Chiffre affine et chiffre de Hill

Arithmétique et cryptographie

SUPINFO

Sommaire

1. Chiffre affine.
2. Chiffre de Hill.

1. Chiffre affine.

1. Chiffre affine.

Chiffre de substitution monoalphabétique : rappel

- Un chiffre de **substitution monoalphabétique** est un algorithme de chiffrement où chaque lettre du message d'origine est remplacée par une autre lettre (ou un autre symbole).

- Il est important de noter que dans un tel système de chiffrement, une même lettre est toujours remplacée par une même autre lettre.

1. Chiffre affine.

Chiffre de substitution monoalphabétique : nombre d'algorithmes

- Le nombre de façons de chiffrer un texte par une telle méthode est assez impressionnant :
  - On a en effet 26 choix pour la lettre 'a'.
  - Une fois ce choix fait, il nous reste 25 choix pour la lettre 'b'.
  - On aura de même 24 choix pour la lettre 'c', etc.
- On a donc $26 \times 25 \times 24 \times \ldots \times 1 = 26!$ algorithmes de substitutions monoalphabétiques.
- Ce qui fait environ 4×10^{26} possibilités.

1. Chiffre affine.

Cryptographie symétrique : rappel

- Un système de chiffrement est dit **symétrique** si la clé utilisée lors du chiffrement est aussi celle utilisée lors du déchiffrement.

- Un tel système est aussi qualifié de **système de chiffrement à clé secrète**.

1. Chiffre affine.

Chiffre affine : objectif

- On va généraliser le chiffre de César, en un chiffre dit affine, et qui sera aussi un algorithme de chiffrement symétrique et de substitution monoalphabétique.

- Rappel : tableau de chiffrement du chiffre de César.

clair	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
chiffré	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

1. Chiffre affine.

Convention de représentation des lettres

- Pour pouvoir effectuer des calculs arithmétiques sur les lettres, et donc définir nos algorithmes, nous devons associer préalablement un entier à chaque lettre :

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

- On confondra une lettre et l'entier la représentant.

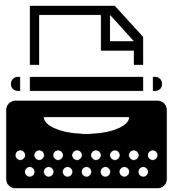
1. Chiffre affine.

Formule du chiffrement affine

- Soit a et b des éléments de $\mathbb{N}$.

- Une lettre x du message d'origine sera chiffrée par la lettre y vérifiant

$$y \equiv ax + b\,[26] \quad \text{et} \quad 0 \leq y < 26$$



1. Chiffre affine.

Formule du chiffrement affine : premier exemple

- Si $a = 1$ et $b = 3$, on retrouve le chiffre de César. On a en effet $y \equiv x + 3\,[26]$, ce qui donne :

clair	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
x	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
y	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	0	1	2
chiffré	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

- Le mot “supinfo“ est alors chiffré en “VXSLQIR“.

1. Chiffre affine.

Formule du chiffrement affine : second exemple

- Si $a = 3$ et $b = 5$, on a alors $y \equiv 3x + 5\,[26]$, ce qui donne :

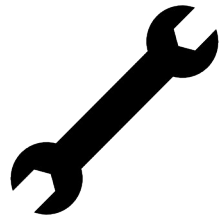
clair	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
y	5	8	11	14	17	20	23	0	3	6	9	12	15	18	21	24	1	4	7	10	13	16	19	22	25	2
chiffré	F	I	L	O	R	U	X	A	D	G	J	M	P	S	V	Y	B	E	H	K	N	Q	T	W	Z	C

- Le mot “supinfo“ est alors chiffré en “HNYDSUV“.

1. Chiffre affine.

Chiffre affine : la clé

- Avec les notations précédentes, la clé du chiffrement affine est le couple (a,b).
- Puisque la transformation précédente est définie modulo 26, on peut se contenter de choisir les entiers a et b entre 0 et 25.
- Cependant toutes ces valeurs ne vont pas convenir.



1. Chiffre affine.

Chiffre affine : exigence légitime sur la clé

- Il paraît naturel d'exiger que deux lettres distinctes soient chiffrées par deux lettres distinctes.

- Si l'on ne choisit pas correctement a et b ce n'est pas nécessairement le cas.

- Par exemple si $a = 2$ et $b = 5$, c'est-à-dire pour la transformation $y \equiv 2x + 5\,[26]$, les valeurs 0 et 13 sont toutes deux transformées en 5 (car $31 \equiv 5\,[26]$).

- Les lettres 'a' et 'n' sont alors toutes deux chiffrées par 'F'.

1. Chiffre affine.

Chiffre affine : condition de validité de la clé

- Un couple (a,b) convient comme clé d'un algorithme de chiffrement affine si et seulement si a et 26 sont premiers entre eux, *i.e.* $\mathrm{PGCD}\,(a,26) = 1$.

- Il n'y a pas de conditions sur b.

- Cela donne comme valeurs possibles pour a : $1,3,5,7,9,11,15,17,19,21,23,25$.

- Il y a donc 12 possibilités pour a et 26 pour b ce qui fait $12 \times 26 = 312$ clés valides.

1. Chiffre affine.

Chiffre affine : démonstration de la condition de validité de la clé

- Supposons que a soit premier avec 26.
- On considère y et y' de la forme $y \equiv ax + b\,[26]$ et $y' \equiv ax' + b\,[26]$ tels que $y \equiv y'\,[26]$.
- On a alors $ax + b \equiv ax' + b\,[26]$, d'où $a\,(x - x') \equiv 0\,[26]$. Ainsi $26 | a\,(x - x')$.
- Or $\mathrm{PGCD}\,(a,26) = 1$, donc d'après le théorème de Gauss $26 |\,(x - x')$. Mais $0 \leq x \leq 25$ et $0 \leq x' \leq 25$ donc $-25 \leq x - x' \leq 25$.
- La seule possibilité est donc que $x - x' = 0$, *i.e.* $x = x'$. Q.E.D.

1. Chiffre affine.

Chiffre affine : démonstration de la condition de validité de la clé

- Réciproquement, supposons que a ne soit pas premier avec 26.
- Soit $d = \mathrm{PGCD}\,(a,26)$ et k,k' les entiers tel que $26 = kd$ et $a = k'd$. On a bien sûr $0 < k < 26$ (car $d > 1$ par hypothèse) et on peut donc considérer la lettre de l'alphabet représentée par k. Notons $ cette lettre.
- Montrons que les lettres 'A' et $, qui sont distinctes puisque représentées respectivement par 0 et k, sont alors chiffrées de la même façon.
- On a $ak + b \equiv k'dk + b \equiv 26k' + b \equiv b\,[26]$. Or il est clair aussi que $a \times 0 + b \equiv b\,[26]$. Q.E.D.

1. Chiffre affine.

Formule du déchiffrement affine

- Soit a et b des éléments de $\mathbb{N}$ tels que $\mathrm{PGCD}\,(a,26) = 1$.
- Soit a' l'inverse multiplicatif de a modulo 26, et soit $b' \equiv -a'b\,[26]$.
- Une lettre y du message chiffré correspondra à la lettre x du message d'origine vérifiant

$$x \equiv a'y + b'\,[26] \quad \text{et} \quad 0 \leq x < 26$$

1. Chiffre affine.

Formule du déchiffrement affine : démonstration

- L'existence de a' est garantie par le critère d'inversibilité (cf. chapitre 3) et le fait que $\mathrm{PGCD}\,(a,26) = 1$.

- Il suffit de démontrer que si l'on applique la formule de déchiffrement à un y de la forme $y \equiv ax + b\,[26]$ on retrouve bien x.

- On a $a'y + b' \equiv a'\,(ax + b) - a'b \equiv a'ax + a'b - a'b \equiv a'ax\,[26]$.

- Or $a'a \equiv 1\,[26]$, on a donc bien $a'y + b' \equiv x\,[26]$.

1. Chiffre affine.

Formule du déchiffrement affine : exemple

- On reprend la clé $a = 3$ et $b = 5$, *i.e.* le chiffrement $y \equiv 3x + 5\,[26]$, et l'on va chercher à déchiffrer "HNYDSUV".

- On détermine la relation de Bézout entre 3 et 26 : $3 \times 9 + 26 \times (-1) = 1$.

- L'inverse multiplicatif modulo 26 de 3 est donc $a' = 9$.

- D'autre part $b' \equiv -a'b \equiv -9 \times 5 \equiv -45 \equiv 7\,[26]$.

1. Chiffre affine.

Formule du déchiffrement affine : exemple

- La formule de déchiffrement est donc $x \equiv 9y + 7\,[26]$. Ce qui donne

chiffré	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
y	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
x	7	16	25	8	17	0	9	18	1	10	19	2	11	20	3	12	21	4	13	22	5	14	23	6	15	24
clair	h	q	z	i	r	a	j	s	b	k	t	c	l	u	d	m	v	e	n	w	f	o	x	g	p	y

- Le mot “HNYDSUV“ se déchiffre alors en “supinfo“.

1. Chiffre affine.

Chiffre affine : décryptement

- On se place ici du point de vue d'un ennemi interceptant un message dont il sait qu'il a été chiffré avec un algorithme de chiffrement affine, mais qui ignore avec quelle clé.

- La possibilité la plus naïve est d'essayer toutes les clés jusqu'à obtenir un texte intelligible.

- Cette méthode dite de "force brute" est tout à fait envisageable car il n'y a que 312 clés possibles.

1. Chiffre affine.

Méthode d’analyse des fréquences

- Cette technique permet plus généralement de décrypter tout message chiffré avec un algorithme de substitution monoalphabétique.

- Elle repose sur une constatation très simple : dans une langue donnée, chaque lettre n’a pas la même fréquence d’utilisation.

- Comme chaque lettre du message d’origine est remplacée par une même autre lettre, on va pouvoir en analysant les fréquences des lettres dans le message chiffré, retrouver la clé de chiffrement.

1. Chiffre affine.

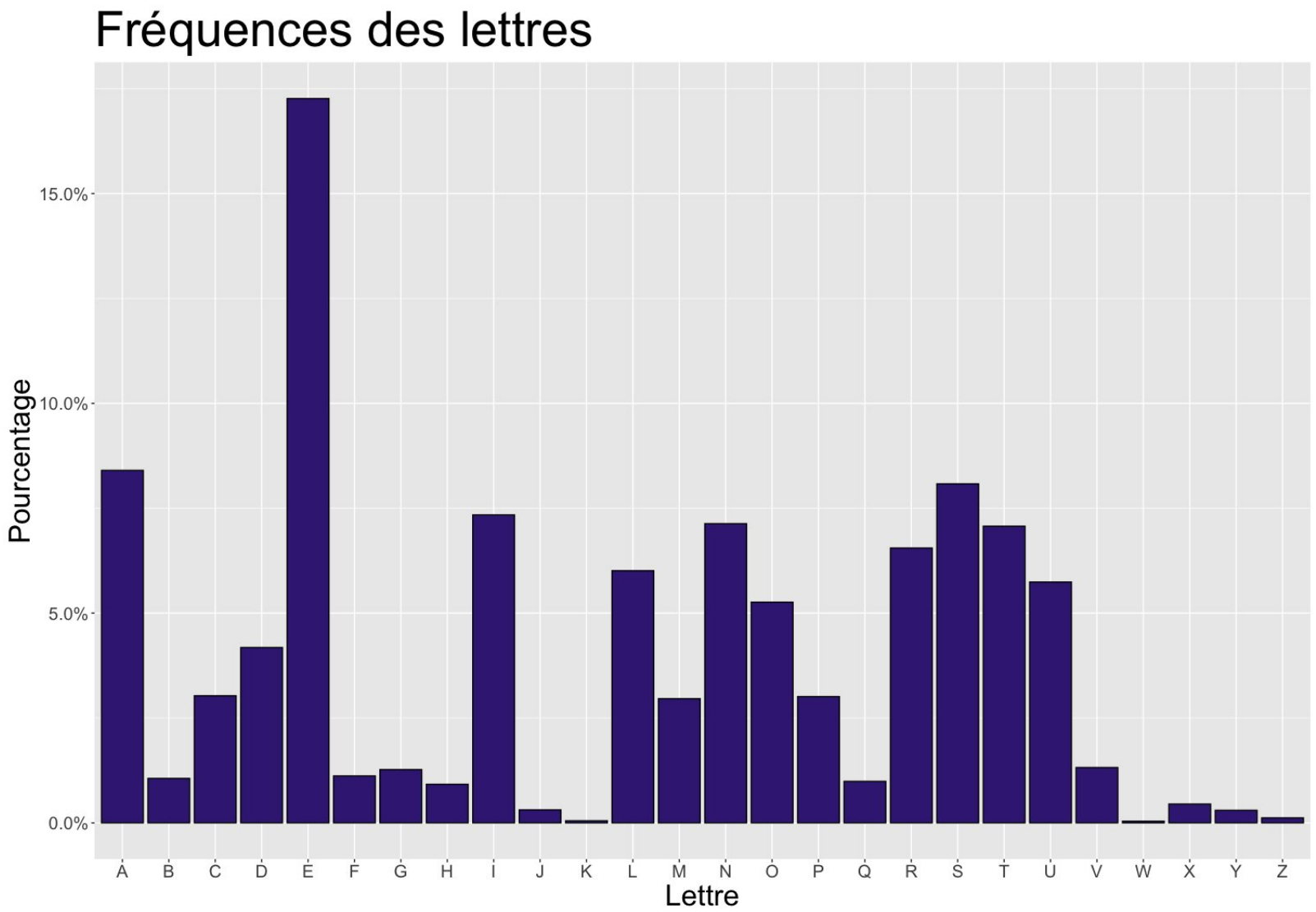
Méthode d'analyse des fréquences

- Il faut connaître les fréquences théoriques d'utilisation des différentes lettres en français.

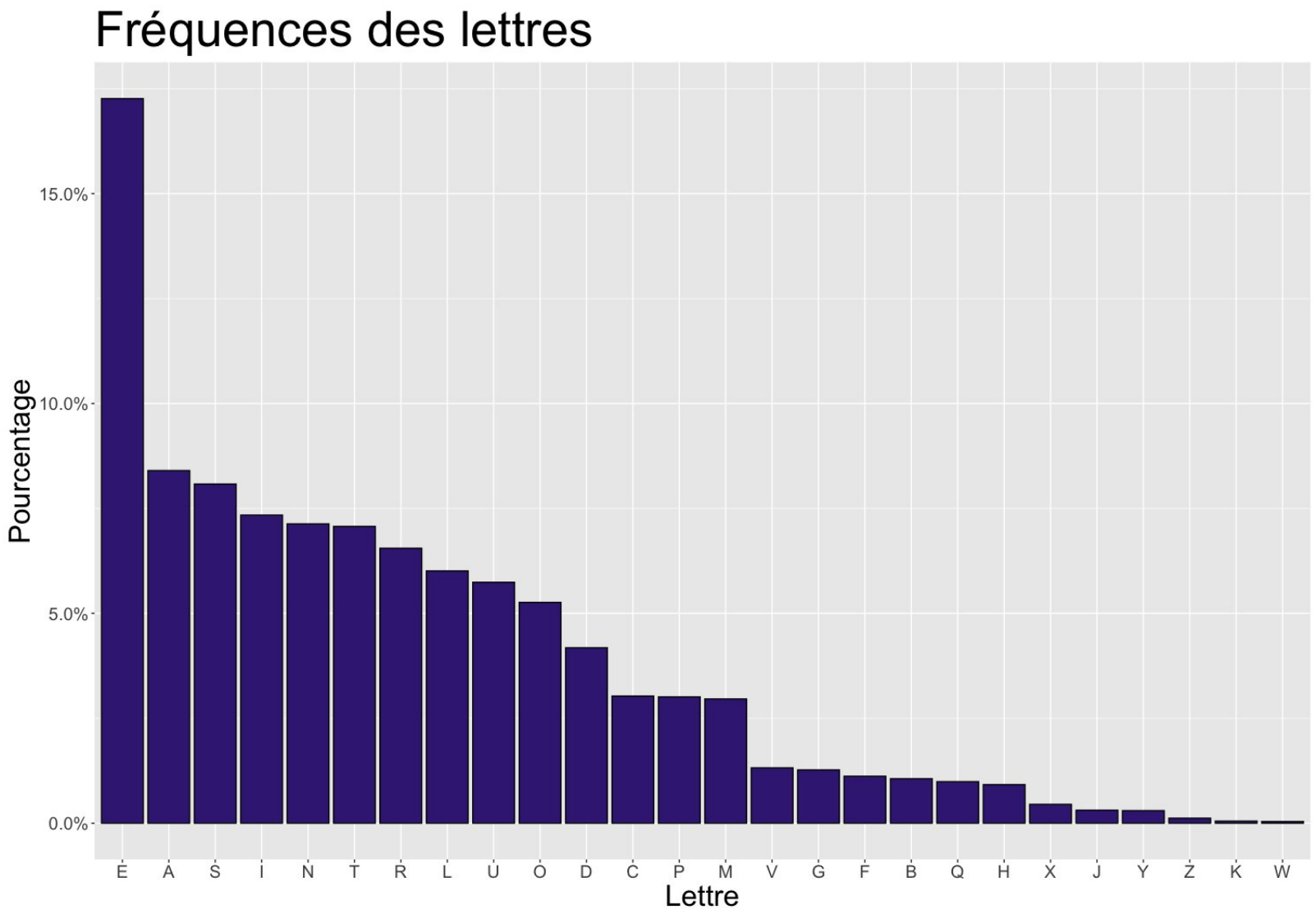
- On peut par exemple effectuer des relevés sur des grands échantillons de lettres provenant de divers textes.

a	b	c	d	e	f	g	h	i	j	k	l	m
8,40	1,06	3,03	4,18	17,26	1,12	1,27	0,92	7,34	0,31	0,05	6,01	2,96
n	o	p	q	r	s	t	u	v	w	x	y	z
7,13	5,26	3,01	0,99	6,55	8,08	7,07	5,74	1,32	0,04	0,45	0,30	0,12

1. Chiffre affine.



1. Chiffre affine.



1. Chiffre affine.

Méthode d’analyse des fréquences

- En pratique on complètera cette étude des fréquences des lettres par celle des doublets. En français les plus fréquents sont dans l'ordre : ‘ss’, ‘ll’, ‘mm’, ‘rr’, ‘tt’, ‘nn’, ‘pp’, ‘ee’, ‘cc’, ‘ff’.

- Si l'on connaît le sujet du message, on pourra également tenter une attaque par mot probable, c’est-à-dire que l'on supposera qu’un mot donné fait partie du message et on essaiera de l’identifier dans le texte. Cela permet alors de décrypter plusieurs lettres d'un seul coup.

1. Chiffre affine.

Méthode d'analyse des fréquences : exemple

- Décrypter le message :

  "GMKAM PPMKK MJCSM BNPMJ VMSGG MQKMK KSNQF DQVCS KNDPO MJTCB KGMQJ KOJCK KCBKN DPOMJ MBNJM GMQJK PCSBK"

- La lettre ayant la plus forte fréquence est le 'M'. On peut penser qu'elle représente le 'e'. Le texte devient :

  "GeKAe PPeKK eJCSe BNPeJ VeSGG eQKeK KSNQF DQVCS KNDPO eJTCB KGeQJ KOJCK KCBKN DPOeJ eBNJe GeQJK PCSBK"

1. Chiffre affine.

Méthode d’analyse des fréquences : exemple

- Ensuite la lettre la plus fréquente est le ‘K’. Elle représente sans doute le ‘a’ ou le ‘s’.

- Sa proximité avec le ‘e’ laisse plutôt penser au ‘s’. Essayons :

  “GesAe PPess eJCSe BNPeJ VeSGG eQses sSNQF DQVCS sNDPO eJTCB sGeQJ sOJCs sCBsN DPOeJ eBNJe GeQJs PCSBs“

1. Chiffre affine.

Méthode d’analyse des fréquences : exemple

- La lettre la plus fréquente est alors le ‘J’. Étant accolée plusieurs fois au ‘e’ c’est sans doute une consonne.

- De même pour le ‘P’. Le ‘a’ sera donc sans doute chiffrée par le ‘C’, lettre la plus fréquente non collée au ‘e’. Cela donne :

“GesAe PPess eJaSe BNPeJ VeSGG eQses sSNQF DQVaS sNDPO eJTaB sGeQJ sOJas saBsN DPOeJ eBNJe GeQJs PaSBs“

1. Chiffre affine.

Extensions possibles

- La technique d'analyse des fréquences a donc mis en évidence toute la faiblesse d’un chiffre de substitution monoalphabétique.

- Pour pallier à ce problème, plusieurs solutions sont envisageables :

    - Utiliser un chiffre de substitution polyalphabétique, comme par exemple le chiffre de Vigenère (cf. chapitre 1).
    - Utiliser un chiffre de substitution polygrammique, comme par exemple le chiffre de Hill (cf. partie suivante).
    - Utiliser un algorithme de cryptographie asymétrique, tel le R.S.A. (cf. chapitre 6).

1. Chiffre affine.

2. Chiffre de Hill.

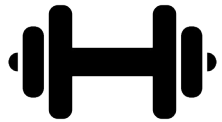
2. Chiffre de Hill.

Chiffre de substitution polygrammique : rappel

- Un chiffre de **substitution polygrammique** est un algorithme de chiffrement où les lettres ne sont pas chiffrées une par une mais par blocs de plusieurs (deux ou trois généralement).

- À noter qu'un même bloc sera toujours remplacé par un même autre bloc.

- On parle également de chiffrement par blocs.



2. Chiffre de Hill.

Chiffre de substitution polygrammique : nombre d'algorithmes

- On va dans la suite ne considérer que des bigrammes, et il y en a $26^2 = 676$ différents.

  - On a en alors 676 choix pour le premier bigramme.
  - Une fois ce choix fait, il nous reste 675 choix pour le deuxième.
  - On aura de même 674 choix pour le troisième, etc.

- On a donc $676 \times 675 \times 674 \times \ldots \times 1 = 676!$ algorithmes de substitutions polygrammiques.

2. Chiffre de Hill.

Chiffre de Hill : objectif

- On va généraliser le chiffre affine en un algorithme de chiffrement lui aussi symétrique mais polygrammique.

- Cet algorithme sera une version à deux dimensions du chiffre affine, nous effectuerons les mêmes types de calculs qu'avec celui-ci.

2. Chiffre de Hill.

Chiffre de Hill : prétraitement du message à chiffrer

- On commence par découper le message en blocs de deux lettres.
- Si le nombre de lettres du message est impair, on rajoute arbitrairement un ‘x’ à la fin.
- On adoptera la même convention de représentation des lettres en nombres que pour le chiffre affine.
- Un bigramme sera donc un couple de deux entiers (x_1,x_2) compris entre 0 et 25.

2. Chiffre de Hill.

Formule du chiffrement de Hill

- Soit a,b,c et d des éléments de $\mathbb{N}$.

- Un bigramme (x_1,x_2) du message d'origine sera chiffré par le bigramme (y_1,y_2) vérifiant

$$\begin{cases} y_1 \equiv ax_1 + bx_2\,[26] & \text{et} \quad 0 \leq y_1 < 25 \\ y_2 \equiv cx_1 + dx_2\,[26] & \text{et} \quad 0 \leq y_2 < 25 \end{cases}$$

2. Chiffre de Hill.

Formule du chiffrement de Hill : exemple

- Si $a = 5, b = 17, c = 4$ et $d = 15$, on a alors

$$\begin{cases} y_1 \equiv 5x_1 + 17x_2\,[26] & \text{et} \quad 0 \leq y_1 < 25 \\ y_2 \equiv 4x_1 + 15x_2\,[26] & \text{et} \quad 0 \leq y_2 < 25 \end{cases}$$

- Chiffrons le mot “supinfo“.
- Le nombre de lettres étant impair, on lui adjoint arbitrairement un ‘x’.
- On découpe ensuite ce mot en blocs de deux lettres "su pi nf ox".

2. Chiffre de Hill.

Formule du chiffrement de Hill : exemple

- L'équivalent numérique du premier bigramme 'su' est $(x_1,x_2) = (18,20)$.
- Il sera donc chiffré en

$$\begin{cases} y_1 \equiv 5 \times 18 + 17 \times 20 \equiv 430 \equiv 14\,[26] \\ y_2 \equiv 4 \times 18 + 15 \times 20 \equiv 372 \equiv\ 8\,[26] \end{cases}$$

- Ce qui donne en lettres 'OI'.
- On procède de même pour les trois autres bigrammes et on obtient "OIDYUXTL".

2. Chiffre de Hill.

Chiffre de Hill : la clé

- Avec les notations précédentes, la clé du chiffrement de Hill est le quadruplet (a,b,c,d).

- Puisque la transformation précédente est définie modulo 26, on peut se contenter de choisir les entiers a,b,c et d entre 0 et 25.

- Cependant toutes ces valeurs ne vont pas convenir, pour les mêmes raisons qu'avec le chiffre affine.

2. Chiffre de Hill.

Chiffre de Hill : condition de validité de la clé

- Il paraît naturel d'exiger que deux bigrammes distincts soient chiffrées par deux bigrammes distincts.

- Une clé (a,b,c,d) d'un chiffre de Hill vérifie cette condition si et seulement si $ad-bc$ et 26 sont premiers entre eux, *i.e.* si $\mathrm{PGCD}\,(ad-bc,26) = 1$.

2. Chiffre de Hill.

Formule du déchiffrement de Hill

- Soit a,b,c et d des éléments de $\mathbb{N}$ tels que $\mathrm{PGCD}\,(ad-bc,26) = 1$.
- Soit i l'inverse multiplicatif modulo 26 de $ad-bc$.
- Un bigramme (y_1,y_2) du message chiffré correspondra au bigramme (x_1,x_2) du message d'origine vérifiant

$$\begin{cases} x_1 \equiv i\,(dy_1 - by_2)\,[26] \quad \text{et} \quad 0 \leq x_1 < 25 \\ x_2 \equiv i\,(-cy_1 + ay_2)\,[26] \quad \text{et} \quad 0 \leq x_2 < 25 \end{cases}$$

2. Chiffre de Hill.

Formule du déchiffrement de Hill : démonstration

- L'existence de i est garantie par le critère d'inversibilité (cf. chapitre 3) et le fait que $\mathrm{PGCD}\,(ad-bc,26) = 1$.

- Vérifions ensuite que si l'on applique la première des deux égalités de la formule de déchiffrement à un couple (y_1,y_2) de la forme

$$\begin{cases} y_1 \equiv ax_1 + bx_2\,[26] \\ y_2 \equiv cx_1 + dx_2\,[26] \end{cases}$$

alors le résultat est x_1.

2. Chiffre de Hill.

Formule du déchiffrement de Hill : démonstration

- On a

$$i\,(dy_1 - by_2) \equiv i\,(d\,(ax_1 + bx_2) - b\,(cx_1 + dx_2)) \equiv i\,(ad - bc)\,x_1\,[26]$$

- Or $i\,(ad - bc) \equiv 1\,[26]$, on a donc bien $i\,(dy_1 - by_2) \equiv x_1\,[26]$
- La formule donnant x_2 se démontre de la même façon. Q.E.D.

2. Chiffre de Hill.

Formule du déchiffrement de Hill : exemple

- On reprend la clé $a = 5, b = 17, c = 4$ et $d = 15$, *i.e.* le chiffrement

$$\begin{cases} y_1 \equiv 5x_1 + 17x_2 \,[26] \\ y_2 \equiv 4x_1 + 15x_2 \,[26] \end{cases}$$

- On va chercher à déchiffrer “OIDYUXTL“.

- On a $ad - bc = 7$ qui est bien premier avec 26. On détermine ensuite la relation de Bézout entre 7 et 26 : $7 \times 15 + 26 \times (-4) = 1$.

- L’inverse multiplicatif modulo 26 de 7 est donc $i = 15$.

2. Chiffre de Hill.

Formule du déchiffrement de Hill : exemple

- La relation de déchiffrement est alors

$$\begin{cases} x_1 \equiv 15 \times (15y_1 - 17y_2)\,[26] \\ x_2 \equiv 15 \times (-4y_1 + 5y_2)\,[26] \end{cases}$$

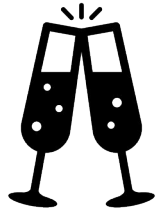
- Considérons le premier bigramme ‘OI’ dont l’équivalent numérique est $(y_1, y_2) = (14,8)$. Il sera déchiffré en

$$\begin{cases} x_1 \equiv 15 \times (15 \times 14 - 17 \times 8) \equiv 1110 \equiv 18\,[26] \\ x_2 \equiv 15 \times (-4 \times 14 + 5 \times 8) \equiv -240 \equiv 20\,[26] \end{cases}$$

2. Chiffre de Hill.

Formule du déchiffrement de Hill : exemple

- Ce couple $(18,20)$ correspond bien au bigramme ‘su’.
- On procède de même pour les trois autres blocs.
- On obtient finalement “supinfox“ et après suppression du ‘x’ final, “supinfo“.



2. Chiffre de Hill.

Chiffre de Hill : décryptement

- Bien qu'il soit plus dur à casser qu'un chiffre affine, le chiffre de Hill est loin de garantir une sécurité totale.
- On peut lui aussi l'attaquer en faisant une analyse de fréquences, mais cette fois sur les bigrammes.
- Les plus fréquents de la langue française étant : 'es', 'en', 'ou', 'de', 'nt', 'te', 'on'.
- On pourra ensuite compléter cette démarche par une attaque avec mot probable.

2. Chiffre de Hill.