Polyalphabetische Kryptosysteme

Beispiel: Vigenère ($n > 1$)

Klar- und Chiffretextraum	$v \in \mathbb{Z}_{26}^n$
Schlüsselraum	$k \in \mathbb{Z}_{26}^n$
Verschlüsselungsfunktion	$E_k(v) = v + k \mod 26$
Entschlüsselungsfunktion	$D_k(v) = v - k \mod 26$

Vigenère Quadrat

```
abcdefghijklmnopqrstuvwxyz
bcdefghijklmnopqrstuvwxyza
            ⋮
efghijklmnopqrstuvwxyzabcd
fghijklmnopqrstuvwxyzabcde
            ⋮
ijklmnopqrstuvwxyzabcdefgh
            ⋮
nopqrstuvwxyzabcdefghijklm
opqrstuvwxyzabcdefghijklmn
            ⋮
zabcdefghijklmnopqrstuvwxy
```

Schlüssel	$k =$ `info` ($n = 4$)
	`infoinfoinfoi`
Klartext	`kryptographie`
Chiffretext	`seddbblficmwm`

Eigenschaften polyalphabetischer Verfahren

Schlüssel	`infoinfoinfoi`
Klartext	`kryptographie`
Chiffretext	`seddbblficmwm`

- gleiche Klartextzeichen werden durch verschiedene Chiffretextzeichen kodiert
- verschiedene Klartextzeichen werden durch das gleiche Chiffretextzeichen kodiert

Folgerung: Direkte Häufigkeitsanalyse schlägt fehl!

Lösung:

- bestimme zuerst die Länge n des Schlüssels
- teile danach den Chiffretext in die n Subtexte auf, die mit dem gleichen Schlüssel kodiert werden
- n Subtexte sind nur durch einfache Caesar-Verschiebungen verschlüsselt
- verwende Häufigkeitsanalyse um Verschiebung der Subtexte zu bestimmen

Zwei Verfahren zur Bestimmung der Schlüssellänge n:

- Kasiski-Test
- Friedman-Test

Chiffretext Vigenère

ncu erlhv ychz sylu kuemcfhvwohpiyu
lr mrtcyq ezl kqvl kmierudyq cex gqv ylvwufp duo evasckthv
ji hqe glak
eoii cvb uuva mip dzthevfwuvcvbvl hqe nlkb ghpi erudyq ulmv
isyu af ylv buhav
cfp yuobv xda ecfpk ghpi uxa
zwk jzh nmzhhz uyu pzyu ie mhqeyp avmvmc zhakeomsn zqv xhz icejvwn wuyu dfawa wlxmyyu
qyl pcvmvb uifp duo mexoqtb ywd brpvh uwjm ucenhzbipuvh
izlykmi... qda yuw lvhq lvl jcvhwmi zucvbhz wohz vcqme mfpvcva xyvxzyob
jndvuzxajvdtc qdz uuv lfwk niohpvl
dtcyv qe xhv ulhkb tx hzykme cvb wohz dcfp lhwmimwm jwkcsfdlv

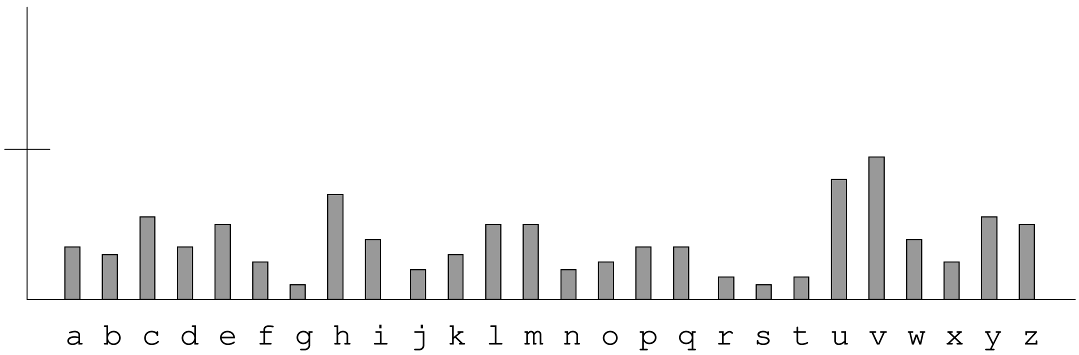


Abbildung 1: Häufigkeitsverteilung bei Chiffretext

Vermutung: Gleichmäßige Häufigkeitsverteilung läßt auf polyalphabetisches Verfahren schliessen.

Kasiski-Test

Betrachte doppelte Buchstabenketten der Länge ≥ 4:

Kette	Abstand	Primfaktorzerlegung
`akeo`	124	$2^2 * 31$
`ghpi`	44	$2^2 * 11$
`hwmi`	100	$2^2 * 5^2$
`wohz`	80	$2^4 * 5$
`fpduo`	192	$2^6 * 3$
`ierudyq`	80	$2^4 * 5$

Bestimme den `ggT` aller Abstände $\Rightarrow$ erhalte Schlüssellänge

Annahme: Schlüsselwort hat die Länge $n = 4$

Gegeben: Chiffretext $c_1 c_2 c_3 \ldots c_m$

Erstelle Häufigkeitsgebirge für die vier Subchiffretexte

- $X_1 = c_1, c_5, c_9, \ldots, c_{4*(t-1)+1}$,
- $X_2 = c_2, c_6, c_{10} \ldots, c_{4*(t-1)+2}$,
- $X_3 = c_3, c_7, c_{11}, \ldots, c_{4*(t-1)+3}$,
- $X_4 = c_4, c_8, c_{12}, \ldots, c_{4*t}$ für $t = \lceil m/4 \rceil$.

<u>Häufigkeitsgebirge 1</u>

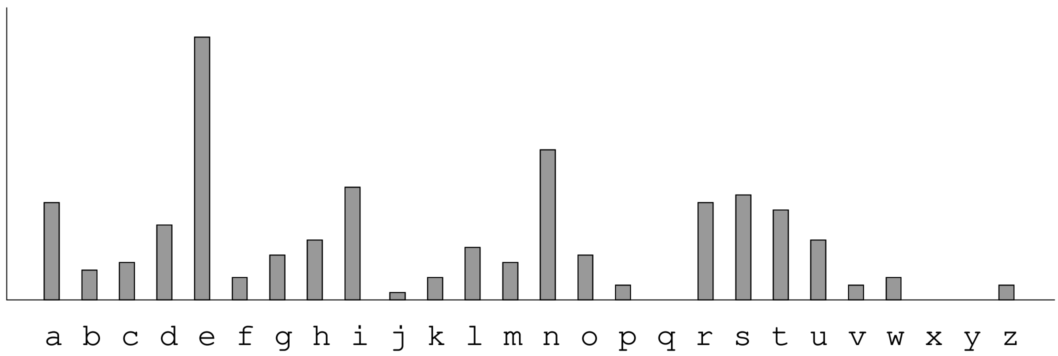


Abbildung 2: Häufigkeitsgebirge im Deutschen

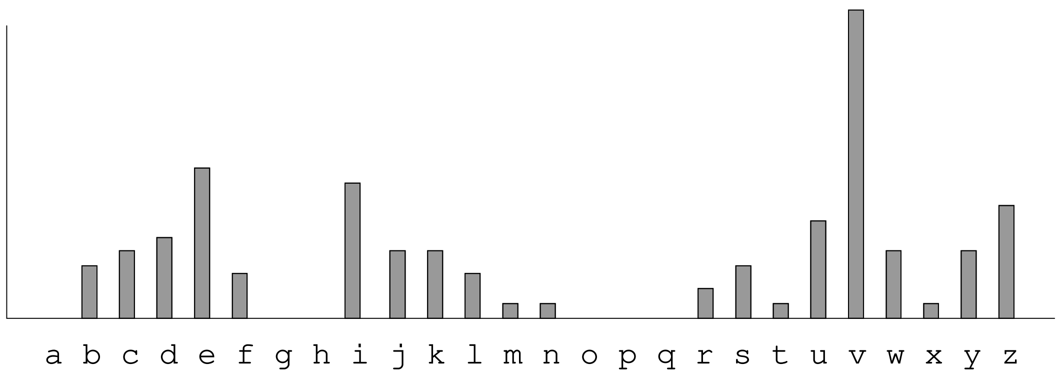


Abbildung 3: Häufigkeitsgebirge für X_1

Vermutung: Caesar Verschiebung mit `r = A`

Häufigkeitsgebirge 2

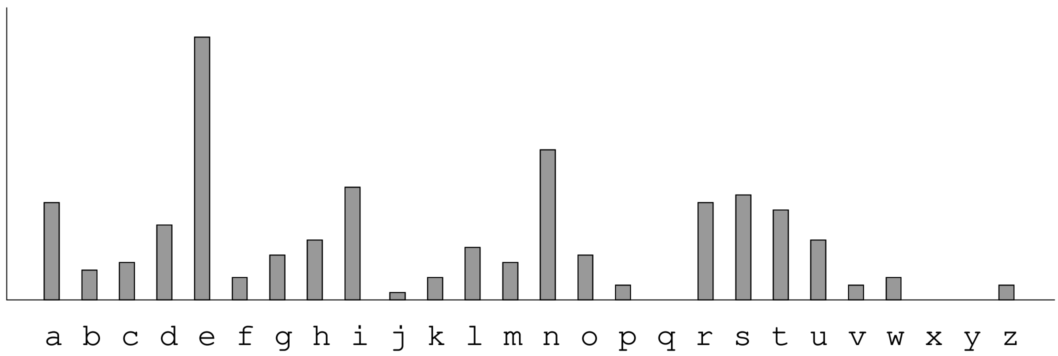


Abbildung 4: Häufigkeitsgebirge im Deutschen

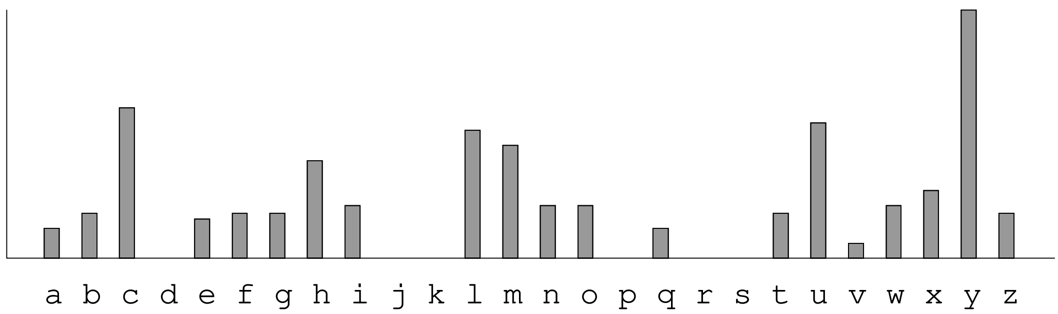


Abbildung 5: Häufigkeitsgebirge für X_2

Vermutung: Caesar Verschiebung mit `u = A`

Häufigkeitsgebirge 3

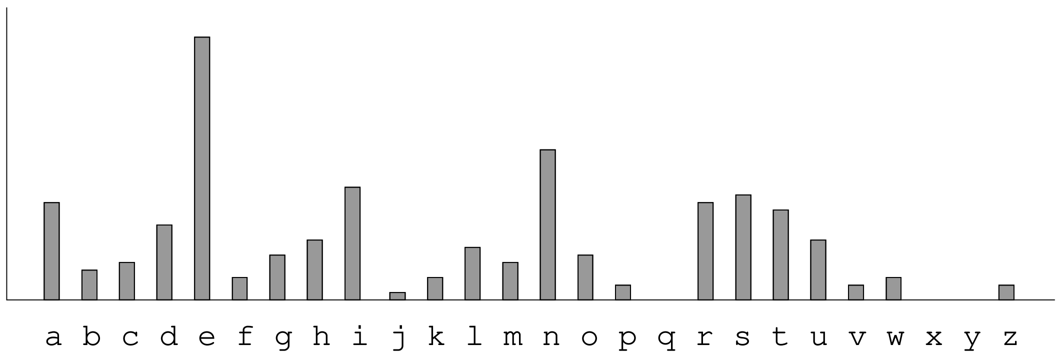


Abbildung 6: Häufigkeitsgebirge im Deutschen

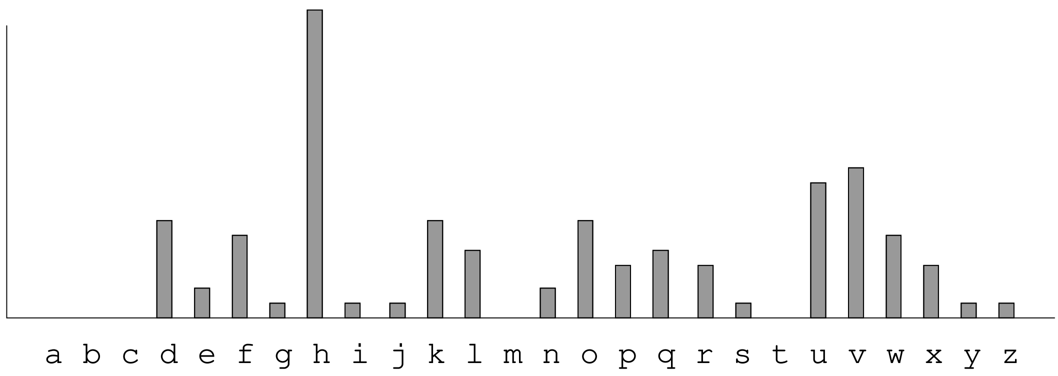


Abbildung 7: Häufigkeitsgebirge für X_3

Vermutung: Caesar Verschiebung mit `d = A`

Häufigkeitsgebirge 4

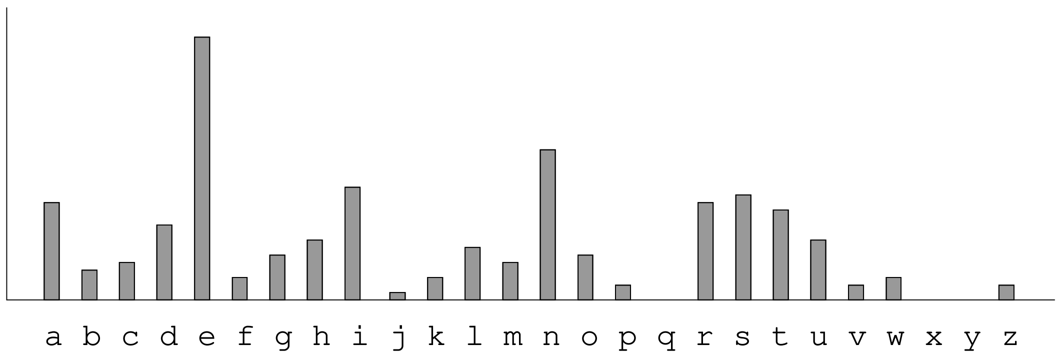


Abbildung 8: Häufigkeitsgebirge im Deutschen

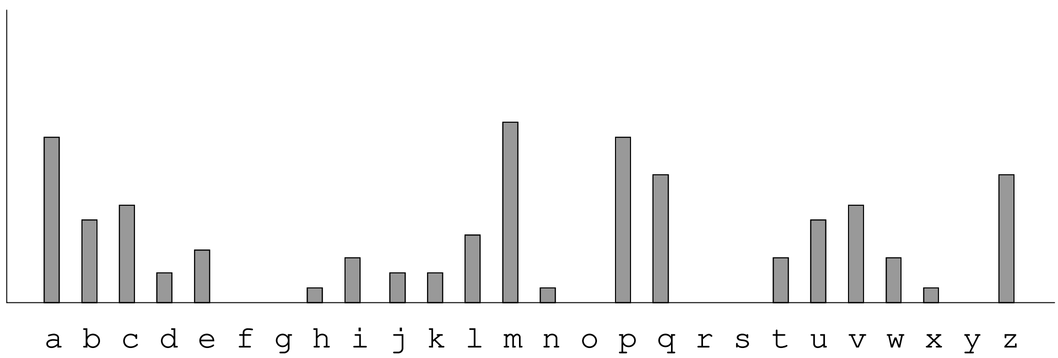


Abbildung 9: Häufigkeitsgebirge für X_4

Vermutung: Caesar Verschiebung mit `i = A`

Auflösung

Benutze Schlüsselwort `rudi` zum Dechiffrieren:

wir waren hier beim tabellenfuehrer
da sollen wir hier herkommen und die einfach mal wegputzen
so ein mist
klar ist dass vom vizeweltmeister ein tick mehr kommen muss
aber so ein kaese
ich halte das nicht mehr aus
ich bin keiner der hier an seinem sessel festklebt wie der ribbeck oder vogts frueher
ihr muesst doch mal endlich vom hohen ross runterkommen
frueher... was hat denn der guenter frueher fuer einen scheiss gespielt
standfussball war das doch frueher
alles in den dreck zu ziehen ist fuer mich unterste schublade

Rudi Völler, 6.9.2003

Nachteile bei Kasiski-Test:

- zufälliges Entstehen von Buchstabenwiederholungen stört bei Längenbestimmung, ignoriere deswegen sich wiederholende Buchstabenketten der Länge ≤ 3
- Schlüssellänge könnte ein Vielfaches (oder ein Teiler) des `ggT` sein

Lösung: Friedman-Test zur Feststellung der *Größenordnung* des Schlüssels.

Vorbereitung Friedman-Test

Frage: Wie wahrscheinlich ist es, dass ein zufällig gewähltes Buchstabenpaar aus einem Text der Länge n aus gleichen Buchstaben besteht?

Antwort: Kommt auf die Häufigkeitsverteilung der Buchstaben an.

Beispiel:

- beliebige Buchstabenfolge der Länge n gegeben
- n_1 Anzahl der `a`'s,
- n_2 Anzahl der `b`'s, ...
- n_{26} Anzahl der `z`'s
- Anzahl von gleichen Buchstabenpaaren:

$$\frac{n_1(n_1-1)}{2} + \cdots + \frac{n_{26}(n_{26}-1)}{2} = \sum_{i=1}^{26} \frac{n_i(n_i-1)}{2}$$

- Wahrscheinlichkeit von gleichem Buchstabenpaar:

$$I = \frac{\sum_{i=1}^{26} \frac{n_i(n_i-1)}{2}}{n(n-1)/2} = \frac{\sum_{i=1}^{26} n_i(n_i-1)}{n(n-1)}$$

Friedmanscher Koinzidenzindex

Häufigkeitsverteilung im Deutschen:

Buchstabe	**Häufigkeit**	**Buchstabe**	**Häufigkeit**
a	6.51%	n	9.78%
b	1.89%	o	2.51%
c	3.06%	p	0.79%
d	5.08%	q	0.02%
e	17.40%	r	7.00%
f	1.66%	s	7.27%
g	3.01%	t	6.15%
h	4.76%	u	4.35%
i	7.55%	v	0.67%
j	0.27%	w	1.89%
k	1.21%	x	0.03%
l	3.44%	y	0.04%
m	2.53%	z	1.13%

Tabelle 1: Charakteristische Buchstabenhäufigkeit im Deutschen

Definiere:

Häufigkeit $\mathsf{a} \rightarrow p_1 = 0.0651$
Häufigkeit $\mathsf{b} \rightarrow p_2 = 0.0189$
$\vdots$
Häufigkeit $\mathsf{z} \rightarrow p_{26} = 0.0113$

Definiere *Koinzidenzindex*:

$$I = \sum_{i=1}^{26} p_i^2$$

Werte für Index

Koinzidenzindex für deutsche Sprache:

$$\sum_{i=1}^{26} p_i^2 = 0.0651^2 + 0.0189^2 + \cdots + 0.0113^2 = \mathbf{0.0762}$$

Bei Gleichverteilung ($p_i = 1/26$ für $1 \leq i \leq 26$):

$$\sum_{i=1}^{26} p_i^2 = \sum_{i=1}^{26} \frac{1}{26^2} = \frac{1}{26} = \mathbf{0.0385}$$

Folgerung: Je gleichmäßiger die Buchstaben im Text verteilt sind, desto kleiner ist der Koinzidenzindex I.

Da bei monoalphabetischen Verfahren die Häufigkeiten der 26 Buchstaben lediglich permutiert werden, ändert sich der Koinzidenzindex beim chiffrieren nicht!

Verfahren zur Spracherkennung: Bei monoalphabetischen Verfahren braucht nur der Koinzidenzindex I des chiffrierten Textes ausgerechnet werden.

Sprache	N	Durchschnittswert für I
Englisch	26	6.61%
Deutsch	26	7.62%
Japanisch	26	8.19%
Russisch	32	5.29%

Koinzidenzindex bei Vigenère

Fakt: Polyalphabetische Verfahren gleichen die Schwankungen in der Häufigkeitsverteilung beim chiffrieren aus.

Frage: Wie kann man aus dem bekannten Index des Klartextes die Länge des Schlüssels beim Vigenère Verfahren herleiten?

Annahme:

- Länge h des Schlüssels bekannt
- Chiffretext $c = c_1 c_2 \dots c_n$ mit $(n = t * h)$ liegt vor

Vorgehensweise: Teile c in h Subtexte auf:

- $X_1 = c_1 c_{h+1} \dots c_{(t-1)*h+1}$
- $X_2 = c_2 c_{h+2} \dots c_{(t-1)*h+2}$
- $\vdots$
- $X_h = c_h c_{2*h} \dots c_{t*h}$

Beobachtung 1: Jedes X_i ist monoalphabetisch chiffriert worden, sollte also einen Index von $I \approx 0.0762$ haben.

Beobachtung 2: Da jede Spalte eine andere Chiffrierung benutzt, stehen Paare von Buchstaben verschiedener Spalten in keiner Beziehung zueinander.

Formelherleitung nach Friedman

Jede Spalte besteht aus n/h Buchstaben. Betrachte wieder Buchstabenpaare:

- Anzahl Buchstabenpaaren in gleichem Subtext
$$n(\frac{n}{h} - 1)/2 = \frac{n(n-h)}{2h}$$
- Anzahl Buchstabenpaaren in verschiedenen Subtexten
$$n(n - \frac{n}{h})/2 = \frac{n^2(h-1)}{2h}$$
- Anzahl A Paare mit gleichen Buchstaben
$$A = \frac{n(n-h)}{2h} * 0.0762 + \frac{n^2(h-1)}{2h} * 0.0385$$
- Wahrscheinlichkeit P, ein solches Paar zu treffen
$$P = \frac{A}{n(n-1)/2} = \frac{0.0377n}{h(n-1)} + \frac{0.0385n - 0.0762}{n-1}$$
- Dieser Wert entspricht in etwa dem Koinzidenzindex des Chiffretextes!
- Berechne Index I für Chiffretext, löse nach h auf
$$h = \frac{0.0377n}{I * (n-1) - 0.0385n + 0.0762}$$

Beispiel für vorherigen Chiffretext

Buchstabe	Anzahl	Buchstabe	Anzahl
a	14	n	8
b	13	o	11
c	23	p	16
d	15	q	16
e	21	r	6
f	12	s	5
g	4	t	7
h	30	u	33
i	18	v	41
j	8	w	18
k	14	x	11
l	22	y	24
m	22	z	21

Tabelle 2: Anzahl Buchstaben im Chiffretext

Länge Chiffretext: $n = 433$, Koinzidenzindex:

$$I = \frac{\sum_{i=1}^{26} n_i(n_i - 1)}{n(n-1)} \approx 0.0469$$

Schlüssellänge nach Friedman:

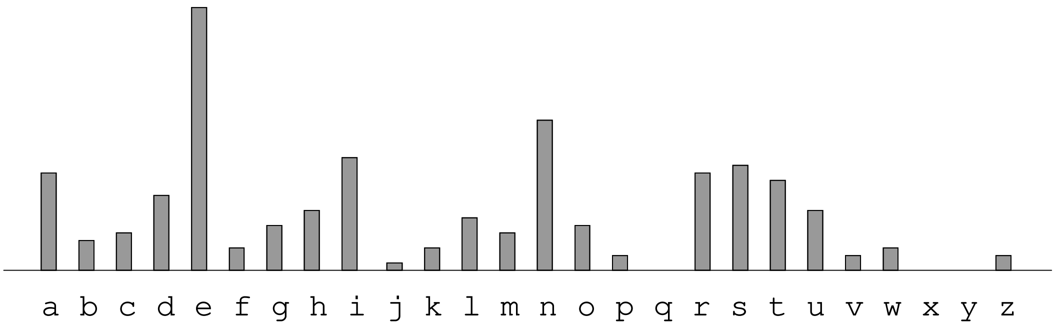
$$h = \frac{0.0377n}{I * (n-1) - 0.0385n + 0.0762} \approx 4.438$$

GUI für Caesar

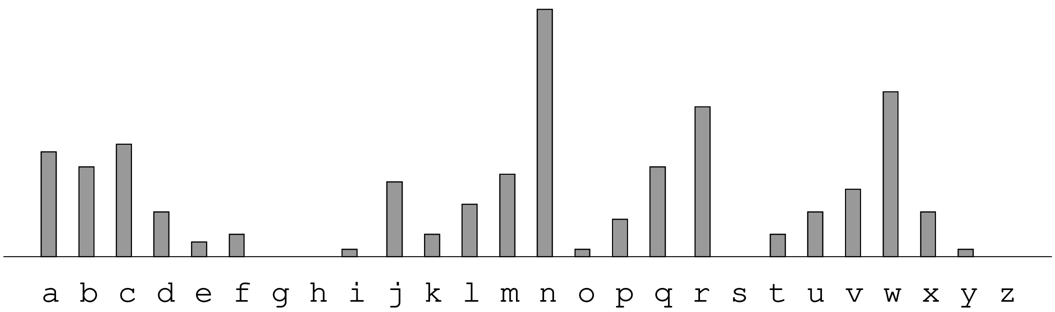
Caesar-Challenge from Peter

afb arbjjpqbk moldoxjjfbobo exybk afb afzhpqbk moldoxjjb

Standard Letter Frequency



Cipher Letter Frequency



◁ $\boxed{a}$ ▷

$\boxed{\text{End Decryption}}$

GUI für Substitution

Substitution-Challenge from Peter

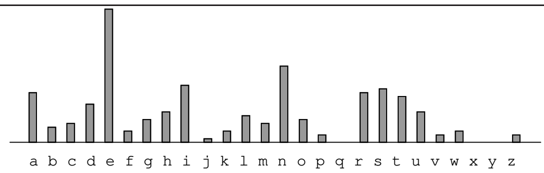
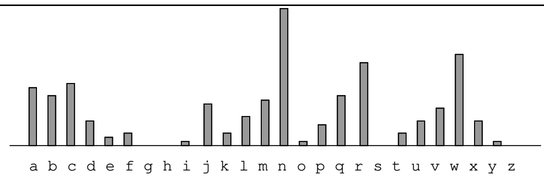
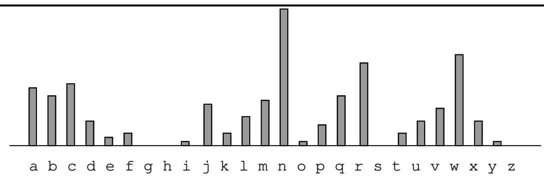
Standard		Ciphertext			Decrypted Message
e	17.40%	x	E	16.39%	
n	9.78%	y	N	8.91%	
i	7.55%	z	I	7.10%	
s	7.27%	m		7.08%	ncu erlhv Nchz sNlu kuemcfhvwohpiNu lr mrtcNq eIl kqvl kmierudNq ceE gqv Nlvwufp duo evasckthv
r	7.00%	o		6.68%	
a	6.51%	q		6.29%	
t	6.15%	e		5.71%	
d	5.08%	j		5.64%	
⋮		⋮			

Bigrams			
en	3.88%	EN	3.29%
er	3.75%	Eo	3.11%
ch	2.75%	fq	2.98%
te	2.26%	uE	2.55%
de	2.00%	mE	2.13%
nd	1.99%	Nm	2.11%
ei	1.88%	EI	2.04%
ie	1.79%	IE	1.77%
⋮		⋮	

Trigrams			
ein	1.22%	EIN	0.99%
ich	1.11%	Ifq	0.88%
nde	0.89%	NmE	0.84%
die	0.87%	mIE	0.81%
und	0.87%	uNm	0.76%
der	0.86%	mEo	0.75%
che	0.75%	fqE	0.73%
end	0.75%	ENm	0.70%
⋮		⋮	

End Decryption

GUI für Vigenère

Vigenère-Challenge from Peter	
`afb arbjjpqbk moldoxjjfbobo exybk afb afzhpqbk moldoxjjb`	
Calculate Friedman-Index	2.45
Calculate Kasiski-Index	2
Enter expected key Length	2
Standard Letter Frequency	
	
Cipher Letter Frequency 1	
	
◁ a ▷	
Cipher Letter Frequency 2	
	
◁ r ▷	
End Decryption	