Cryptanalysis of the Vigenere cipher

James Fricker a1706059

October 14, 2019

In submitting this work I am indicating that I have read the University's Academic Honesty Policy. I declare that all material in this assessment is my own work except where there is clear acknowledgement and reference to the work of others.

Abstract

The Vigenere cipher is a method by which messages can be encrypted to prevent third party access. In this report, a ciphertext encrypted with the Vigenere cipher will be decrypted. Methods used include the Kasiski test, Index of Coincidence and Mutual Index of Coincidence.

1 Introduction

Maintaining confidentiality of data in today's world is very important. More than ever, data is sent over networks, which can allow attackers easier access to message content. Cryptography is the study and practice of keeping these messages safe.

Cryptographic techniques are used to make sure data is securely transferred between two parties, without being intercepted by third parties[1].

Typical cryptographic techniques involve processing the message with a key, to produce a ciphertext. This ciphertext should be difficult for an attacker to manipulate to find the original message, without knowledge of the key. Some of these techniques include mono-alphabetic techniques, poly-alphabetic techniques and many others.

The main point of analysis in this report will be on a cipher known as the Vignere cipher. This cipher is a poly-alphabetic cipher and uses a repeated key to encrypt the message.

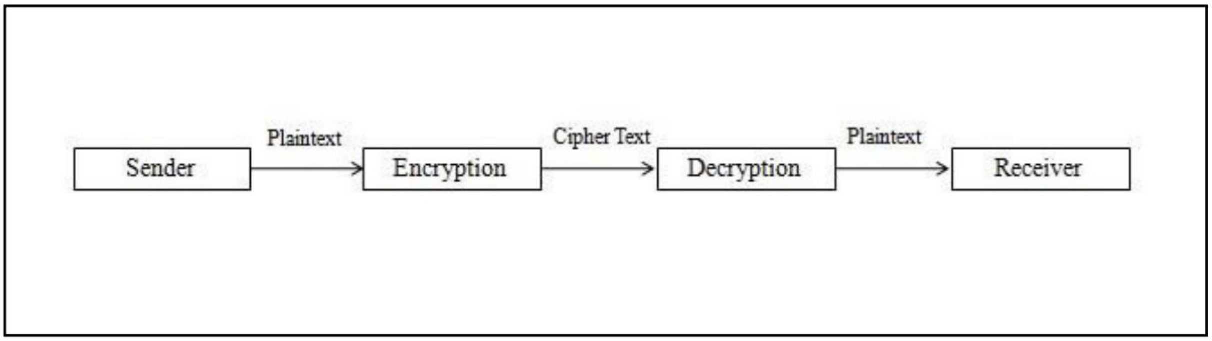


Figure 1: Process of Cryptography [2]

This report will investigate the Vigenere cipher and attempt to find the key and decrypt the message, with only knowledge of the ciphertext. The analysis will be done using the Kasiski test, as well as the Index of Coincidence method. The key will finally be found using the Mutual Index of Coincidence method.

The full code used for the analysis can be found in the appendix.

2 Vigenere cipher

The Vigenere cipher uses a random key to encrypt the message. The key is repeated until it has the same length as the length of the message.
The rules for encryption and decryption of the message are as follows.

Suppose there is an alphabet A = (a_1,a_2,a_3,..a_n,), key with length m K = (k_1,k_2,k_3,..k_m,).

$$E_k(a_i) = (a_i + k_i \mod n)$$
$$D_k(c_i) = (c_i - k_i \mod n)$$

The Vigenere cipher can be used on words, by first converting the letter to it's number format. That is the letter A = 0, B=1, C=2 and so on. This allows mathematical operations to be completed on text based messages. All operations are completed in mod 26.

An example of this is the encryption of CRYPTOGRAPHY with the key, ABC. The key is used to encrypt the entire message.

	C	**R**	**Y**	**P**	**T**	**O**	**G**	**R**	**A**	**P**	**H**	**Y**
	2	17	24	15	19	14	6	17	0	15	7	24
+	**A**	**B**	**C**	**A**	**B**	**C**	**A**	**B**	**C**	**A**	**B**	**C**
	0	1	2	0	1	2	0	1	2	0	1	2
=	2	18	0	15	20	16	6	18	2	15	8	0
	C	**S**	**A**	**P**	**U**	**Q**	**G**	**S**	**C**	**P**	**I**	**A**

Figure 2: Vigenere Cipher Example

In this situation, the ciphertext is CSAPUQGSCPIA.

The Vigenere cipher uses m shift ciphers. Each plaintext letter has the possibility to be up to m different letters in the ciphertext. This means that the key space is 26^m, which makes it infeasible to attempt brute force. There are other methods that can be used to find the keyword, and this will be used in this report to decode the following ciphertext.

```
KOIRZHZQYVVXSQYDWHVGGOTNBOKHWGFIYVVXSTRVMWFQFPCHJBURKHYHBSJWJFEVUWIDQ
OIPTTKKJUROFLPONSJDXAROQIEUJURUISUBJZCRBGLQTFSLYWEJYVZVFHRGNGKDSQVRKF
FXLVCBSWEHYMKZTAZOQWFQRWCHXWJDSIKWJFCBNBJLLBZINQRQYZZWYZVEQIVJWSVQUZR
QJHNKTGVDUSUHXQVQISUONTVITFDVFFVVTODDEWEJQMGUNAZWNJVWMOKWMSPVYWCOYVZQ
PRZJNHROBOKFMSJDWSRSWSKWDBVDYWUHFHYLXDCDSSKKFGFUWOKKJFYDIOGUTPCHRKYLH
VNDXHYLXAFVYCWWMSGHTDCHTBZWBSIHZBYDUDPITFGUJHKBRITKTTKKJHZPJARQDGFOZH
ZRSGNHWSJXLUVVYSUITFKKNGGUTPCHRPLWRCJWTTKKJGVZJFVOFFXHQMTRSQVUSSUZNHY
WMSDRASDHSHJRKGDDQZXUJSESNSTHXCWSFDVUBVZFMWJRIRSHHOLVJCEWMSNKTZVLYKRV
SOKKJGDDQZXUJSESNSTHXCWSFDVUYVRWBSIHZBYDUDPDSRJRYVVSWCSOJAIHROZQJRCRY
GFIYVVSJCGOJKVUJAVDSOEGRCJWTTKKJANHWSDLXSIDGZVHASEWMSFQJGNLYVULLWKDQK
RWHVVVROEBBSIHNBTUJOJLSUCBTTKKJCGLSWFQYVRWYVVBFOCOROUHFPZJRWJWFYVLSQF
PNBXGTKEIWCDWMSKUJSJLSHYHKWIVYDCDHSRQIGFPJGRLIHYDYSMHSHYHYFVHXVRGGSVQ
FPRGRCMHFBUWMOKQTCEHXVFXQRVYJFYDASCHKHKKJCTHFBJDSRKKJBFQJHYXWGUDDBVDW
ZPWBCKKTIJDSRPHFFJDKHVUTBVPFBYDIPVHSBRLQSUWTOKUJSWRWGRBNBXKTKXUJOKLYK
FXQRSHYCSH
```

3 Determine Cipher Period m

There are two main ways to discover the period of the cipher, these are the Kasiski examination, and the Index of Coincidence. These both target the main weakness of the Vigenere cipher, the repetition of the keyword[3].

The Kasiski examination works by finding repeated parts of the ciphertext. If these repeated parts are of significant length, the distance between each is likely to be a multiple of the keyword length. The likelihood of two three-letter sequences not being from the same plaintext fragment is $1/n^3 = 0.0000569$ for $n=26$ [4].

To try and find m, all triplets were calculated and run through the text to see which ones appear most, and where they appear. The string 'KKJ' was found a total of 9 times throughout the ciphertext. 'WMS' appeared 6 times, while 'TTK', 'TKK' and 'YVV' appeared 5 times each.

'KKJ' was first found in position 74.

Starting Position	Distance From Previous	Factors
326	252	2, 3, 4, 6, 7, 9, 12, 14, 18, 21, 28, 36, 42, 63, 84, 126
398	72	2, 3, 4, 6, 8, 9, 12, 18, 24, 36
454	56	2, 4, 7, 8, 14, 28
554	100	2, 4, 5, 10, 20, 25, 50
650	96	2, 3, 4, 6, 8, 12, 16, 24, 32, 48
718	68	2, 4, 17, 34
868	150	2, 4, 37, 74
878	10	2, 3, 4, 6

Figure 3: Kasiski Factor Table

From this analysis, it can be seen that the only factors similar amongst all appearances are 4 and 2. This makes these numbers prime candidates for possible lengths of the keyword. They can both be tested further using the Index of Coincidence (IC) method.

The Index of Coincidence method calculates the probability that randomly chose two elements in a string x, are the same [5].

The probability of any letter being chosen is the number of those letters in the text n_i, divided by the length of the text N. Then the probability of choosing that letter again is (n_i-1) divided by the remaining text length (N-1). The sum of this, for all letters, can be written as the following equation.

$$IC(x) = \frac{\sum_0^{25} n_i(n_i - 1)}{N(N-1)}$$

Suppose we denote Y as the English alphabet, “A,B,C,...Z”. The Index of Coincidence can be calculated using the frequency of each letter.

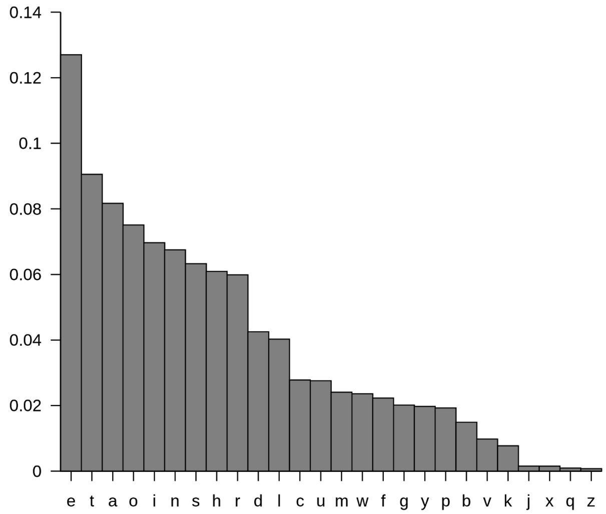


Figure 4: English Letter Frequency Table

Using the letter frequencies, the Index of coincidence of the English language is found to be 0.065.

$$IC(Y) = \sum_0^{25} p_i^2 = 0.065$$

This can now be applied to the key size. If the key size is equal to 4, then there are 4 different simple shift ciphers in the ciphertext. A shift cipher is simply that all letters in the ciphertext have been encrypted with the same letter. For example, it is easy to understand that if the key is ABCD, then, starting from the first letter, every 4th letter will be encrypted with the letter A. This means if each 4th letter is taken out, they are effectively only encrypted using a shift cipher. For example with cipher CIPHERTEXT and key ABCD.

Encrypted with	A	B	C	D
Shift Cipher	CEX	IRT	PT	HE

Figure 5: Key example

Therefore, for example, letters IRT can be examined in the same way as a shift cipher.

A shift cipher will return a similar Index of Coincidence as the English language. It will simply change which letters appear the most, but the sum of the probabilities will remain the same [6]. Therefore if the text is split up using the correct key length, a similar IC to the English language is expected since the text will simply be a permutation of English characters. This works as the ciphertext is of sufficient length. This property will be used to determine the key length.

Starting Index (m=2)	Index of Coincidence
1	0.052892
2	0.056939

Starting Index (m=4)	Index of Coincidence
1	0.067092
2	0.065102
3	0.065506
4	0.072388

Figure 6: IC Calculations

From this analysis, it can be seen that the IC from m=4 is much closer to 0.065 than the m=2 calculations. Therefore it is very likely that the key length m is equal to 4.

Another method for calculating the key length m can be used. This method involves using the Index of Coincidence in a different way. This method will lead to an equation of the key length m, in terms of the IC of the entire text[7].
When splitting up the text into shift ciphers, each has length approximately $\frac{n}{m}$, where n is the length of the ciphertext.

The probability that the next letter comes from the same column is the following.

$$\frac{\frac{n}{m}-1}{n-1}$$

Therefore, the chance that 2 characters are the same and chosen from the same shift cipher is the following.

$$\frac{\frac{n}{m}-1}{n-1}\times 0.065$$

Further, the probability that the 2 characters chosen are the same, but from different shift ciphers is the following.

$$\frac{n-\frac{n}{m}}{n-1}\times 0.038$$

An approximation of the Index of Coincidence can be found by the following.

$$IC(x)\approx\frac{\frac{n}{m}-1}{n-1}\times 0.065+\frac{n-\frac{n}{m}}{n-1}\times 0.038$$

This can be rearranged to create a formula for the key length m [8].

$$m\approx\frac{0.027n}{(n-1)IC+0.065-0.038n}$$

This formula can provide the following table.

m	1	2	3	4	5	6	7	8	9	10	∞
IC	0.0660	0.0520	0.047	0.0449	0.0435	0.0426	0.0419	0.0414	0.0410	0.0407	0.0388

Figure 7: IC Table

The IC of the ciphertext is 0.04426, which is closest to m=4. This confirms the earlier findings that the key length is 4.

4 Finding the Key

Now that the key length has been found, the next step is to find the key. The method used to find the key is called the Mutual Index of Coincidence (MIC). This is similar to the Index of Coincidence used earlier. Instead of calculating the probability of taking the same letter from the same text in IC, the MIC compares taking a letter from two different texts.

The mutual index of coincidence of x and y, denoted MIC(x,y), is defined to be the probability that two random elements, one from each, are identical [9].

This can be shown with the following formula.

$$MIC(x) = \frac{\sum_{i=0}^{25} n_i k_i}{NK}$$

The Mutual Index of Coincidence compares taking a letter from the English language and taking a letter from random text. Therefore, the formula can be changed to include the probabilities of English letters p_i.

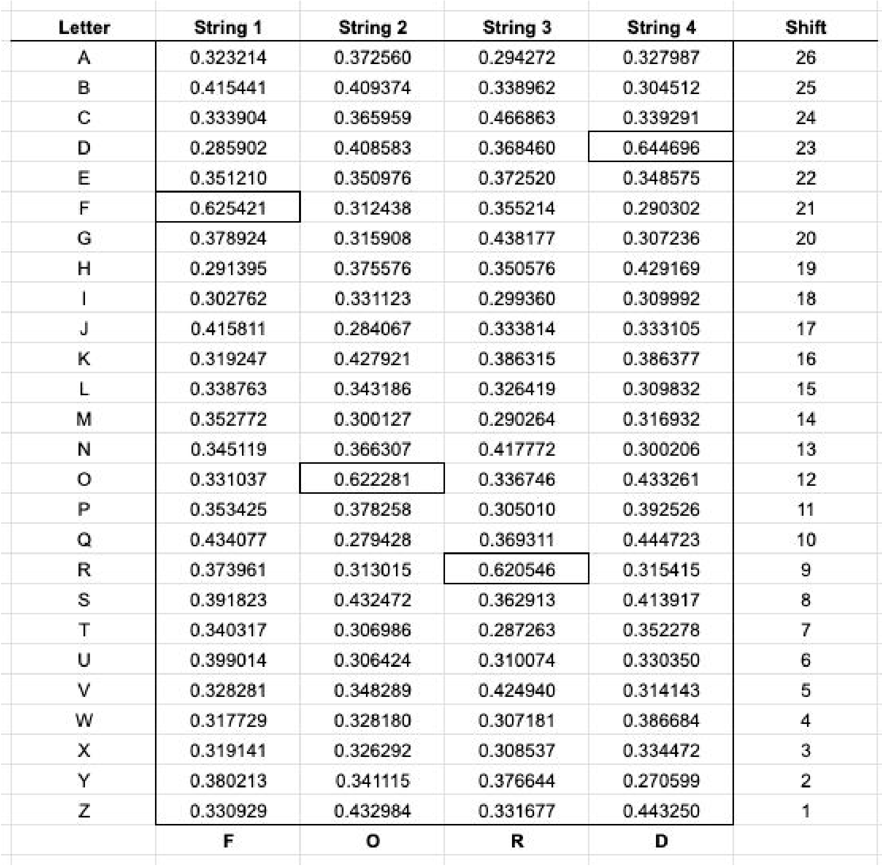
$$MIC(x) = \frac{\sum_{i=0}^{25} p_i k_i}{K}$$

The MIC will be used on the shift ciphertexts that were mentioned in section three. These shift ciphers will be analysed separately. The total ciphertext can now be split into four parts, one for each letter of the keyword which now has known length of 4.

Suppose k_1,k_2,k_3, and k_4 are the shift ciphers of the ciphertext. Each of these is obtained by a shift from the plaintext. Suppose each k is shift by some amount b to get k_n^b for n ={1,2,3,4} and b = {1,2,3,...,26}. The MIC can now be calculated for each shift b. There should be some shift b, that gives an MIC close to the IC of the English language at 0.065.

This would mean that the shifted text is similar to English text and also represents the correct shift to uncover the plaintext. This would also represent the inverse shift of the keyword. The keyword value is calculated by finding the additive inverse of the shift b. This will reveal the characters of the keyword.

Each shift b was completed, and the resulting MIC was calculated. These can be seen in Figure 8.



Letter	String 1	String 2	String 3	String 4	Shift
A	0.323214	0.372560	0.294272	0.327987	26
B	0.415441	0.409374	0.338962	0.304512	25
C	0.333904	0.365959	0.466863	0.339291	24
D	0.285902	0.408583	0.368460	0.644696	23
E	0.351210	0.350976	0.372520	0.348575	22
F	0.625421	0.312438	0.355214	0.290302	21
G	0.378924	0.315908	0.438177	0.307236	20
H	0.291395	0.375576	0.350576	0.429169	19
I	0.302762	0.331123	0.299360	0.309992	18
J	0.415811	0.284067	0.333814	0.333105	17
K	0.319247	0.427921	0.386315	0.386377	16
L	0.338763	0.343186	0.326419	0.309832	15
M	0.352772	0.300127	0.290264	0.316932	14
N	0.345119	0.366307	0.417772	0.300206	13
O	0.331037	0.622281	0.336746	0.433261	12
P	0.353425	0.378258	0.305010	0.392526	11
Q	0.434077	0.279428	0.369311	0.444723	10
R	0.373961	0.313015	0.620546	0.315415	9
S	0.391823	0.432472	0.362913	0.413917	8
T	0.340317	0.306986	0.287263	0.352278	7
U	0.399014	0.306424	0.310074	0.330350	6
V	0.328281	0.348289	0.424940	0.314143	5
W	0.317729	0.328180	0.307181	0.386684	4
X	0.319141	0.326292	0.308537	0.334472	3
Y	0.380213	0.341115	0.376644	0.270599	2
Z	0.330929	0.432984	0.331677	0.443250	1
	F	O	R	D	

Figure 8: Mutual Index of Coincidence Table

As can be seen in the table, there is exactly one shift that results in a Mutual Index of Coincidence close to 0.065 for all shift ciphers. This shift can be converted back into text, which leaves the keyword FORD.

Using this keyword to decrypt the ciphertext leads to the following plaintext.

```
FAR OUT IN THE UNCHARTED BACKWATERS OF THE UNFASHIONABLE END OF THE
WESTERN SPIRAL ARM OF THE GALAXY LIES A SMALL UNREGARDED YELLOW SUN
ORBITING THIS AT A DISTANCE OF ROUGHLY NINETY TWO MILLION MILES IS AN
```

```
UTTERLY INSIGNIFICANT LITTLE BLUE GREEN PLANET WHOSE APE DESCENDED
LIFE FORMS ARE SO AMAZINGLY PRIMITIVE THAT THEY STILL THINK DIGITAL
WATCHES ARE PRETTY NEAT …
```

5 Conclusions

Using the Kasiski method, the Index of Coincidence and the Mutual Index of Coincidence methods, the ciphertext has been successfully decrypted. The strength of the Vigenere cipher could be increased if a larger keyword was chosen. Further analysis in this area could include creating an algorithm to solve ciphertexts using this method[3], or using various other algorithms such as Cuckoo Search [10], genetic algorithms [11] and other hybrids[12][13].

6 References

[1] Rivest, Ronald L. (1990). "Cryptography". In J. Van Leeuwen (ed.). Handbook of Theoretical Computer Science. 1. Elsevier.

[2] Saraswat, A., Khatri, C., Thakral, P., & Biswas, P. (2016). An Extended Hybridization of Vigenere and Caesar cipher techniques for secure communication. Procedia Computer Science, 92, 355-360.

[3] Dalkilic, M. E., & Gungor, C. (2000). An Interactive Cryptanalysis Algorithm for the Vigenere Cipher. Lecture Notes in Computer Science, 341–351.

[4] Beker, H., & Piper, F. (1982). Cipher systems: the protection of communications. Northwood Books.

[5] D. R. Stinson, (1995). Cryptography: Theory and Practice. CRC Press.

[6] B. A. Forouzan and D. Mukhopadhyay.(2014). Cryptography and Network Security, TMH, 2nd Edition.

[7] Beker, H., & Piper, F. (1982). Cipher systems: the protection of communications. Northwood Books.

[8] Chris Christensen. (2015). Cryptanalysis of the Vigenère Cipher: The Friedman Test.

[9] D. R. Stinson. (1997). A More Efficient Method of Breaking a Vigenere Cipher. unpublished manuscript.

[10] Bhateja, A. K., Bhateja, A., Chaudhury, S., & Saxena, P. K. (2015). Cryptanalysis of vigenere cipher using cuckoo search. Applied Soft Computing, 26, 315-324.

[11] Bhateja, A., & Kumar, S. (2014). Genetic Algorithm with elitism for cryptanalysis of Vigenere cipher. 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT).

[12] Kester, Q. A. (2013). A Hybrid Cryptosystem based on Vigenere cipher and Columnar Transposition cipher. arXiv preprint arXiv:1307.7786.

[13] Saraswat, A., Khatri, C., Thakral, P., & Biswas, P. (2016). An Extended Hybridization of Vigenere and Caesar cipher techniques for secure communication. Procedia Computer Science, 92, 355-360.