Homework 1

Fabio Massimo Ercoli *

October 1st 2024

1 Introduction

The *substitution cipher* technique consists in selecting uniformly at random one of the possible permutations of the n character symbols of the given alphabet to determine a key to use both for decrypting and for encrypting.

The key establishes a bijective function having as domain and codomain the character symbols of the alphabets. Bijection is required since we also need to decode, and without it we would lose information applying the function.

Even though we have a very large key space, that is $n!$, frequency analysis can make computationally affordable decoding the ciphertext without any knowledge of the key used to encrypt.

2 Try to use frequency analysis

Let's suppose that the file is an English text file, we know that not all the letter of the alphabet has the same probability to be used.

In the meantime we also received ad hint from the Professor, saying that: 'The alphabet consists of 27 elements, 26 capital letters and a space; each symbol is replaced.', confirming our guess.

This should be confirm this hypothesis, since 26 is the size of the English alphabet.

As a first step let's count the number of occurrences of each letter symbols, ordering them from the most used to less used. To do the job we can use one of the many online tools [1].

Here is the result:

*More on me: https://github.com/fax4ever https://www.linkedin.com/in/fabioercoli/

[1] https://www.dcode.fr/frequency-analysis

K	35×	20.47%
R	13×	7.6%
J	13×	7.6%
Q	12×	7.02%
X	11×	6.43%
Z	11×	6.43%
M	9×	5.26%
A	7×	4.09%
P	6×	3.51%
I	6×	3.51%
F	5×	2.92%
’	5×	2.92%
Y	5×	2.92%
V	5×	2.92%
T	5×	2.92%
O	5×	2.92%
D	4×	2.34%
white space	4×	2.34%
G	4×	2.34%
N	2×	1.17%
H	2×	1.17%
S	1×	0.58%
L	1×	0.58%

2.1 White spaces guess

The most recurrent character is the **K** letter symbol. Looking at the cypher text we notice that this letter symbol splits it in a way similar to the way the white space splits plain text. Moreover, the number of occurrence of K looks suspiciously too high compared with the other. It is a sort of outlier.

At this point we guess that the K letter symbol corresponds to the white space. Moreover, the hint from the Professor mentions the fact that also the white space has been somehow ‘replaced‘.

Since we already have the white space in the original ciphertext, before to substitute all Ks with the white spaces, we need to substitute the white space with something else, since we don’t want to lose information. This is a general ‘empirical‘ technique we’re going to apply anytime we need to substitute a letter with a letter that is already present in the text.

For instance in this case since the letter B is not used, we can replace first all the pre-existing white spaces with B, then finally substitute all the K letters with the white space. Here is the result:

PIFFM MQI’YR J PQM DJBR J PXA ZQXVR NGJMXZ’ XZ THR VTYRRT AQZZJ PR J PXA DJZ VQDR FJM MQI AQT DIF QZ MQ’ SJOR MQI PXA FXVAYJOR BXOBXZ’ MQIY OJZ JGG QLRY THR NGJOR VXZAXZ’

You can notice that the shape of the result looks similar to a common text, in which the words are nicely separate from each other. This should confirm the fact that we made a good choice.

Another thing we can do is to substitute the apostrophe character with some (not already used letter) for instance with C. This is the result

PIFFM MQICYR J PQM DJBR J PXA ZQXVR NGJMXZC XZ THR VTYRRT AQZZJ PR J PXA DJZ VQDR FJM MQI AQT DIF QZ MQC SJOR MQI PXA FXVAYJOR BXOBXZC MQIY OJZ JGG QLRY THR NGJOR VXZAXZC

2.2 Frequency analysis on capitol letters

It's time to redo a frequency-analysis in this new, we can say somewhat normalized text. This time bounding the analysis to the only capital letters from A to Z. Here is the result:

R	13×	9.63%
J	13×	9.63%
Q	12×	8.89%
X	11×	8.15%
Z	11×	8.15%
M	9×	6.67%
A	7×	5.19%
P	6×	4.44%
I	6×	4.44%
F	5×	3.7%
C	5×	3.7%
Y	5×	3.7%
V	5×	3.7%
T	5×	3.7%
O	5×	3.7%
D	4×	2.96%
G	4×	2.96%
B	3×	2.22%
N	2×	1.48%
H	2×	1.48%
S	1×	0.74%
L	1×	0.74%

Now the result looks much more similar to the letters by expected frequency of appearance in English.

2.3 VTYRRT pattern

At this point an idea would be to find a term having some special properties (or pattern) that we can exploit to start translating the capital letters. In the text we identified the term VTYRRT.

The word *VTYRRT* has a nice topology, it has a double R in position 4 and 5, that is our most used letter. Moreover, has other two equal letters in position 2 and 6.

We're going to use another tool [2] found on the web, to perform a query returning all possible worlds matching:

```
PATTERN: VTYRRT
```

The result seems to be a limited (that is good!) set of possible matching:

```
alcool, kousso, markka, nielli, paella, quippu, street, unseen, upkeep.
```

I'm guessing we can exclude some of them, so the list could be even more limited:

```
alcool, paella, street, unseen, upkeep.
```

Using the frequency analysis we can order this list from the most likely to the less ones. Knowing for instance that R in the original text is very used (13 occurrences), while the other symbols (V, T, Y) are on the average of usages.

According to this we can start testing the matching of street, unseen, upkeep. This gives only an order to the tries, now that we've restricted the cases, we're going to try a brute force approach on it.

2.4 Try street as VTYRRT

Substituting the symbols of the term 'VTYRRT' with those of 'STREET', we have the function: $V => S$, $T => T$, $Y => R$, $R => E$.

Here is the result:

PIFFM MQICRE J PQM DJBE J PXA ZQXSE NGJMXZC XZ THE STREET AQZZJ PE J PXA DJZ SQDE FJM MQI AQT DIF QZ MQC KJOE MQI PXA FXSARJOE BXOBXZC MQIR OJZ JGG QLER THE NGJOE SXZAXZC

having as knowing words: "STRE"

In the tool now we can provide the pattern of another word and the sequence discovered on this path. For instance we can query for:

```
PATTERN: FXSARJOE
KNOWN LETTERS: --s-r--e
DOES NOT CONTAIN: t
```

[2] https://design215.com/toolbox/wordpattern.php

The result is now:
disgrace, disprove, misdrove, misgrade
Another nice pattern to search is:

```
PATTERN: SXZAXZC
KNOWN LETTERS: S------
DOES NOT CONTAIN: tre
```

here the results are even less (the lesser the better of course):
silkily, sinking
But those words are not very common words.

Try disgrace as FXSARJOE Substituting the symbols of the term 'FXSARJOE' with those of 'DISGRACE', we can add the following matching:
$F => D$, $X => I$, $A => G$, $J => A$,$O => C$
The result would be:

P3DDM MQ36RE A PQM 2ABE A PIG ZQISE N4AMIZ6 IZ THE STREET GQZZA PE A PIG 2AZ SQ2E DAM MQ3 GQT 23D QZ MQ6 1ACE MQ3 PIG DISGRACE BICBIZ6 MQ3R CAZ A44 QLER THE N4ACE SIZGIZ6

having as knowing words: "STREDIGAC"
At this point it looks to be difficult to find a mapping for the term SIZGIZ6 with the current mapping. Thus we backtrack to try something else.

2.5 Try unseen as VTYRRT

Substituting the symbols of the term 'VTYRRT' with those of 'UNSEEN', we have the function: $V => U$, $T => N$, $Y => S$, $R => E$.
Here is the result:

PIFFM MQICSE J PQM DJBE J PXA ZQXUE 1GJMXZC XZ NHE UNSEEN AQZZJ PE J PXA DJZ UQDE FJM MQI AQN DIF QZ MQC 2JOE MQI PXA FXUASJOE BXOBXZC MQIS OJZ JGG QLES NHE 1GJOE UXZAXZC

But in this case it seems to be not possible found a word matching UXZAXZC.

3 Changing a bit our initial hypothesis

We assumed (guessing) that the apostrophe symbols should have been treated as any other characters. Now let's try to keep those values as they are, without substituting it.
Starting back from the text with apostrophes:

PIFFM MQI'YR J PQM DJBR J PXA ZQXVR NGJMXZ' XZ THR VTYRRT AQZZJ PR J PXA DJZ VQDR FJM MQI AQT DIF QZ MQ' SJOR MQI PXA FXVAYJOR BXOBXZ' MQIY OJZ JGG QLRY THR NGJOR VXZAXZ'

We're going to apply the same reasoning we lead to try street as VTYRRT and disgrace as FXSARJOE.

This is the corresponding result with apostrophos:

P3DDM MQ3'RE A PQM 2ABE A PIG ZQISE N4AMIZ' IZ THE STREET GQZZA PE A PIG 2AZ SQ2E DAM MQ3 GQT 23D QZ MQ' 1ACE MQ3 PIG DISGRACE BICBIZ' MQ3R CAZ A44 QLER THE N4ACE SIZGIZ'

having as knowing words: "STREDIGACH". I added also the H, since the THE can be seen as THE.

Now backing to the SIZGIZ' term, this could seen as SINGIN'. Sometimes native speakers tend to use the term in' in place of ing.

Continuing on this idea, applying the same techniques we eventually get the following text:

BUDDY YOU'RE A BOY MAKE A BIG NOISE PLAYIN' IN THE STREET GONNA BE A BIG MAN SOME DAY YOU GOT MUD ON YO' FACE YOU BIG DISGRACE KICKIN' YOUR CAN ALL OVER THE PLACE SINGIN'

That we can be recognized as a plausible plain text[3].

[3]https://en.wikipedia.org/wiki/We_Will_Rock_You