MATH 314 Spring 2018 - Class Notes

09/12/2018

Scribe: Brian Gonch

Summary: In this class we finished examining attacks on hill ciphers, discussing one time pads, and began learning about the Euclidean Algorithm.

Hill Cipher: Known Plaintext

- The hill cipher's encryption algorithm $E(\vec{x}) \equiv \vec{x}k \equiv \vec{w}$ where $\vec{x}$ and $\vec{w}$ are vectors with m elements, and where k is an mxm invertible matrix
- If we combine multiple blocks of encryption we can create a m x m matrix out of the vectors $\vec{x}$ and $\vec{w}$. this lets us figure out the key by multiplying both side by the inverse of the input.
- **Example:** if we know the first letters in a message `linear` encrypt to `LTPVPI` we have the matrix equation $\begin{bmatrix} l & i \\ n & e \end{bmatrix} K \equiv \begin{bmatrix} L & T \\ P & V \end{bmatrix}$
  if we multiply both sides of equation by the inverse of our input on the left side:
  $\begin{bmatrix} l & i \\ n & e \end{bmatrix}^{-1} \begin{bmatrix} l & i \\ n & e \end{bmatrix} K \equiv \begin{bmatrix} l & i \\ n & e \end{bmatrix}^{-1} \begin{bmatrix} L & T \\ P & V \end{bmatrix}$
  we get:
  $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} K \equiv \begin{bmatrix} l & i \\ n & e \end{bmatrix}^{-1} \begin{bmatrix} L & T \\ P & V \end{bmatrix} (mod 26)$
  Note: We must multiply by the inverse on the left side of the matrix ($a^{-1}a \neq aa^{-1}$)

  by converting the ciphertext and plaintext to numbers we get the equation
  $\begin{bmatrix} 11 & 8 \\ 13 & 4 \end{bmatrix} K \equiv, \begin{bmatrix} 11 & 19 \\ 15 & 21 \end{bmatrix} (mod 26)$
  We need to find the inverse of $\begin{bmatrix} 11 & 8 \\ 13 & 4 \end{bmatrix}$, but the matrix might not be invertible
  you can invert a 2x2 matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$using the equation $\frac{1}{ad-bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$, but we cannot divide in modular arithmetic we need to find the inverse of $ad - bc$ instead.

  $11(4) * 8(13) \equiv 18(mod 26)$ doesn't have and inverse mod 26. However we can keep trying matrices until we find one we can invert.

  lets try changing the bottom letters to the `ar` in linear which encrypts to `PI` we get the E(x)$\equiv \begin{bmatrix} 11 & 8 \\ 0 & 17 \end{bmatrix} K \equiv \begin{bmatrix} 11 & 19 \\ 15 & 8 \end{bmatrix} (mod 26)$

$11(17) * 8(0) \equiv 5(mod 26)$
the inverse of $\begin{bmatrix} 11 & 8 \\ 0 & 17 \end{bmatrix}$ is $5^{-1} \begin{bmatrix} 17 & -8 \\ 0 & 11 \end{bmatrix} \equiv 21 \begin{bmatrix} 17 & 18 \\ 0 & 11 \end{bmatrix} \equiv \begin{bmatrix} 19 & 14 \\ 0 & 23 \end{bmatrix} (mod 26)$
Finally we get: $\begin{bmatrix} 19 & 14 \\ 0 & 23 \end{bmatrix} \begin{bmatrix} 11 & 8 \\ 0 & 17 \end{bmatrix} K \equiv \begin{bmatrix} 19 & 14 \\ 0 & 23 \end{bmatrix} \begin{bmatrix} 11 & 19 \\ 15 & 8 \end{bmatrix} (mod 26)$
$K \equiv \begin{bmatrix} 19 * 11 + 14 * 15 & 19 * 11 + 14 * 8 \\ 11 * 0 + 23 * 15 & 0 * 19 + 23 * 15 \end{bmatrix} \equiv \begin{bmatrix} 3 & 5 \\ 7 & 2 \end{bmatrix} (mod 26)$

Hill Cipher: Ciphertext Only

- We can preform frequency analysis to find on groups of letters.
- For instance the most common letter pairing in english is `th`. we can use this digraph to guess what the most likely key is.
- This technique falls apart when the block's length enters the early teens.

One-Time Pads

- A one-time pad uses a key with equal key length to the message to provide perfect secrecy. It functions like a vigenere cipher
- Perfect secrecy means that no information about the plaintext can be gained from ciphertext. For instance, if you received a three letter message it `AAA` the plaintext could be `cat`, but it could also be be `dog`. It is impossible to know what the plaintext contains without having the cipher. Eve is unable to even determine the actual length of the message because the plaintext is probably padded to match the cipher's length.
- The problem with one-time pads is they can only be used once. Alice and Bob need to have copies. Since the cipher needs to be as long as the plaintext it is very difficult to securely transport the cipher.

The Euclidean Algorithm: Efficiently finding the gcd of two numbers

- We use the Euclidean Algorithm because it is an efficient way find the gcd of two numbers
- The Algorithm works because $gcd(a, b) = gcd(b, r)$ where r is the remainder $a/b = q+r$. Another way of thinking about the relationship between a,b, and r this is $a = q(b) + r$.
- By repeatedly using this algorithm we can take a large difficult problem, and turn it into a smaller one.
- **Example:**find the gcd of 100 and 15
  $100 = 6(15) + 10$
  $15 = 1(10) + 5$
  $10 = 2(5) + 0$ From this we know the gcd(100,15)=5