187 - Day 02

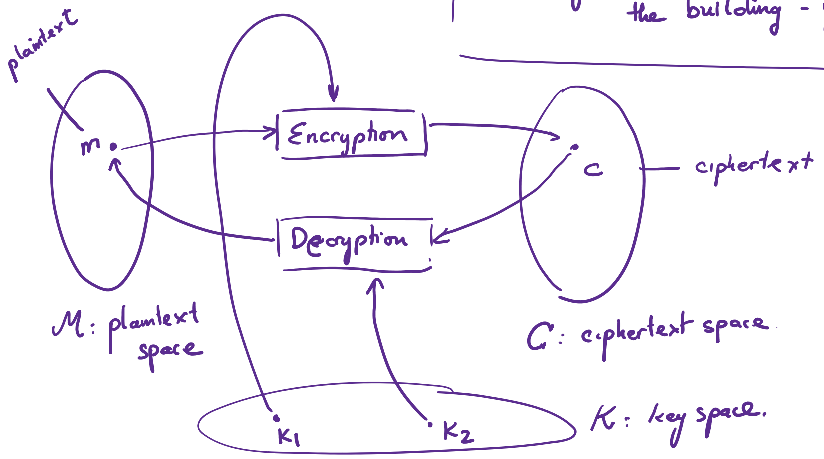
Wednesday, April 4, 2018 7:45 AM

Day 02 – Classical Cryptosystems

Last time

- Terminology of cryptography

- TA's office hours are posted.
- Q.B.'s OH:
  Friday 3-5pm in APM B402A.
  Saturday 11:30 am - 12:30pm
  in APM 5829
  (use your Student ID to get in the building - if locked)



- Caesar Cipher:

A → E
B → F
C → G
⋮

- keep the order of the alphabet
- 26 keys

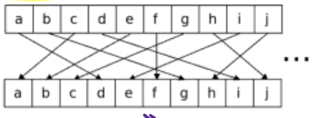
keys = 26 ways to shift "a"
25 ways to shift "b"
24 ________ "c"
⋮
1 ________ "z"

$26! = 1 \cdot 2 \cdot 3 \cdots 26$

b/w 2^{88} and 2^{89}

Improvements for Caesar ciphers

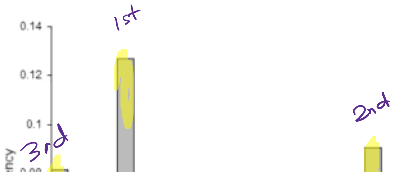
1. **Generalized mono-alphabetic substitution:** randomize the order of substitution:

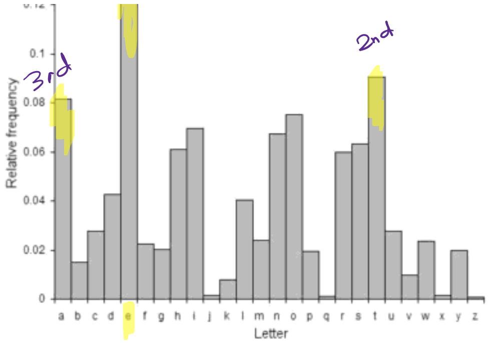


"count the letters"

Vulnerable to frequency analysis: Both plaintext and ciphertext still follow the frequency distribution of the letters.

- Take a long text (Declaration of Indep.) Count # of each letter





(Declaration of Indep.)
Count # of each letter
Divide by total # of letter
to get the frequency.

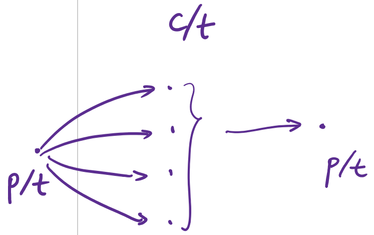
Remark: If there's a 1-1 corres. b/w plaintext letters and ciphertext letters → then freq. analysis will break it.

2. **Homophonic substitution:** assign more than one ciphertext symbol for each plaintext letter. **The more** common letters will have more possible replacements

a	b	c	d	e	f	g	h	i	j	k	l	m	n
! † ↙ Θ	4 ∘	# ξ	$ **N**	1 6 ⇓ ⇑	% ↗	& ♡ ○	∗ ♯	(♭ ∅	)	3	2 ↘	= ◁	+ ▷ ♠

o	p	q	r	s	t	u	v	w	x	y	z
[8 ∞	9 ♣	]	{ Ω	} ⋆ ♮ ∉	: ÷ □ ↖	; ∪ ℏ	7	<	>	5 ≬	?

$$\begin{array}{cccccccccc} T & E & N & N & E & E & S & S & E & E \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ \square & \Downarrow & \triangleright & + & 6 & \Uparrow & \star & \natural & 1 & 1 \\ \div & 1 & ♠ & ♠ & \Uparrow & \Downarrow & \} & \star & 6 & \Downarrow \end{array}$$



- Can fool freq. analysis
- Need a big alphabet.

→

- poly-alphabetic subs.
  1 pt → mult. ct letter.
- entropy of the language
  (↳ later)

p/t: $\underbrace{\qquad\qquad}_{\text{D- shift}} \Big| \underbrace{\qquad\qquad}_{\text{J- shift}} \Big| \underbrace{\qquad\qquad}_{\text{Y- shift.}}$

3. **Poly-alphabetic Substitution:** In 1467, Leon Battista Alberti invented the **cipher disk** which allowed the sender to use different alphabet for different portion of the plaintext.

→ not very good.

In the 1500s Blaise de Vigenère used **multiple Caesar ciphers** to encrypt the data based on a given keyword ⇒ Vigenère cipher.

merge different Caesar shifts together.

Keyword

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
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
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
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

A VE|N GE|D SE|VEN|FOL|D

R E U M W U

Z K W I S

Z Z Z I G

R ZZ EKZ UWZ MII WS GU

Key space = $26^{\text{length of key word.}}$

Security of Vigenère cipher:

- This cipher survived frequency analysis for **three centuries**.
- It was broken in 1863 by Charles Babbage using Kasiski's test in order to deduce the length of the keyword.
- There is another approach using Claude Shannon's index of coincidence.
- In 1700s Thomas Jefferson came up with a cipher system very similar to the Vigenère Cipher except with higher security.
  - This machine uses 26 wheels with a randomized alphabet on each.
  - The wheels can be arranged in any order chosen by the sender.

length tells me how many Caesar ciphers that I have to break.

Rectangular transposition - Row Version

Encryption teps:

1. Prepare a $p \times q$ table where $p \cdot q = n =$ length of the text.
2. Put the letters of the plaintext in the table, row-by-row from left to right, from top to bottom.
3. Rearrange the columns and read the ciphertext the same way as above.

Example: Encrypt: **THE BABOONS ARE COMING FOR YOU** = length 25

3	2	5	1	4
T	H	E	B	A
B	O	O	N	S
A	R	E	C	O
M	I	N	G	F
O	R	Y	O	U

→

1	2	3	4	5
B	H	T	A	E
N	O	B	S	O
C	R	A	O	E
G	I	M	F	N
O	R	O	U	Y

BHTAE NOBSO CRAOE

GIMFN OROUY

What if I have 24 letters ?

6 × 4

4 × 6

8 × 3

3 × 8

12 × 2

2 × 12

24 × 1 → trivial

1 × 24

breaking this comes latter

Playfair cipher

- Invented by Charles Wheatstone popularized by Lyon Playfair
- Used by the British forces in the second Boer war and World War I and by the Australians in World War II

Step 1: construct the Playfair square (key) using the keyword **DIVERGENT** (skip)

# of keys ? 25 !

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

. Treat I & J the same.

. Put letters in the keyword in the table.

. put the rest of the alphabet in the table using alphabetical order

Step 2: Break the plaintext into 2-grams:

- Use a letter 'X' to break up repeated letters
- Put a letter 'Q' at the end if there is an odd number of letter

Step 3: Encrypt the plaintext using the Playfair square.

- When two letters of the 2-gram are in the same row

  shift right

  GT → NA | NB → TG
  TG → AN |

  wrap around.

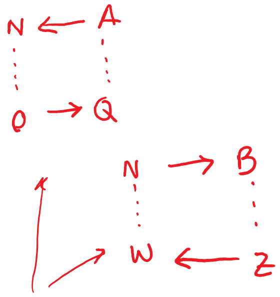
- When two letters of the 2-gram are in the same column

  shift ~~up~~ down

  MG → ~~CO~~ UC ; DC → ~~UG~~ GM , GU → CD

- When two letters of the 2-gram are in the opposite corner of a rectangle

  replace w/ the other corners



AO → NQ
OA → QN
NZ → BW

CHEER → CH | ~~EE~~ | X CH | EX | ER.

HOOD → HO | OD ✓

EGG → EG | GQ