Vigenère Cipher

- It is a polyalphabetic substitution cipher
- In Vigenère cipher each plaintext letter has multiple corresponding ciphertext letters
- The Vigenère Cipher was developed by mathematician Blaise de Vigenère in the 16th century.

Vigenère Cipher

- Def: Given m, a positive integer and $K = (k_1, k_2, \ldots, k_m)$ a key where each $k_i \in Z_{26}$, the Vigenere cipher is defined as:
- Encryption: $c_i = p_i + k_{i \text{ (mod } m)}$ (mod 26)
- Decryption: $p_i = c_i - k_{i \text{ (mod } m)}$ (mod 26)
- Example: Consider 'CODE' as the key and CRYPTANALYSIS as the plaintext

Plaintext:	C	R	Y	P	T	A	N	A	L	Y	S	I	S
Key	C	O	D	E	C	O	D	E	C	O	D	E	C
Ciphertext	E	F	B	T	V	O	Q	E	N	M	V	M	U

Cryptanalysis of Vigenère Cipher

- The key space of the Vigenere cipher is 26^m, m is key size
- Brute force techniques infeasible for sufficiently large values of m.
- Cryptanalysis of the Vigenere cipher has 2 main steps:
  - identify the period of the cipher (the length of the key)
    - Kasiski method
    - Index of Coincidence
  - finding the specific key

Kasiski Method

- Published by Friedrich Kasiski in 1863
- The Kasiski examination involves looking for strings of three or more characters that are repeated in the ciphertext.
- Find the distances between consecutive occurrences of the strings (are likely to be multiples of the length of the keyword)
- Find the greatest common divisor of all the distances.
- If a repeated substring in a plaintext is encrypted by the same substring in the keyword, then the ciphertext contains a repeated substring and the distance of the two occurrences is a multiple of the keyword length.
- Not every repeated string in the ciphertext arises in this way; but, the probability of a repetition by chance is small.

Example: Kasiski Method

- Intercepted message:

  VHVSSPQUCEMRVBVBBBVHVSURQGIBDU GRNICJQUCERVUAXSSR
- The gap between the "VHVS" pair is 18, implies key length may be 18, 9, 6, 3 or 2. The gap between the "QUCE" pair is 30, implies key length 30, 15, 10, 6, 5, 3 or 2.
- So looking at both together the most likely key length is 6 or possibly 3 (though in practice this is unlikely).

Index of Coincidence (Friedman Test)

- Invented by William F. Friedman in 1922
- Putting two texts side-by-side and counting the number of times that identical letters appear in the same position in both texts.
- The index of coincidence provides a measure of how likely it is to draw two matching letters by randomly selecting two letters from a given text.
- It is a ratio of the total and the expected count for a random source model.

Index of Coincidence

- The index of coincidence (IC): the probability of having two identical letters from the text is.

$$IC = \frac{\sum_{i=1}^{n} f_i\left(f_i - 1\right)}{N(N-1)}$$

  Where f_i is the frequency count of *i*th letter in the ciphertext of length N.
- $IC_{\text{English}} = 0.0686$, $IC_{\text{Random}} \approx 1/26 = 0.038466$
- For a ciphertext encrypted by a monoalphabetic cipher IC will be the same as for the original plaintext
- For polyalphabetic ciphers (like Vigenère) it is between IC_{English} and IC_{Random}.

Finding length of the key

- This procedure of breaking up the ciphertext and calculating the I.C. for each subsequence is repeated for all the key lengths we wish to test.
- If IC for a particular length say *k* is very close to $IC_{English}$ stop and declare the length of the key is *k*.

Example: Vigenère Cipher

- Vigenere cipher of size 313 characters

CHREEVOAHMAERATBIAXXWTNXBEEOPH
BSBQMQEQERBWRVXUOAKXAOSXXWEAHB
WGJMMQMNKGRFVGXWTRZXWIAKLXFPSK
AUTEMNDCMGTSXMXBTUIADNGMGPSREL
XNJELXVRVPRTULHDNQWTWDTYGBPHXT
FALJHASVBFXNGLL**CHR**ZBWELEKMSJIK
NBHWRJGNMGJSGLXFEYPHAGNRBIEQJT
AMRVLCRREMNDGLXRRIMGNSNRW**CHR**QH
AEYEVTAQEBBIPEEWEVKAKOEWADREMX
MTBHH**CHR**TKDNVRZ**CHR**CLQOHPWQAIIW
XNRMGWOIIFKEE

Finding length by Kasiski Method

- The text CHR, starts at 1, 166, 236, 276 and 286.
- The distances between the occurrences are 10, 70, 110, 120, 165, 235, 275 and 285.
- Thus k = gcd(10, 70, 110, 120, 165, 235, 275, 285) = 5.

Verifying the length of key by IC

CHREEVOAHMAERATBIAXXWTNXBEE
OPHBSBQMQEQERBWRVXUOAKXAOS
XXWEAHBWG

A	B	C	E	G	H	I	K	M	N
7	6	1	8	1	4	1	1	2	1
O	P	Q	R	S	T	U	V	W	X
4	1	3	4	2	2	1	2	4	7

Finding length by IC

Original: CHREEVOAHMAERATBIAXXWTNXBEEOPH…

if key were length 2:

sequence 1: C R E O H A R T I X W N B E P ...

sequence 2: H E V A M E A B A X T X E O H …

if key were length 3:

sequence 1: C E O M R B X T B O ...

sequence 2: H E A A A I X N E P …

sequence 3: R V H E T A W X E H …

- For $k = 1, 2, 3, 4$ IC ≈ 0.04
- For $k = 5$, IC = 0.065 ($\approx IC_{English}$)

Mutual Index of Coincidence

- Suppose x = x_1, x_2 …., x_n, and y = y_1, y_2 …., $y_{n'}$ are strings of n and n' alphabetic characters, respectively. The mutual index of coincidence of x and y, denoted MIC(x, y), is the probability that a random element of x is identical to a random element of y.

$$MIC(\mathrm{x}, \mathrm{y}) = \frac{\sum_{i=0}^{25} f_i f_i'}{nn'}$$

Where f_i and f_i' are the frequency count of ith letter in x and y respectively.

Suppose K = (k_1 , k_2, . . . , k_m) is the keyword.

- To estimate $MIC(y_i, y_j)$

  Consider a random character in y_i and a random character in y_j. The probability that both characters are A is p_{0-k_i} , p_{0-k_j} , the probability that both are B is p_{1-k_i} , p_{1-k_j}, etc.

$$MIC(y_i, y_j) \approx \sum_{h=0}^{25} p_{h-k_i} p_{h-k_j} = \sum_{h=0}^{25} p_h p_{h+k_i-k_j}$$

  The value of this estimate depends only on the difference (k_i – k_j) mod 26, which is called the relative shift of y_i and y_j.

$$\sum_{h=0}^{25} p_h p_{h+l} = \sum_{h=0}^{25} p_h p_{h-l}$$

  i.e. relative shift of l yields the same estimate of *MIC* as does a relative shift of 26 – l.

- Note: If the relative shift is not zero, these estimates vary between 0.031 and 0.045; whereas, a relative shift of zero yields an estimate of 0.065.

- This observation can be used to formulate a likely guess for $l = k_i - k_j$, the relative shift of y_i and y_j, as follows:
  - Suppose we fix y_i and consider the effect of encrypting y_j by e_0, e_1, e_2,. . . . Denote the resulting strings by y_j^0, y_j^1, etc.
  - It is easy to compute the indices $MIC(y_i, y_j^g)$ $0 \le g \le 25$. This can be done using the formula

$$MIC(\mathrm{x}, y^g) = \frac{\sum_{i=0}^{25} f_i f'_{i-g}}{nn'}$$

- When $g = l$, the *MIC* should be close to 0.065, since the relative shift of y_i and y_j^l is zero. However, for values of $g \neq l$, the *MIC* should vary between 0.031 and 0.045.
- In this way, relative shifts of any two of the substrings y_i can be obtained. This leaves only 26 possible keywords, which can easily be obtained by exhaustive key search.

Methodology

- Let keyword length be m.
- Compute values of $MIC(y_i, y_j^g)$, where $1 \leq i < j \leq m$, $0 \leq g \leq 25$.
- For each i and j, look for values of $MIC(y_i, y_j^g)$ that are close to 0.065.
- If there is a unique such value (for a given (i, j) pair), then the value of g is the value of the relative shift. i.e. $k_i - k_j = g$.
- Solve all such equations and with some heuristics/guess/exhaustive key search find all k_i's.

Assignment 1

Cryptanalysis of Vigenere Cipher

- Find the key length by Kasiski method
- Verify the key length by Index of Coincidence.
- Find the actual key using Mutual Index of Coincidence.

Language: C/C++ or Python or Matlab

Last date to submit is 25th October 2020 midnight.