Math 134: Cryptography

Lecture 3: substitution ciphers

Eoin Mackall

January 10, 2025

University of California, Santa Cruz

Ciphers so far:

Shift ciphers

1. Symmetric-key encryption.
2. Key is an integer k with $1 \leq k \leq 25$.
3. Shifts the alphabet by k modulo 26.

Ciphers so far:

Shift ciphers

1. Symmetric-key encryption.
2. Key is an integer k with $1 \leq k \leq 25$.
3. Shifts the alphabet by k modulo 26.

Affine ciphers

1. Symmetric-key encryption.
2. Key is a pair of integers (α, β) where $1 \leq \alpha \leq 25$ is odd, $\alpha \neq 13$, and $0 \leq \beta \leq 25$.
3. Applies the affine transformation $x \mapsto \alpha \cdot x + \beta$ modulo 26 to the alphabet.

Table of contents

Substitution ciphers

Substitution ciphers (set-up)

Alice and Bob want to communicate through some channel. They plan on encrypting their messages for privacy. They aren't convinced by the security of an affine cipher and come up with the following encryption scheme instead:

Substitution ciphers (set-up)

Alice and Bob want to communicate through some channel. They plan on encrypting their messages for privacy. They aren't convinced by the security of an affine cipher and come up with the following encryption scheme instead:

1. Alice and Bob agree beforehand on a permutation of their alphabet (if the alphabet is a set A, then a permutation is a bijection $f : A \to A$.)

Substitution ciphers (set-up)

Alice and Bob want to communicate through some channel. They plan on encrypting their messages for privacy. They aren't convinced by the security of an affine cipher and come up with the following encryption scheme instead:

1. Alice and Bob agree beforehand on a permutation of their alphabet (if the alphabet is a set A, then a permutation is a bijection $f : A \to A$.)
2. To securely send a message $M = m_0 m_1 m_2 m_3 \ldots$, they first apply the permutation letterwise to get the ciphertext $C = f(m_0)f(m_1)f(m_2)\ldots$. Then, they send C.

Substitution ciphers (set-up)

Alice and Bob want to communicate through some channel. They plan on encrypting their messages for privacy. They aren't convinced by the security of an affine cipher and come up with the following encryption scheme instead:

1. Alice and Bob agree beforehand on a permutation of their alphabet (if the alphabet is a set A, then a permutation is a bijection $f : A \to A$.)
2. To securely send a message $M = m_0 m_1 m_2 m_3 \ldots$, they first apply the permutation letterwise to get the ciphertext $C = f(m_0) f(m_1) f(m_2) \ldots$. Then, they send C.
3. If the receiver wants to decrypt the message C, they apply the inverse permutation f^{-1} letterwise to get back M.

Substitution ciphers (set-up)

Alice and Bob want to communicate through some channel. They plan on encrypting their messages for privacy. They aren't convinced by the security of an affine cipher and come up with the following encryption scheme instead:

1. Alice and Bob agree beforehand on a permutation of their alphabet (if the alphabet is a set A, then a permutation is a bijection $f : A \to A$.)
2. To securely send a message $M = m_0 m_1 m_2 m_3 \ldots$, they first apply the permutation letterwise to get the ciphertext $C = f(m_0)f(m_1)f(m_2)\ldots$. Then, they send C.
3. If the receiver wants to decrypt the message C, they apply the inverse permutation f^{-1} letterwise to get back M.

Substitution ciphers (encryption)

Example

Using the permutation below, let's send the message `This is a message`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Substitution ciphers (encryption)

Example

Using the permutation below, let's send the message `This is a message`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Example (cont'd)

The plaintext `This is a message` converts to the ciphertext `Tbds ds c kesscae`.

Substitution ciphers (decryption)

Example

Using the same permutation, let's decrypt the message `Pqyntmaqcnby ds gdlh mr pmmj`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Substitution ciphers (decryption)

Example

Using the same permutation, let's decrypt the message
`Pqyntmaqcnby ds gdlh mr pmmj`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Example (cont'd)

`Pqyntmaqcnby ds gdlh mr pmmj`

.

.

.

Substitution ciphers (decryption)

Example

Using the same permutation, let's decrypt the message `Pqyntmaqcnby ds gdlh mr pmmj`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Example (cont'd)

`Pqyntmaqcnby ds gdlh mr pmmj`

- We find that the α for which $f(\alpha) = p$ is $\alpha = c$.
-
-

Substitution ciphers (decryption)

Example

Using the same permutation, let's decrypt the message `Pqyntmaqcnby ds gdlh mr pmmj`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Example (cont'd)

`Cqyntmaqcnby ds gdlh mr cmmj`

- We find that the α for which $f(\alpha) = p$ is $\alpha = c$.
-
-

Substitution ciphers (decryption)

Example

Using the same permutation, let's decrypt the message `Pqyntmaqcnby ds gdlh mr pmmj`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

…	n	o	p	q	r	s	t	u	v	w	x	y	z
…	l	m	n	o	q	s	t	u	v	w	x	y	z

Example (cont'd)

`Cqyntmaqcnby ds gdlh mr cmmj`

- We find that the α for which $f(\alpha) = p$ is $\alpha = c$.
- Next, $f(r) = q$.
-

Substitution ciphers (decryption)

Example

Using the same permutation, let's decrypt the message `Pqyntmaqcnby ds gdlh mr pmmj`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Example (cont'd)

`Cryntmarcnby ds gdlh mr cmmj`

- We find that the α for which $f(\alpha) = p$ is $\alpha = c$.
- Next, $f(r) = q$.
-

Substitution ciphers (decryption)

Example

Using the same permutation, let's decrypt the message
`Pqyntmaqcnby ds gdlh mr pmmj`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Example (cont'd)

`Cryntmarcnby ds gdlh mr cmmj`

- We find that the α for which $f(\alpha) = p$ is $\alpha = c$.
- Next, $f(r) = q$.
- Next, $f(y) = y$.

Substitution ciphers (decryption)

Example

Using the same permutation, let's decrypt the message `Pqyntmaqcnby ds gdlh mr pmmj`.

α	a	b	c	d	e	f	g	h	i	j	k	l	m
$f(\alpha)$	c	i	p	h	e	r	a	b	d	f	g	j	k

...	n	o	p	q	r	s	t	u	v	w	x	y	z
...	l	m	n	o	q	s	t	u	v	w	x	y	z

Example (cont'd)

If we continue in this way, we should find that the plaintext message was `Cryptography is kind of cool`.

Substitution ciphers

Alice and Bob have described the general formula for a substitution cipher (which include both shift ciphers and affine ciphers).

Substitution ciphers

Alice and Bob have described the general formula for a substitution cipher (which include both shift ciphers and affine ciphers).

1. Since a permutation is used to both encrypt and decrypt, a substitution cipher is a symmetric key algorithm.

Substitution ciphers

Alice and Bob have described the general formula for a substitution cipher (which include both shift ciphers and affine ciphers).

1. Since a permutation is used to both encrypt and decrypt, a substitution cipher is a symmetric key algorithm.
2. They keyspace for a substitution cipher is the set of all permutations of the chosen alphabet. Using the English alphabet, with 26 elements, there are

   $$26! = 403291461126605635584000000 \approx 4.03 \times 10^{26}$$

   many permutations.

Substitution ciphers

Alice and Bob have described the general formula for a substitution cipher (which include both shift ciphers and affine ciphers).

1. Since a permutation is used to both encrypt and decrypt, a substitution cipher is a symmetric key algorithm.
2. They keyspace for a substitution cipher is the set of all permutations of the chosen alphabet. Using the English alphabet, with 26 elements, there are

   $$26! = 403291461126605635584000000 \approx 4.03 \times 10^{26}$$

   many permutations.
3. This is probably too big for a brute force attack to be useful on most computers

Substitution ciphers

Alice and Bob have described the general formula for a substitution cipher (which include both shift ciphers and affine ciphers).

1. Since a permutation is used to both encrypt and decrypt, a substitution cipher is a symmetric key algorithm.
2. They keyspace for a substitution cipher is the set of all permutations of the chosen alphabet. Using the English alphabet, with 26 elements, there are

$$26! = 403291461126605635584000000 \approx 4.03 \times 10^{26}$$

   many permutations.
3. This is probably too big for a brute force attack to be useful on most computers (also: questionably useful for small messages).

Substitution ciphers

Alice and Bob have described the general formula for a substitution cipher (which include both shift ciphers and affine ciphers).

1. Since a permutation is used to both encrypt and decrypt, a substitution cipher is a symmetric key algorithm.
2. They keyspace for a substitution cipher is the set of all permutations of the chosen alphabet. Using the English alphabet, with 26 elements, there are

$$26! = 403291461126605635584000000 \approx 4.03 \times 10^{26}$$

   many permutations.
3. This is probably too big for a brute force attack to be useful on most computers (also: questionably useful for small messages).

Question

So, are substitution ciphers secure?

Substitution ciphers (security)

The answer is not always. With sufficiently many characters of ciphertext known, substitution ciphers become vulnerable to *frequency attacks*.

Substitution ciphers (security)

The answer is not always. With sufficiently many characters of ciphertext known, substitution ciphers become vulnerable to *frequency attacks*.

Frequency attack

A frequency attack uses probability distributions innate to the language used to write plaintext messages in order to gain information about the message.

Substitution ciphers (security)

The answer is not always. With sufficiently many characters of ciphertext known, substitution ciphers become vulnerable to *frequency attacks*.

Frequency attack

A frequency attack uses probability distributions innate to the language used to write plaintext messages in order to gain information about the message.
With enough information, frequency attacks can often yield entire messages.

Substitution (security)

Let's try to decrypt this message.

Ciphertext

Mitsfj ufyhdiqfwhpy oe pswboly vwest ij kwdpskwdouwl dpsify wjt uikhcdsf euosjus hfwudous; ufyhdiqfwhpou wlqifodpke wfs tseoqjst wficjt uikhcdwdoijwl pwftjsee weeckhdoije, kwmojq ecup wlqifodpke pwft di vfswm oj wudcwl hfwudous vy wjy wtbsfewfy. Wpols od oe dpsifsdouwlly hieeovls di vfswm ojdi w asll-tseoqjst eyedsk, od oe ojrsweovls oj wudcwl hfwudous di ti ei. Scup eupskse, or asll tseoqjst, wfs dpsfsrifs dsfkst "uikhcdwdoijwlly esucfs". Tpsifsdouwl wtbwjuse (s.q., okhfibsksjde oj ojdsqsf rwudifoxwdoij wlqifodpke) wjt rwedsf uikhcdojq dsupjiliqy fsgcofs dpses tseoqje di vs uijdojcwlly fssbwlcwdst wjt, or jsuseewfy, wtwhdst. Ijrifkwdoij-dpsifsdouwlly esucfs eupskse dpwd hfibwvly uwjjid vs vfimsj sbsj aodp cjlokodst uikhcdojq hiasf, ecup we dps ijs-doks hwt, wfs kcup kifs torroucld di ces oj hfwudous dpwj dps vsed dpsifsdouwlly vfswmwvls vcd uikhcdwdoijwlly esucfs eupskse.

Substitution (security)

The most common characters occurring in the English language tend to be: “`e`”, “`t`”, “`a`”, “`o`”, “`i`”, “`n`”, “`s`”, “`h`”, “`r`”.

Substitution (security)

The most common characters occurring in the English language tend to be: “`e`”, “`t`”, “`a`”, “`o`”, “`i`”, “`n`”, “`s`”, “`h`”, “`r`”.

The character that appears most often in the above text is `s`. Since notation is also preserved, we see `s.q.` in the line `wtbwjuse (s.q., okhfibsksjde oj ojdsqsf rwudifoxwdoij`.

Substitution (security)

The most common characters occurring in the English language tend to be: “`e`”, “`t`”, “`a`”, “`o`”, “`i`”, “`n`”, “`s`”, “`h`”, “`r`”.

The character that appears most often in the above text is `s`. Since notation is also preserved, we see `s.q.` in the line `wtbwjuse (s.q., okhfibsksjde oj ojdsqsf rwudifoxwdoij`.

Probably then `s` is the ciphertext corresponding to `e` and `q` is the ciphertext corresponding to `g`.

Substitution (security)

The most common characters occurring in the English language tend to be: “`e`”, “`t`”, “`a`”, “`o`”, “`i`”, “`n`”, “`s`”, “`h`”, “`r`”.

The character that appears most often in the above text is `s`. Since notation is also preserved, we see `s.q.` in the line `wtbwjuse (s.q., okhfibsksjde oj ojdsqsf rwudifoxwdoij`.

Probably then `s` is the ciphertext corresponding to `e` and `q` is the ciphertext corresponding to `g`.

The letters `d` and `w` show up the next most often, with roughly the same frequency.

Substitution (security)

The most common adjacent pair of plaintext letters in English tends to be `th`. The most common triple of adjacent plaintext English letters tends to be `the`.

Substitution (security)

The most common adjacent pair of plaintext letters in English tends to be `th`. The most common triple of adjacent plaintext English letters tends to be `the`.

Since `dp` shows up very frequently and `dps` shows up very frequently, we can most likely conclude `d` is the ciphertext corresponding to `t` and `p` is the ciphertext corresponding to `h`.

Substitution (security)

Continuing in this way (i.e. trial and error), we will probably recover the message:

Plaintext

Modern cryptography is heavily based on mathematical theory and computer science practice; cryptographic algorithms are designed around computational hardness assumptions, making such algorithms hard to break in actual practice by any adversary. While it is theoretically possible to break into a well-designed system, it is infeasible in actual practice to do so. Such schemes, if well designed, are therefore termed ”computationally secure”. Theoretical advances (e.g., improvements in integer factorization algorithms) and faster computing technology require these designs to be continually reevaluated and, if necessary, adapted. Information-theoretically secure schemes that provably cannot be broken even with unlimited computing power, such as the one-time pad, are much more difficult to use in practice than the best theoretically breakable but computationally secure schemes.