Diffusion and Confusion

Two properties that a good cryptosystem should have:

**Diffusion:** change of one character in the plaintext results in several characters changed in the ciphertext

**Confusion:** the key does not relate in a simple way to the ciphertext (in particular, each character of ciphertext should depend on several parts of the key)

What about the cryptosystems we've seen so far?

No
Block Ciphers

- blocks of letters encrypted simultaneously
- in general, have the diffusion and confusion properties

Simple examples:

The **Playfair cipher** (used in WWI by the British):
- encrypts digrams by digrams (for details see Section 2.6)
  
  problem: we have frequency tables for digrams (see the book) → not safe

The **ADFGX cipher** (used in WWI by the Germans):
- encrypts letters by digrams, followed by permuting the encrypted letters within each block (for details see Section 2.6)

The **Hill cipher**: see next slide (Section 2.7)

Remark: Many modern cryptosystems (DES, AES, RSA) are also block ciphers.
**Hill Cipher**

**Key:** an **invertible** $m \times m$ matrix (where $m$ is the block length) 
[defines a linear transformation]

**Encryption:**
- view a block of $m$ letters as a vector, multiply by the key

**Example:**

Key $K = \begin{pmatrix} 2 & 5 \\ 9 & 4 \end{pmatrix}$

What is $m$? $m = 2$

How to encrypt $\text{blah}$?

Encrypting 'bl':

$(1, 11) \cdot \begin{pmatrix} 2 & 5 \\ 9 & 4 \end{pmatrix} = (2 \cdot 1 + 5 \cdot 11, 9 \cdot 1 + 4 \cdot 11) = (12, 55) \equiv (23, 23) \pmod{26}$

Encrypting 'ah':

$(0, 7) \cdot \begin{pmatrix} 2 & 5 \\ 9 & 4 \end{pmatrix} = (2 \cdot 0 + 5 \cdot 7, 9 \cdot 0 + 4 \cdot 7) = (35, 28) \equiv (11, 2) \pmod{26}$
Hill Cipher

Decrypting:
- multiply each block by $K^{-1}$

How to invert a matrix $K$?
- invertible (mod 26) iff $\gcd(\det(K), 26) = 1$
- if $m=2$ and invertible, then:

$$K^{-1} = \det(K)^{-1} \begin{pmatrix} k_{2,2} & -k_{1,2} \\ -k_{2,1} & k_{1,1} \end{pmatrix}$$

what is $K^{-1} \pmod{26}$: an $m \times m$ matrix s.t.
$$K \cdot K^{-1} \equiv \begin{pmatrix} 1 \\ 0 \end{pmatrix} \pmod{26}$$

- inverting matrices for other values of $m$: see any basic linear algebra text

- invert $K$ for our example

For our example:
$$K = \begin{pmatrix} 2 & 5 \\ 9 & 4 \end{pmatrix}$$

$$K^{-1} \equiv \begin{pmatrix} 4 & -5 \\ -9 & 2 \end{pmatrix}^{-1} \equiv \begin{pmatrix} 28 & -35 \\ -63 & 14 \end{pmatrix} \equiv \begin{pmatrix} 2 & 17 \\ 15 & 14 \end{pmatrix} \pmod{26}$$

$\det(K) = k_{1,1}k_{2,2} - k_{1,2}k_{2,1}$

$\det(K)^{-1} \equiv 7 \pmod{26}$

we hope...
Hill Cipher

Decrypting:
- multiply each block by $K^{-1}$

How to invert a matrix $K$?
- invertible (mod 26) iff $\gcd(\det(K),26)=1$
- if $m=2$ and invertible, then:

$$K^{-1} = \det(K)^{-1} \begin{pmatrix} k_{2,2} & -k_{1,2} \\ -k_{2,1} & k_{1,1} \end{pmatrix}$$

- inverting matrices for other values of $m$: see any basic linear algebra text

what is $K^{-1} \pmod{26}$: an $mxm$ matrix s.t.

$$K \cdot K^{-1} \equiv \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$\pmod{26}$

don’t need to memorize but know that we can compute this

if using a software to compute $K^{-1}$ (e.g. Mathematica, Matlab, Maple)

we will likely get matrices containing real number (not numbers $\in \mathbb{Z}_{26}$)

solution: 0 multiply by the $\det(K)$ -> getting

(Chances $L$ is not an inverse of $K$)

since

$$L \cdot K = \begin{pmatrix} \text{odd} \\ \text{odd} \end{pmatrix}$$

@ divide by $\det(K) \pmod{26}$
Hill Cipher

Remark: The Hill cipher is a generalization of the permutation cipher (permute the letters within each block)

Cryptanalysis:
- hard with ciphertext-only
- easy with known plaintext:
  - suppose we know m:

  e.g. permutation \( m = 3 \)
  \( 1 \to 3 \to 2 \)
  'abc' \( \mapsto \) 'bca'

  instance of Hill:
  \[
  K = \begin{pmatrix}
  0 & 0 & 1 \\
  1 & 0 & 0 \\
  0 & 1 & 0
  \end{pmatrix}
  \]

  example:
  we know
  'be' \( \mapsto \) 'xx'
  we want to find \( K \) s.t.
  \[
  (1, 11) \cdot K \equiv (23, 23) \pmod{26}
  \]
  need to solve the system of congruences

  how many keys do we have?
  if ignoring the gcd requirement,
  then
  \( 26^m \) matrices

  cannot brute force

  - how to find m?
  dry m = 2, m = 3, ... (brute-force)