Lecture 2
Classical Cryptosystems

Shift cipher

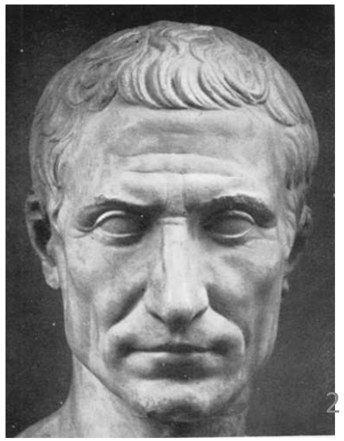
Substitution cipher

Vigenère cipher

Hill cipher

Shift Cipher

- A Substitution Cipher
- The Key Space:
  - [0 … 25]
- **Encryption** given a key K:
  - each letter in the plaintext P is replaced with the K'th letter following the corresponding number (**shift right**)
- **Decryption** given K:
  - **shift left**
- History: K = 3, Caesar's cipher



Shift Cipher

- Formally:
- Let P=C=K=Z_{26} For 0≤K≤25

  $e_k(x) = x$+K mod 26

  and

  $d_k(y) = y$-K mod 26

  $(x, y \in Z_{26})$

Shift Cipher: An Example

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

- P = CRYPTOGRAPHYISFUN
- K = 11
- C = NCJAVZRCLASJTDQFY
- C → 2; 2+11 mod 26 = 13 → N
- R → 17; 17+11 mod 26 = 2 → C
- …
- N → 13; 13+11 mod 26 = 24 → Y

Note that punctuation is often eliminated

Shift Cipher: Cryptanalysis

- Can an attacker find K?
  - YES: exhaustive search, key space is small (<= 26 possible keys).
  - Once K is found, very easy to decrypt

Exercise 1: decrypt the following ciphertext
hphtwwxppelextoytrse

Exercise 2: decrypt the following ciphertext
jbcrclqrwcrvnbjenbwrwn

VERY useful MATLAB functions can be found here:
http://www2.math.umd.edu/~lcw/MatlabCode/

General Mono-alphabetical Substitution Cipher

- The key space: all possible permutations of

$$\Sigma = \{A, B, C, \ldots, Z\}$$

- Encryption, given a key (permutation) π:
  - each letter X in the plaintext P is replaced with $\pi(X)$
- Decryption, given a key π:
  - each letter Y in the ciphertext C is replaced with $\pi^{-1}(Y)$
- Example

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
π	B	A	D	C	Z	H	W	Y	G	O	Q	X	S	V	T	R	N	M	S	K	J	I	P	E	F	U

- **BECAUSE → AZDBJSZ**

Strength of the General Substitution Cipher

- Exhaustive search is now infeasible
  - key space size is $26! \approx 4*10^{26}$
- Dominates the art of secret writing throughout the first millennium A.D.
- Thought to be unbreakable by many back then

Affine Cipher

- The Shift cipher is a special case of the Substitution cipher where only 26 of the 26! possible permutations are used
- Another special case of the substitution cipher is the **Affine cipher**, where the **encryption** function has the form

$$e(x) = ax+b \bmod 26 \qquad (a, b \in Z_{26})$$

- Note that with a=1 we have a Shift cipher.
- **When decryption is possible ?**

Affine Cipher

- Decryption is possible if the affine function is *injective*
- In order words, for <u>any</u> y in Z_{26} we want the congruence

$$ax+b \equiv y \pmod{26}$$

to have a <u>unique solution</u> for x.

- This congruence is equivalent to

$$ax \equiv y-b \pmod{26}$$

- Now, as y varies over Z_{26}, so, too, does $y-b$ vary over Z_{26} Hence it suffices to study the congruence

$$\mathbf{ax \equiv y \pmod{26}}$$

Affine Cipher

$ax \equiv y$ (mod26)

- This congruence has a unique solution *for every y* <u>if and only if</u> gcd(a,26)=1 (i.e.: a and 26 are *relatively prime*)
- *gcd=greatest common divisor*

- Suppose that gcd(a,26)=d>1

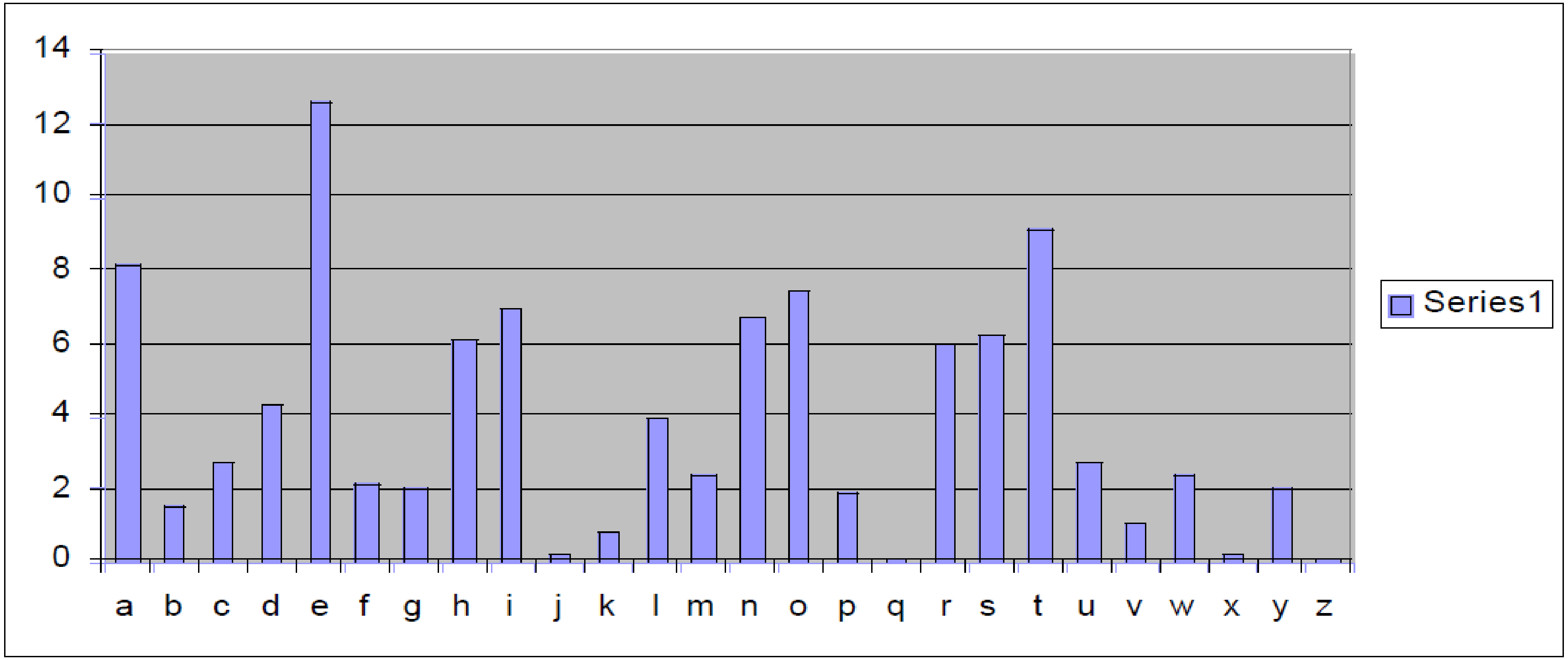
  for example gcd(4,26)=2
- *e(x) = 4x+7 mod 26* is NOT a valid encryption function
- For example, both 'a' and 'n' encrypt to H
  (more in general: x and x+13 will encrypt to the same value)

⇒ affinecrypt('a',4,7)=affinecrypt('n',4,7)='h'

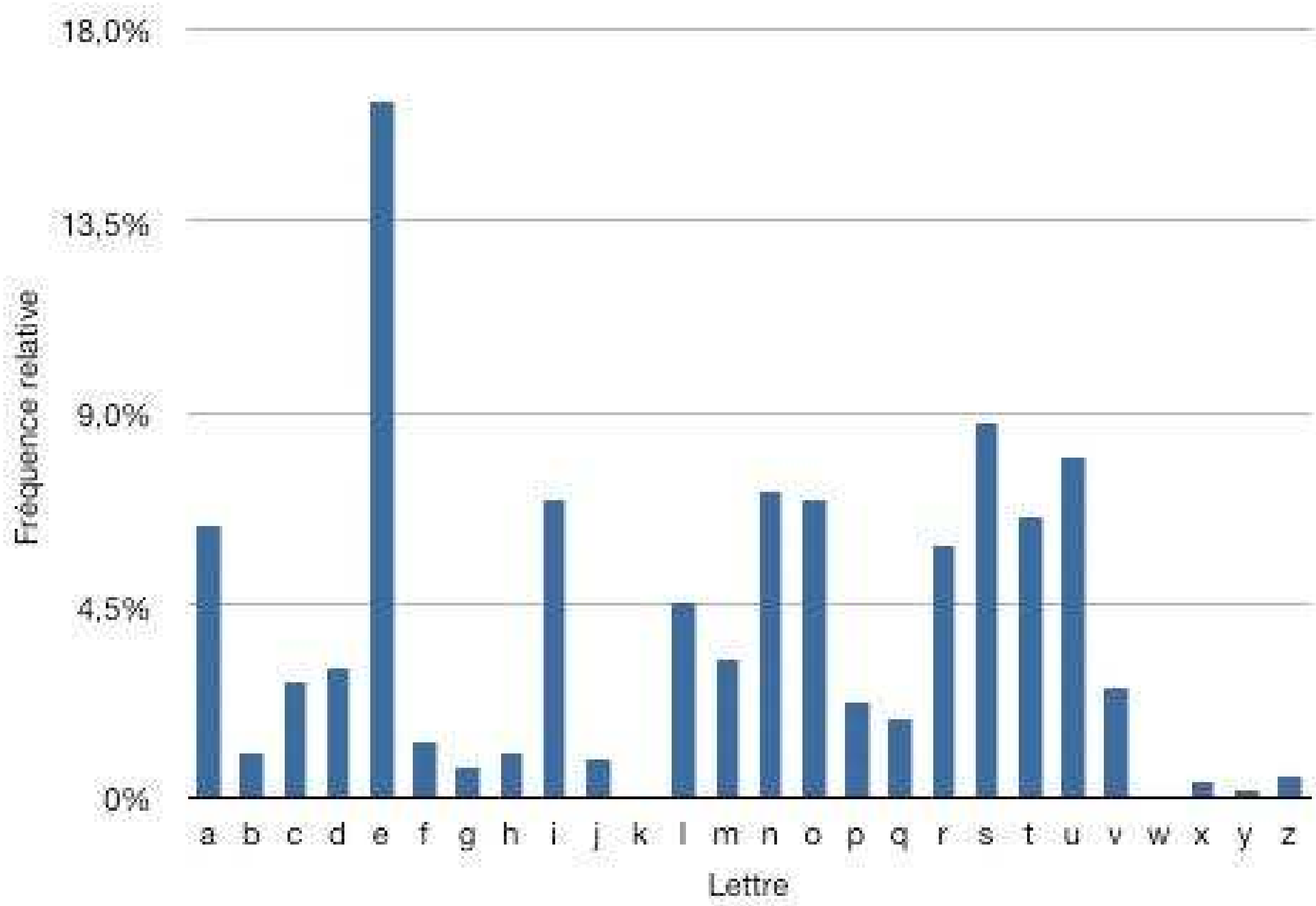
Cryptanalysis of Substitution Ciphers: Frequency Analysis

- Basic ideas:
  - Each language has certain features: frequency of letters, or of groups of two or more letters.
  - Substitution ciphers preserve the language features.
  - *Substitution ciphers are vulnerable to frequency analysis attacks.*

Frequency of Letters in English



Frequency of Letters in French



Other Frequency Features of English

- Vowels, which constitute 40 % of plaintext, are often separated by consonants.
- Letter “**A**” is often found in the beginning of a word or second from last.
- Letter “**I**” is often third from the end of a word.
- Letter “**Q**” is followed only by “**U**”
- And more …

Substitution Ciphers: Cryptanalysis

- The number of different ciphertext characters or combinations are counted to determine the frequency of usage.
- The cipher text is examined for patterns, repeated series, and common combinations.
- Replace ciphertext characters with possible plaintext equivalents using known language characteristics

Frequency Analysis History

- Earliest known description of frequency analysis is in a book by the ninth-century scientist al-Kindi
- Rediscovered or introduced in Europe during the Renaissance
- *Frequency analysis made substitution cipher insecure*

Improve the Security of the Substitution Cipher

- Using nulls
  - e.g., using numbers from 1 to 99 as the ciphertext alphabet, some numbers representing nothing are inserted randomly
- Deliberately misspell words
  - e.g., “Thys haz thi ifekkt off diztaughting thi ballans off frikwenseas”
- Homophonic substitution cipher
  - each letter is replaced by a variety of substitutes
- These make frequency analysis more difficult, but not impossible

Summary

- Shift ciphers are easy to break using brute force attacks, they have small key space.
- Substitution ciphers preserve language features and are vulnerable to frequency analysis attacks.

Towards the Polyalphabetic Substitution Ciphers

- Main weaknesses of monoalphabetic substitution ciphers
  - each letter in the ciphertext corresponds **to only one letter** in the plaintext
  - Idea for a stronger cipher (1460’s by Alberti)
    - use more than one cipher alphabet, and switch between them when encrypting different letters
- Developed into a practical cipher by Vigenère (published in 1586)

The Vigenère Cipher

- **Definition:**

  Given m, a positive integer, $P = C = (Z_{26})^n$, and $K = (k_1, k_2, \dots, k_m)$ a key, we define:

- **Encryption:**

  $e_k(p_1, p_2 \dots p_m) = (p_1+k_1, p_2+k_2 \dots p_m+k_m) \pmod{26}$

- **Decryption:**

  $d_k(c_1, c_2 \dots c_m) = (c_1-k_1, c_2-k_2 \dots c_m-k_m) \pmod{26}$

- **Example:**

Plaintext:	C	R	Y	P	T	O	G	R	A	P	H	Y
Key:	L	U	C	K	L	U	C	K	L	U	C	K
Ciphertext:	N	L	A	Z	E	I	I	B	L	J	J	I

Vigenère Square

Plaintext:

CRYPTOGRAPHY

Key:

LUCKLUCKLUCK

Ciphertext:

NLAZEIIBLJJI

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Security of Vigenere Cipher

- Vigenere *masks the frequency* with which a character appears in a language: one letter in the ciphertext corresponds to multiple letters in the plaintext. Makes the *use of frequency analysis more difficult*.
- Any message encrypted by a Vigenere cipher is a collection of as *many shift ciphers* as there are letters in the key.

Vigenere Cipher: Cryptanalysis

- Find the *length of the key*.
- *Divide* the message into that many shift cipher encryptions.
- *Use frequency analysis* to solve the resulting shift ciphers.
  - how?

How to Find the Key Length?

- For Vigenere, as the length of the keyword increases, the letter frequency shows less English (or French)-like characteristics and becomes more random (when key length -> infinite, see One Time Pad).
- Two methods to find the key length:
  - Kasisky test
  - Index of coincidence (Friedman)

Kasisky Test

- (First described in 1863 by Friedrich Kasiski)
- Note: two identical segments of plaintext, will be encrypted to the same ciphertext, if they occur in the text at a distance Δ, ($\Delta \equiv 0 \pmod{m}$), m is the key length).
- Algorithm:
  - Search for pairs of identical segments of length at least 3
  - Record distances between the two segments: $\Delta 1$, $\Delta 2$, …
  - m divides $\gcd(\Delta 1, \Delta 2, \ldots)$

Example of the Kasisky Test

- Key:

  K I N G K I N G K I N G K I N G K I N G K I N G
- Plaintext:

  t h e s u n a n d t h e m a n i n t h e m o o n
- Ciphertext:

  D P R Y E V N T N **<u>B U K</u>** W I A O X **<u>B U K</u>** W W B T

8 positions

The lenght of the keyworld *probably* divides 8 evenly

(e.g. it may be 2, 4 or 8)

Index of Coincidence (Friedman)

- **Informally:** Measures the probability that two random elements of the n-letters string **x** are identical.
- **Definition:**

  Suppose **x** = $x_1x_2 \ldots x_n$ is a string of n alphabetic characters. Then, the index of coincidence of **x**, denoted I_c(**x**), is defined to be the probability that two random elements of **x** are identical.

Index of Coincidence (cont.)

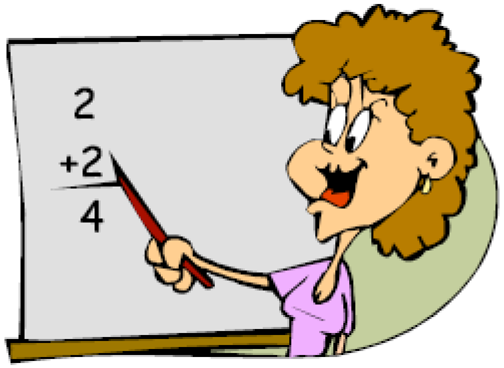
- Reminder: binomial coefficient

$$\binom{n}{k} = \frac{n!}{k!\left(n-k\right)!}$$

- It denotes the number of ways of choosing a subset of k objects from a set of n objects.
- Suppose we denote the frequencies of A, B, C … Z in **x** by f_0, f_1, … f_{25} (respectively).

- We want to compute I_c(**x**)

Begin Math



Elements of Probability Theory

- A random experiment has an unpredictable outcome.
- **Definition**

  The *sample space (S)* of a random phenomenon is the *set of all outcomes* for a given experiment.
- **Definition**

  The *event (E) is a subset of a sample space*, an event is any collection of outcomes.

Basic Axioms of Probability

- If *E* is an event, *Pr(E) is the probability that event E* occurs, then
  - 0 ≤ Pr(A) ≤ 1 for any set ***A in S.***
  - Pr(S) = 1 , where S is the sample space.
  - If E_1, E_2, … E_n is a sequence of mutually exclusive events, that is $E_i \cap E_j = 0$, for all ***i ≠ j*** then:

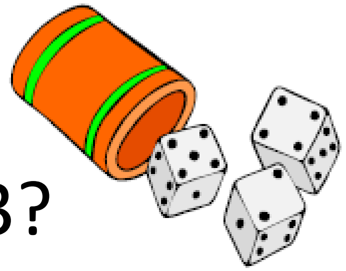
$$\Pr\left(E_1 \cup E_2 \cup \ldots \cup E_n\right) = \sum_{i=1}^{n} \Pr\left(E_i\right)$$

Probability: More Properties

- If E is an event and Pr(E) is the probability that the event E occurs then
  - Pr(Ê) = 1 - Pr(E) where Ê is the complimentary event of E
  - If outcomes in S are equally like, then

    Pr(E) = |E| / |S|
    (where |S| denotes the cardinality of the set S)

Example



- Random throw of a pair of dice.
- What is the probability that the sum is 3?

  Solution: Each dice can take six different values {1,2,3,4,5,6}. The number of possible events (value of the pair of dice) is 36, therefore each event occurs with probability 1/36.

  Examine the sum: 3 = 1+2 = 2+1

  The probability that the sum is 3 is 2/36.

- What is the probability that the sum is 11?

End Math

Index of Coincidence (cont.)

- Reminder: binomial coefficient

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

- It denotes the number of ways of choosing a subset of k objects from a set of n objects.

- Suppose we denote the frequencies of A, B, C … Z in **x** by f_0, f_1, … f_{25} (respectively).
- We want to compute I_c(**x**)

Index of Coincidence (cont.)

- We can choose two elements of **x** (whose size is n) in $\binom{n}{2}$ ways.

  Example: if n=3, nchoosek(3,2)=3; there are 3 ways of choosing couples of n=3 items. *Example*: string ABC

- For each *i* in [0...25], there are $\binom{f_i}{2}$ ways of choosing both elements to be *i*. Hence we have the formula

$$I_c(x) = \frac{\sum_{i=0}^{25} \binom{f_i}{2}}{\binom{n}{2}} = \frac{\sum_{i=0}^{25} f_i(f_i - 1)}{n(n-1)}$$

Example: IC of a String

- Consider the text

  x= "**THEINDEXOFCOINCIDENCE**"

$$I_c(x) = \frac{\sum_{i=0}^{25} f_i(f_i - 1)}{n(n-1)}$$

3 C
2 D
4 E
1 F
1 H
3 I
3 N
2 O
1 T
1 X

- There are 21 characters, with frequencies
- I_c = (3*2+ 2*1+ 4*3+ 1*0+ 1*0+ 3*2+ 3*2+ 2*1+ 1*0+ 1*0) / 21*20 = 34/420 = 0.0809

Index of Coincidence (cont.)

- Now, if we suppose that *n* is very big (e.g., we take all words in the English dictionary), then we can further approximate the formula:

$$I_c(x) = \frac{\sum_{i=0}^{25} \binom{f_i}{2}}{\binom{n}{2}} = \frac{\sum_{i=0}^{25} f_i(f_i - 1)}{n(n-1)} \approx \frac{\sum_{i=0}^{25} f_i^{\,2}}{n^2} = \sum_{i=0}^{25} p_i^2$$

These are the real frequencies of letters in English (see Table)

THIS IS AN APPROXIMATION IF *n* is VERY BIG

Example: IC of a Language

- For English, p_i can be estimated as follows

Letter	p_i	Letter	p_i	Letter	p_i	Letter	p_i
A	0.082	H	0.061	O	0.075	V	0.010
B	0.015	I	0.070	P	0.019	W	0.023
C	0.028	J	0.002	Q	0.001	X	0.001
D	0.043	K	0.008	R	0.060	Y	0.020
E	0.127	L	0.040	S	0.063	Z	0.001
F	0.022	M	0.024	T	0.091		
G	0.20	N	0.067	U	0.028		

$$I_c(x) = \sum_{i=0}^{25} p_i^2 = 0.065$$

IC of a ciphertext

- Now, the same reasoning applies if **x** is a ciphertext obtained by means of any monoalphabetic cipher. In this case, the individual probabilities will be permuted, BUT the quantity

$$I_c(x) = \sum_{i=0}^{25} p_i^2 = 0.065$$

will be *unchanged!*

Find the Key Length

- For Vigenere, as the length of the keyword increases, the letter frequency shows less English-like characteristics and becomes more random.
- Two methods to find the key length:
  - Kasisky test
  - Index of coincidence (Friedman)

Finding the Key Length

- Suppose we start with a ciphertext $q = q_1 q_2 \ldots q_n$
- Define m substrings $y_1 \ldots y_m$ as follows

$$\begin{array}{cc} \begin{bmatrix} q_1 & q_{m+1} & & q_{n-m+1} \\ q_2 & q_{m+2} & & q_{n-m+2} \\ \cdots & \cdots & & \\ q_m & q_{2m} & & q_n \end{bmatrix} & \begin{array}{c} y_1 \\ y_2 \\ \\ y_m \end{array} \end{array}$$

Finding the Key Length

- In our previous example, supposing we already guessed, n=12, m=4

Plaintext:	C	R	Y	P	T	O	G	R	A	P	H	Y
Key:	L	U	C	K	L	U	C	K	L	U	C	K
Ciphertext:	N	L	A	Z	E	I	I	B	L	J	J	I

$$\begin{bmatrix} q_1 & q_{m+1} & & q_{n-m+1} \\ q_2 & q_{m+2} & & q_{n-m+2} \\ \cdots & \cdots & & \\ q_m & q_{2m} & & q_n \end{bmatrix} \quad \begin{matrix} y_1 = \text{CTA} \\ y_2 = \text{ROP} \\ = \text{YGH} \\ y_m = \text{PRY} \end{matrix}$$

Guessing the Key Length

- If this is done, and *m* is indeed the key length, then each $I_c(y_i)$ should be roughly equal to 0.065 (e.g. it will "look like" **English** text)

$$I_c(y_i) = \sum_{i=0}^{25} p_i^2 = 0.065 \quad \forall 1 \le i \le m$$

- If m is not the key length, the text will "look like" much more **random**, since it is obtained by shift encryption with different keys. Observe that a completely random string will have:

$$I_c(x) \approx \sum_{i=0}^{25} \left(\frac{1}{26}\right)^2 = 26 \cdot \frac{1}{26^2} = \frac{1}{26} = 0.0385 \quad \forall 1 \le i \le m$$

Guessing the Key Length

- For French language, the index of coincidence is approximately 0.0778
- The values 0.065 (or 0.0778 for French) and 0.0385 are sufficiently far apart that we will often be able to determine the correct keyword length (or confirm a guess that has already been made using the Kasiski test)

Finding the Key, if Key Length Known

- Consider vectors y_i, and look for the most frequent letter
- Check if mapping that letter to *e* will not result in unlikely mapping for other letters
- Use *mutual index of coincidence* between two strings
  - To determine relative shifts, and hence the key

Summary

- Vigenère cipher is vulnerable:

  once the key length is found, a cryptanalyst can apply frequency analysis.

The Hill Cipher

- Use *linear equations*
  - each output bit (ciphertext, C) is a linear combination of the input bits (plaintext message, M)
  - the key *k is a matrix*
    - $C = k\, M$
    - $M = k^{-1}\, C$
  - known as the *Hill cipher*
  - easily breakable by known-plaintext attack

The Hill Cipher

- It's another polyalphabetic cryptosystem, invented in 1929 by Lester S. Hill.
- Let m be a positive integer (we will see an example with m=2), and define $P=C=(Z_{26})^m$
- The idea is to take m linear combinations of the m alphabetic characters in one plaintext element, thus producing the m alphabetic characters in one ciphertext element.

The Hill Cipher

- Example with m=2
- We can write a plaintext element as $x=(x_1,x_2)$ and a ciphertext element as $y=(y_1,y_2)$.
- Here y_1 would be a linear combination of x_1 and x_2, as would be y_2
- We might take

$$y_1 = 11x_1 + 3x_2$$
$$y_2 = 8x_1 + 7x_2$$

All computed Mod 26

The Hill Cipher

- We might take

  $y_1 = 11x_1 + 3x_2$

  $y_2 = 8x_1 + 7x_2$

- Of course, this can be written more succinctly in matrix notation as follows:

  (y_1, y_2)=$(x_1, x_2)\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}$

- In general, we will take an m x m matrix K as our key. We will write y=xK
- The ciphertext is obtained from the plaintext by means of a linear transformation.

The Hill Cipher (Decryption)

- To decrypt, we should multiply both sides for the inverse of K, K^{-1}:
  - $yK^{-1}=xKK^{-1}$
  - hence $x=yK^{-1}$
- Does K^{-1} always exist ? Of course not!
- By definition, the *inverse matrix* to an m x m matrix K (if it exists) is the matrix K^{-1} such that $K K^{-1} = I_m$
- For example:

$$I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- We can verify that the encryption matrix above has an inverse modulo 26

$$\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{7} & \mathbf{18} \\ \mathbf{23} & \mathbf{11} \end{pmatrix} \Rightarrow \begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix} \begin{pmatrix} 7 & 18 \\ 23 & 11 \end{pmatrix} = \begin{pmatrix} 261 & 286 \\ 182 & 131 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

The Hill Cipher (example)

- The key is $K = \begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}$
- From the computation above $K^{-1} = \begin{pmatrix} 7 & 18 \\ 23 & 11 \end{pmatrix}$
- We want to encrypt the plaintext *july*
  - Hence we have two elements of plaintext to encrypt: (9,20), corresponding to *ju* and (11,24) corresponding to *ly*
- We compute as follows:

$$(9,20)\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix} = (99 + 60, 72 + 140) = (3,4)$$

DE

$$(11,24)\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix} = (121 + 72, 88 + 168) = (11,22)$$

LW

DELW

The Hill Cipher (example)

- Verify that DELW decrypts to *july* using the matrix $K^{-1} = \begin{pmatrix} 7 & 18 \\ 23 & 11 \end{pmatrix}$

The Hill Cipher

- Now the question is: when K is invertible?
- The invertibility of a matrix depends on the value of its determinant (det K $=k_{11}k_{22}-k_{12}k_{21}$)
- We know that a *real* matrix K has an inverse if and only if its determinant is non-zero
- However, it is important to remember that we are working over Z_{26}
- The relevant result for our purposes is that a matrix K has an inverse modulo 26 if and only if gcd(det K, 26)=1
  – In our example, det K=53 (mod26) = 1 and gcd(1,26)=1

The Hill Cipher

- How to compute K^{-1} (when it exists, of course)?
- Recall that det K $=k_{11}k_{22}-k_{12}k_{21}$
- It can be shown that:

$$K^{-1} = (detK)^{-1}\begin{pmatrix} k_{22} & -k_{12} \\ -k_{21} & k_{11} \end{pmatrix}$$

- In our example
  - det K = 53 (mod 26)=1
  - Now, 1^{-1}mod 26=1
  - Hence

$$K^{-1} = 1\begin{pmatrix} 7 & -8 \\ -3 & 11 \end{pmatrix} mod26 = \begin{pmatrix} 7 & 18 \\ 23 & 11 \end{pmatrix}$$