1 Classical Cryptosystems

(Continued from Lecture 3.)

1.1 Affine Cipher

Recall that the encryption function for the Caesar cipher is given by

$$E(x) = (x + b) \pmod{26},$$

where $b = 0, 1, 2, \ldots, 25$ is the shift. Here, x represents the number associated with the letter (e.g., A is 0, B = 1, C = 2, and so on). We can generalize this to the *affine cipher*. Specifically, an **affine cipher** is one whose encryption function is of the form

$$E(x) = (ax + b) \pmod{26},$$

where a and b are integers which form the key.

(Example.) Suppose that $a = 3$ and $b = 5$. The encryption function is defined by

$$E(x) = (3x + 5) \pmod{26}.$$

Suppose we wanted to encrypt the letter `Y`.

Note that the letter `Y` corresponds to the number `24`. So,

$$E(24) = (3 \cdot 24 + 5) \pmod{26} = (72 + 5) \pmod{26} = 77 \pmod{26} = 25.$$

Therefore, the encryption of `Y` is `Z`, which corresponds to 25.

(Exercise.) Use the same encryption function as above with $a = 3$ and $b = 5$.

(a) What is the encryption of `A`?

Note that `A` corresponds to the number `0`. So,

$$E(0) = (3 \cdot 0 + 5) \pmod{26} = 5 \pmod{26}.$$

Here, the number 5 corresponds to the letter `F`.

(b) What is the encryption of `D`?

`D` corresponds to the number `3`, so

$$E(3) = (3 \cdot 3 + 5) \pmod{26} = 14 \pmod{26}.$$

Here, the number 14 corresponds to the letter `O`.

Lemma 1.1: Affine Cipher

Suppose

$$E : \{0, \dots, 25\} \mapsto \{0, \dots, 25\}$$

is a function of the form

$$E(x) = (ax + b) \pmod{26}$$

for some integers a and b. Then, there exists a function

$$D : \{0, \dots, 25\} \mapsto \{0, \dots, 25\}$$

such that $D(E(x)) = x$ if and only if a is invertible mod 26. Moreover, if $c \equiv a^{-1} \pmod{26}$, then

$$D(y) = c(y - b) \pmod{26}.$$

(Example.) Suppose again $a = 3$ and $b = 5$. Using the process for finding the inverse of a mod 26, we find that this must be 9. So, the Affine Cipher Lemma tells us that the decryption function must be given by

$$D(y) = 9(y - 5) \pmod{26}.$$

Suppose we wanted to decrypt the letter `Z`, which corresponds to the number 25. Then,

$$D(25) = 9(25 - 5) \pmod{26} = 9 \cdot 20 \pmod{26} = 180 \pmod{26} = 24,$$

which corresponds to `Y` as expected.

(Exercise.) Alice and Bob are using the same affine encryption function as above with $a = 3$ and $b = 5$. Bob has just received the message `LNKRLFKH`. Decrypt it.

The letters correspond to the numbers:

$$L \mapsto 11 \qquad N \mapsto 13 \qquad K \mapsto 10 \qquad R \mapsto 17 \qquad F \mapsto 5 \qquad H \mapsto 7.$$

Decrypting each letter results in

- L: $D(11) = 9(11 - 5) \pmod{26} = 9(6) \pmod{26} = 2 \mapsto C$
- N: $D(13) = 9(13 - 5) \pmod{26} = 9(8) \pmod{26} = 20 \mapsto U$
- K: $D(10) = 9(10 - 5) \pmod{26} = 9(5) \pmod{26} = 19 \mapsto T$
- R: $D(17) = 9(17 - 5) \pmod{26} = 9(12) \pmod{26} = 4 \mapsto E$
- F: $D(5) = 9(5 - 5) \pmod{26} = 9(0) \pmod{26} = 0 \mapsto A$
- H: $D(7) = 9(7 - 5) \pmod{26} = 9(2) \pmod{26} = 18 \mapsto S$

Therefore, we have `CUTECATS`, or `cute cats`.

(Exercise.) Suppose the encryption function for an affine cipher is $E(x) = (5x + 17) \pmod{26}$. What is the corresponding decryption function D?

We need to find the inverse of $a = 5 \bmod 26$. So, first, let's find $\gcd(5, 26)$.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
5	26	$26 = 5q + r$	5	1
1	5	$5 = 1q + r$	5	1

Since the GCD is 1, there exists an inverse. Moreover, because we only have one equation with a nonzero remainder, it follows that

$$\gcd(5, 26) = 1 = 26(1) + 5(-5).$$

Therefore, the inverse is $-5 \equiv 21 \pmod{26}$. From here, it follows that the decryption function is

$$D(y) = 21(y - 17) \pmod{26}.$$

Remark: Of the numbers between 0 and 25, there are 12 that are invertible mod 26:

$$\{1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25\}.$$

So, the number of pairs (a, b) such that $E(x) = ax + b \pmod{26}$ is a legitmate encryption function for an affine cipher is $12 \cdot 26 = 312$.

(Exercise.) The *Atbash cipher* is a simple substitution cipher in which encryption and decryption both simply reverse the order of the alphabet. In other words, `A` and `Z` are interchanged, `B` and `Y` are interchanged, and so forth. For example, the plaintext `APPLE` corresponds to the ciphertext `ZKKOV`. Show that the Atbash cipher is a special case of the affine cipher. What are the corresponding values of a and b?

To see why this is a special case of the affine cipher, we need to understand how the affine cipher works. Consider the encryption function

$$E(x) = (ax + b) \pmod{26}.$$

First, let's set $b = 0$. This way, we just need to try all valid values of a. Notice that, when $a = 25$, we have

- $(25 \cdot 0) \pmod{26} = 0$.
- $(25 \cdot 1) \pmod{26} = 25$.
- $(25 \cdot 2) \pmod{26} = 24$.
- $(25 \cdot 3) \pmod{26} = 23$.
- $(25 \cdot 4) \pmod{26} = 22$.
- $\dots$
- $(25 \cdot 24) \pmod{26} = 2$.
- $(25 \cdot 25) \pmod{26} = 1$.

This looked very similar to what the Atbash cipher does, albeit with one of the numbers being off (remember that A is supposed to map to Z, but with $a = 25$ and $b = 0$, A maps to A still). However, at that point, it became kind of obvious that if you set $b = -1 \equiv 25$, you'll end up with the correct values of a and b.

(Exercise.)

(a) Make sense of and justify the following statement: "Two affine ciphers in succession result in just another affine cipher."

Consider

$$E_1(x) = (a_1x + b_1) \pmod{26}$$

and

$$E_2(x) = (a_2x + b_2) \pmod{26}.$$

We note that

$$\begin{aligned} E_1(E_2)(x) &= (a_1(a_2x + b_2) + b_1) \pmod{26} \\ &= a_1a_2x + a_1b_2 + b_1 \pmod{26} \\ &= (a_1a_2x) + (a_1b_2 + b_1) \pmod{26}. \end{aligned}$$

(b) Is it possible for "two affine ciphers in succession" to result in a Caesar cipher? Explain.

Consider $a_1 = a_2 = 1$. Then, from the previous part, we'll end up with

$$E_1(E_2)(x) = x + (b_2 + b_1) \pmod{26}.$$

So, it's possible.

1.2 Simple Substitution

We can use a general **simple substitution cipher**, also known as a *simple monoalphabetic substitution cipher* or *monoalphabetic substitution cipher*, by using a full conversion table as a key. For example, we might use a table like the following:

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

This tells us that

- to *encrypt*, we just need to convert every instance of the top letter to the corresponding bottom letter. For example, encrypting A becomes P, encrypting B becomes V, and so on.
- to *decrypt*, we just need to convert every instance of the bottom letter to the corresponding top letter. For example, decrypting P becomes A, decrypting V becomes B, and so on.

(Example.) Suppose Alice wants to encrypt the message `You must destroy all of the horcruxes!` She starts by encoding the message[a]:

`YOUMUSTDESTRYALLOFTHEHORCRUXES`

Then, she converts each letter using the table:

`XEANAIGWDIGREXPFFECGTDTERJRALDI`

This is the ciphertext she sends to Bob. To decrypt the message, Bob uses the same table backwards.

[a] Removing all spaces, punctuations, and then capitalizing everything.

Notice that, if the entire table is our key, the number of possible keys is 26!, a *huge* number. Despite this, simple substitution can still be broken relatively easily using some ideas from probability theory.

(Exercise.) Using the same table given above, do the following by hand.

(a) Encrypt the message `The moon is pitted with holes!`

Encoding the message gives THEMOONISPITTEDWITHHOLES. Then, we just need to map each letter appropriately.

```
plaintext  T H E M O O N I S P I T T E D W I T H H O L E S
ciphertext G T D N E E Q S I Y S G G D W M S G T T E F D I
```

The answer is GTDNEEQSIYSGGDWMSGTTEFDI.

(b) Decrypt the message TEMPRDXEAWESQHGEWPX.

Mapping each letter appropriately gives us

```
ciphertext T E M P R D X E A W E S Q H G E W P X
plaintext  H O W A R E Y O U D O I N G T O D A Y
```

Which, decoded, gives us `How are you doing today?`

1.3 Polybius Square

The **Polybius Square** is another simple substitution cipher which replaces each letter of the plaintext with *two* letters of ciphertext. The idea behind a Polybius square is that it's a table with labeled rows and columns; the alphabet for the messages we're encrypting lives inside the table. For example, if the alphabet we're encrypting includes the capital letters `A` through `Z` and the digits `0` through `9`, then we have 36 letters – perfectly enough to fit in a 6×6 grid. Consider the following arrangement, using the rows and columns `ADFGVX`:

	A	D	F	G	V	X
A	N	A	1	C	3	H
D	8	T	B	2	O	M
F	E	5	W	R	P	D
G	4	F	6	G	7	I
V	9	J	0	K	L	Q
X	S	U	V	X	Y	Z

This table represents our key. To encrypt a message, we convert each letter in the plaintext to a pair of letters indicating the *row* and *column* of that letter in the table above. For example, `K` would be replaced with `VG`. Similarly, `S` would be replaced with `XA`.

(Example.) Suppose Alice wants to encrypt the message

`Storm the gates at 14:37.`

She begins by encoding the message:

`STORMTHEGATESAT1437`

Then, she goes through and replaces each letter by the corresponding pairs as described above:

`XADDDVFGDXDDAXFAGGADDDFAXAADDDAFGAAVGV`

This is the ciphertext. Bob, who knows the table, can undo this process to decrypt the message.

(Exercise.) Use the square given above.

(a) Encrypt the message `Hide tide at 7:01am`.

Encoding the message gives us `HIDETIDEAT701AM`. Then, we can map each individual character in the plaintext to its ciphertext representation:

Plain	**Cipher**
H	AX
I	GX
G	GG
H	AX
T	DD
I	GX
D	FX
E	FA
A	AD
T	DD
7	GV
0	VF
1	AF
A	NN
M	DX

Combining all of this gives us

`AXGXGGAXDDGXFXFAADDDGVVFAFNNDX`

(b) Decrypt the message `XAAAADVGFAFVADDDAXADDDDGFVDX`.

To decrypt, we can map each pair of characters in the ciphertext to its plaintext representation:

Cipher	**Plain**
XA	S
AA	N
AD	A
VG	K
FA	E
FV	P
AD	A
DD	T
AX	H
AD	A
DD	T
DG	2
FV	P
DX	M

Combining and decoding gives us

```
Snake path at 2pm
```