Hill Cipher Matrix Cryptography

Jared-Michael E. Torres

Eastern Oregon University

Throughout this project, we will be exploring the idea of matrices and their inverses to encode and decode messages. This will demonstrate how the invertible matrix theorem works, and that when a matrix ‘A’ is invertible, then every in has a unique solution and will be consistent. Also, if ‘A’ is invertible then , where ‘I’ is the identity matrix will always be true. Using these ideas and the application of modular arithmetic, we will be able to encode, and decode text messages using what is commonly known as a Hill Cipher. Included at the end of this project is a program written in the Java computer programming language that gives an example of how messages on a computer can be encrypted/decrypted in Appendix A, which has many applications from securing messages between two or more people, or even securing passwords in a database by storing them as their encrypted form.

Firstly, modular arithmetic may sound intimidating but is rather simplistic. For this project, we will just require the elementary understanding of modular arithmetic. Modular arithmetic is essentially keeping track of the remainders of numbers during division. For example, is the same as the remainder of , where 27 goes into two, zero times, leaving a remainder of 2, where n = 2. Another example would be , because 27 goes into 29, 1 time and has a remainder of 2. This can be applied for any real number to find the modular remainder. While this is a basic understanding of modular arithmetic, it is all that is required for this cryptography project.

Second, we can apply the following formulas to matrices to help understand how the vectors transform under matrix vector multiplication. If , which means, if we take the resulting vector ‘b’, and multiply it by the inverse of the matrix ‘A’, we will end up with the original vector ‘x’. This process is our encryption and decryption of messages, where is our vector of text characters assigned an integer value, and is the resulting values to be converted back to a corresponding text character value. To perform this operation our matrix ‘A’ must be a square matrix or ‘n’ x ‘n’, where n is any positive integer, and ‘A’ must also be invertible. Following this our vector must be a column vector of ‘n’ x 1.

Following this idea, we will create a substitution table for all the letters of the alphabet and include a ‘\_’ to represent a space for a better visual understanding.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| \_ | 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 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Now let’s use a matrix , and where and . To find the inverse of we take the original matrix ‘A’ and mod each entry by mod27. Then we perform the following operations on ‘A’ to find it’s inverse.

For this to work, we require the inverse of ‘A’ to be in modular form. The values of a-d must all be some value ‘n’ where .

Following this we find

Solving where ‘n’ is some value between 0-27 that makes . Therefore and . Now solving for the modular inverse of ‘A’ is easy. Following the above formula, we have:

.

The modular of a matrix is simply finding the modular of every value within the matrix.

E.g.) (The easiest way to find the modular of a negative number is to add the modular number to the negative number until you reach the first positive value. Ex.

Now that we have the two required matrices and a substitution table we can do an example of encoding and decoding a message.

We will encode the plain text “A\_dad” as an example. Following our substitution table ‘A’ = 1, ‘\_’ = 0, ‘D’ = 4. Our plain text then translates into 1 0 4 1 4. Since we need an even number of values to construct our we will add an additional space or ‘\_’ to the end of our plain text, giving us 1 0 4 1 4 0. You will see why in just a moment while we encode the message. Our . Now we will multiply each individual by our encode matrix.

This then gives us the encoded message of “DBSJPH”, and we have 6 letters for each message. Notice how repeated characters such as the ‘A’, ‘D’, and ‘\_’ have all been given completely different values when encoded. This provides an added benefit for the encoded message making it nearly impossible to crack without the original encode matrix or decode matrix.

Now we can decode our encrypted text of “DBSJPH” to demonstrate how the matrix transformation can be undone using the inverse/decode matrix, where our encrypted text using our substitution table becomes 4 2 19 10 16 8. And our set of .

Since the values are greater than 26 we are required to use the

modular of the values to determine their letter value for the substitution.

This then provides us with the following decoded values of 1 0 4 1 4 0. Using the substitution table, we get the following message “A\_DAD\_”. The encode and decode matrices were successful and now we can encrypt messages using the Hill Cipher technique.

Demonstrating the basic idea of linear transformations using the Hill Cipher technique, we can see one of the many benefits of linear transformations. The Hill Cipher is not limited to using only a 2x2 matrix but any square matrix that is invertible while using modular arithmetic. These techniques can be applied to personal private messages, or even science fields such as computer science where encrypting data is vital in an online world filled with potential threats of a hacker who is capable/willing to steal information.

References

Greg Schulberg – Blue Mountain Community College, Computer Science/Mathematics department.

<http://practicalcryptography.com/ciphers/hill-cipher/>

https://www.khanacademy.org/computing/computer-science/cryptography/modarithmetic/a/modular-inverses

Appendix A