhold attempts an analytic calculation. We will go through the essence of his calculation explaining exactly what probability he is approximately computing and how that differs from the probability that is desired. We will show that his probability calcuation produces a number that has to be too small from the probability that an experiment produces.

2 The Reinhold Calculation

The Reinhold calculation is based on a letter permuted population of monkey texts. Reinhold uses the Codefinder program. That program lets the user specify a Torah text such as Genesis, or the Five Books, or any one book of the Tanach, or the entire Tanach. The program then lets the user set a fixed maximum skip specification, say of 1 to 15,000. The user provides a

list of key words. Then the program finds all the ELSs satisfying the skip specification of the given list of key words in the specified text. Next the user does some interactive manipulation attempting to construct in some undefined sense the smallest table having at least one ELS of the key words. In terms of a completely specified algorithmic procedure, this constitutes a weak link. But it is in fact not the cause of the difficulty of the Reinhold calculation.

Once the user has constructed a table, the user has a list of the ELSs that the table contains. Associated with each ELS is its absolute skip. The Torah code effect has been hypothesized to occur at the smaller ranked skip lengths and therefore, the compactness score function should put more weight on those ELS with relatively smaller skips.

In a text of length Z a key word of length L characters, the number N of possible placements an ELS can have of skip length 1 though skip length D, is given by

$$N(Z, L, D) \;=\; D * (2Z - (L-1) * (D+1))$$

In the letter permuted text population, the ELS placement probability for ELSs of a word whose letters are $< \alpha_1, \ldots, \alpha_K >$ is given by

$$p \;=\; \prod_{k=1}^{K} p(\alpha_k)$$

Hence, the expected number of ELSs in a randomly sampled text from the letter permuted text population is $E = pN$.

The expected number that the codefinder program provides is not based on the search, but on the skip of the ELS. That is, D is set to the absolute value of the ELS skip.

The codefinder program provides this ELS expected number in terms of an R-value defined by

$$R = log(1/E) \tag{1}$$

ELSs having small expected number, small being less than one, will have R-values larger than 1.

Suppose that within the area A of the interactively constructed table there are M ELSs, with corresponding expected numbers $E_1, \ldots, E_M$ and R-values of $R_1, \ldots, R_M$ Let us also suppose that each key word has at least one

ELS present.[1] Reinhold then multiples each expected number by the fraction A/Z to obtain what might be called the expected number E' of ELSs within the table area.[2] And the corresponding matrix R-values, here denoted by R', are computed from these expectations.

$$\begin{aligned} E' &= \frac{A}{Z}E \\ R' &= log(1/E') \\ &= log(\frac{1}{(A/Z)E} \\ &= log(1/(A/Z)) + log(1/E) \\ &= R + log(Z/A) \end{aligned}$$

Each matrix R' value can be seen to be the R-value plus the log of the length of the text divided by the area of the table. Reinhold then sums up the positive matrix R-values to obtain what he calls the matrix R-value, an initial summary score for the R-value of the table.

$$R_{matrix} = \sum_{\substack{m=1 \\ R'_m>0}}^{M} {R'}_m$$

Let us for the moment assume that the user has constructed the smallest area table containing at least one ELS of each of the key words. Some of the key words in the table may have more than one ELS. The Reinhold summing method gives extra reward when there is more than one ELS of a key word in the table. Some of the Torah code researchers have argued that this is important. Here, however, there is a problem with the probability calculation itself. Suppose that a key word has only one ELS in the table and that its R-value is not greater than zero. Then, in effect, the assumed *a priori* word list has been changed based on information obtained from the search. And the effect of this change in the score calculation is to bias the score in favor of Torah text. The reason that the bias is toward the Torah text is that

[1]This supposition itself is problematic because what typically happens is that a key word with no ELSs in a table will just be thrown away, making the key word set not *a priori*. But this problem is a problem with the *a priori* specification of the key words and not a problem with probability calculation itself.

[2]It can be seen from (1) that if the length of the text is reduced to half, the expected number E' of ELSs is in fact not reduced to half so this calculation is not quite right itself.

in Reinhold's analytic calculation, the very same procedure is not done for each text of the text population as required by the symmetry or uniformity condition of the experiment.

The next step of the Reinhold method is to exponentiate the R_{matrix} to obtain what we might call an inverse matrix expected number.

$$\frac{1}{E_{matrix}} = e^{R_{matrix}}$$

Before going on to the rest of the Reinhold calculation, let us try to understand the meaning of the calculation up to this point. The summations are a sum of log values. The exponentiation undoes the log function. So in essence the result is a calculation of the product of the expectations. We ignore from now on the omission of the terms in the sum whose matrix R-value is negative. We note only that this makes whatever calculation is done to produce a probability that is biased low by an unknown factor. Also for the sake of making our language simpler, we will just assume that there are M key words and each one has one ELS in the table.

$$Q = \prod_{m=1}^{M} \frac{1}{E'_m}$$

The case of interest is when each E'_m is very small, less than one. Recall that E'_m is an approximation for the expected number of ELSs of the corresponding key word that might occur in the area of the matrix. When E'_m is small, $E'_m << 1$, by the Poisson approximation to the binomial distribution, E'_m is the probability that at least one ELS of the absolute skip of the m^{th} ELS or smaller will appear in the table. Hence, the product of the expectations is the product of the probabilities that at least one ELS of the absolute skip of the ELS or less will occur in the table. Probabilities are multiplied when events are independent. So under the assumption that the occurrence of one key word having an ELS in the table is independent of the occurrence of another key word having an ELS in the table, the product $\prod_{m=1}^{M} E'_m$ is the probability that each key word has at least one ELS in the table. From this we understand the Reinhold's Q is the odds ratio $1 : Q$ that each key word would have at least one ELS in the table which is given at a fixed place, where the absolute skips of the ELSs are less than or equal to the absolute skips of the ELSs actually found in the table.

This is the glaring problem in the Reinhold calculation. It is biased in favor of the Torah text since the skips of the ELSs in the monkey texts are

now limited to be less than the corresponding absolute skips of the ELSs in the Torah text. This means, for example, that there could be monkey texts which have ELSs in a much smaller area table, but some with absolute skips higher than the corresponding ELSs in the Torah text and some with absolute skips lower than the corresponding ELSs in the Torah text. And these monkey text tables, which are better than that found in the Torah text are not counted as better.

Reinhold next proceeds to reduce the value of Q. This is because the $1 : Q$ ratio computed is for the probability of a table of the area determined by the search He reasons the following way. The user had interactively constructed the table. In this interaction the user had to go through and select from a potentially enormous number of combinations and try them out. However, the user is smart and does not do what a dumb brute force computer program might do. Nevertheless the Q needs to be penalized for the user interaction. He posits that the first key word the user employs is special in that the cylinder size is going to be selected so that it is resonant to an ELS of that key word. His resonance specification is that the row skip of the ELS on the cylinder must be at least 1 and no more than 6. Thus if the ELS of the first key word has absolute skip D, the cylinder sizes tried should be D, $D/2$, $D/3$, $D/4$, $D/5$, and $D/6$. Given an ELS of the first key word, in what window on the cylinder should the user look. Obviously he should look at a window centered around the position of the first ELS. This fixes the position of the window but not its size. Now around this position, the user in some non-algorithmic way extends the window so that it includes at least one ELS of each of the key words. So in effect the user is examining one position and six cylinder sizes for each ELS of the first key word. Reinhold therefore penalizes the user with a Bonferroni tax for each ELS of the first key word and for each cylinder size tried. Therefore the Bonferroni penalty on the odds ratio is the number of ELSs that the first key word had times 6. The adjusted odds ratio is then $1 : Q/B$ where B is the Bonferroni penalty.

This Bonferroni penalty is based on the number of ELSs of the first key word found in the Torah text. But the number of ELSs of the first key word found in the monkey texts are not the same as that found in the Torah text. And so the calculation uses a quantity from the Torah text that is not applicable to each monkey text.

Had the user been required to make the window extend around the first ELS the same number of columns to the left and right and the same number of rows to the left and right, there would be no question that the Bonferroni

bound is sufficient. But because this was not a requirement, the user had additional extension possibilities not accounted for and the Bonferroni bound is too low. Thus, the Q/B of the odds ratio $1 : Q/B$ is too high and the probability calculated is therefore too small.

In summary, for the reasons stated, the Reinhold calculation of the probability that as a compact table can be found in a monkey text can be too small, even by an order of magnitude.