MATH 314 Spring 2018 - Class Notes

2/6/2018

Scribe: James Garrison

Summary:
Problems with Monoalphabetic Ciphers; introducing the Vigenere Cipher

Notes:
Previously, we've covered

- **Caesar Cipher**- 26 possible keys
- **Affine Cipher**- 312 possible keys

These ciphers share two problems:

1. Their keyspaces are sufficiently small that they're easily defeated by brute force attacks.
2. They are *monoalphabetic ciphers*, or ciphers where each letter of plain text (PT) maps to exactly one letter of cipher text (CT)

All monoalphabetic ciphers share a vulnerability to *frequency analysis*.

Consider the **Substitution Cipher**, a monoalphabetic cipher where the key is any permutation of the alphabet; i.e. each letter of PT is encoded to exactly one arbitrarily chosen letter of CT.

This gives us a keyspace 26 factorial, or $26! \approx 4.03 \times 10^6 \approx 2^{88}$. This is a large enough number of possible keys to be resistant to brute force attacks. However, there are still several options for attacking a Substitution Cipher.

- Known Plain Text- Recover elements of the key by reading off characters in the message
- Chosen Plain Text- Recover the key by encrypting the entirety of the alphabet.
- Cipher Text Only- Given a sufficiently large CT, frequency analysis and informed guesses will quickly decrypt the message and recover the key.

  Frequency Analysis attacks were made more difficult with the inclusion of null characters, but they were still very successful.

  Note: refer to frequency analysis examples against the Caesar Cipher

Vigenere Cipher:

Instead of using integer values, the Vigenere Cipher uses a key word or phrase.

To Encrypt:

1. As with the Caesar and Affine Ciphers, write out the PT message as a row of numbers, called a vector
2. Below that, repeatedly write the key word as a vector until there are corresponding characters of the key for each character of PT.
3. The sum of the two vectors $\pmod{26}$ produces the CT.

E.g. consider the PT `"here is how it works"` and the key "vector"

$$\begin{aligned} PT &= [07, 04, 17, 04, 08, 18, 07, 14, 22, 08, 19, 22, 14, 17, 10, 18] \\ &+ \\ KEY &= [21, 04, 02, 19, 14, 17, 21, 04, 02, 19, 14, 17, 21, 04, 02, 19] \\ &\equiv \\ CT &= [02, 08, 19, 23, 22, 09, 02, 18, 24, 01, 07, 13, 09, 21, 12, 11] \\ CT &= \texttt{CITXWJCSYBHNJVML} \end{aligned}$$

Note that the PT `"i"` is encrypted to both `"W"` and `"B"`. This means that the Vigenere Cipher is not monoalphabetic, and is not vulnerable to typical frequency analysis.

To Decrypt: Reverse the process, such that $PT \equiv CT - KEY \pmod{26}$

$$\begin{aligned} CT &= [02, 08, 19, 23, 22, 09, 02, 18, 24, 01, 07, 13, 09, 21, 12, 11] \\ &- \\ KEY &= [21, 04, 02, 19, 14, 17, 21, 04, 02, 19, 14, 17, 21, 04, 02, 19] \\ &\equiv \\ PT &= [07, 04, 17, 04, 08, 18, 07, 14, 22, 08, 19, 22, 14, 17, 10, 18] \\ PT &= \texttt{hereishowitworks} \end{aligned}$$

Attacking the Vigenere Cipher:

- Known Plain Text- Similarly to a known PT attack against the Caesar Cipher, simply subtract the PT from the CT (mod 26) one letter at a time to recover the key.
- Chosen Plain Text- Encrypting a string of PT "a"s will yield the key.
- Cipher Text Only- The hardest part of a CT only attack is determining the key length. The difficulty of this problem led the Vigenere Cipher to be considered unbreakable for over 200 years. The below technique for determining the key length was first developed by Charles Babbage.

  To find key length:

  1. Write out the CT, with all spaces and punctuation removed
  2. Repeat the CT on the line below, but shifted one character to the left (start with second character).
  3. Repeat this process, incrementing the shift by one each line.
  4. For each shifted line of CT, count the number of times a letter in the shifted CT matches the letter in the corresponding position of the unshifted CT (the first line). These are called coincidences.
  5. Compare the number of coincidences per shift length. There will be a spike in coincidences when the shift length is a multiple of the key length. This is a consequence of the same properties that make frequency analysis effective.

  From here, you could perform standard frequency analysis on the groups of CT character encoded with the same character of the key.

  Note: This technique requires a relatively large CT sample.