Senior Math Circles – Cryptography and Number Theory Week 1

Dale Brydon

Feb. 2, 2014

1 One-Time Pads

Cryptography deals with the problem of encoding a message in such a way that only the intended recipient can decode it. This ensures secrecy even in the case where an adversary intercepts the encoded message. There are many methods that have been used to attempt to solve this problem throughout history, but we will start with one that saw much use throughout World War II.

A *one-time pad* is an encryption method where each letter in the message is encoded by adding a random letter to it. We call the sequence of random letters the *key*. An essential property of the security of a one-time pad is that each key is only used one time *ever*.

Notice that the scheme requires us to be able to add two letters together. We can accomplish this by assigning a number to each letter and then adding the numbers. There are two notes we should think about when doing this. First, one natural choice for assigning numbers we be to set $A = 1,\ B = 2$, and so on up to $Z = 26$. This would indeed work out fine, but for various reasons, we will instead set $A = 0,\ B = 1$, and so on up to $Z = 25$. We will see why we prefer starting at 0 in a little. The second thing to consider is you might get a sum greater than 25, for example by adding $N = 13$ and $R = 17$. In this case what we will do is wrap back around by saying that $A = 26,\ B = 27,\ \ldots,\ Z = 51$. So then $N + R = 13 + 17 = 30 = E$. Another, equivalent, way of viewing this is to subtract 26 from any number greater than 25 and use our original set of values. When we do this we get that $N + R = 30 = 4 = E$.

As an example, suppose Alice wanted to send the message "WE STRIKE AT DAWN" to Bob. She would need to have a randomly generated a key that is the same length as her message (note, we ignore spaces). For example, a random key might be "EKCPZSLUENIXFR". To get the *ciphertext* Alice must add each letter in the key to the corresponding letter in the plaintext. This is done as shown in the table below.

Message	W	E	S	T	R	I	K	E	A	T	D	A	W	N
	22	4	18	19	17	8	10	4	0	19	3	0	22	13
Key	E	K	C	P	Z	S	L	U	E	N	I	X	F	R
	4	10	2	15	25	18	11	20	4	13	8	23	5	17
Ciphertext	26	14	20	34	42	26	21	24	4	32	11	23	27	30
	0	14	20	8	16	0	21	24	4	6	11	23	1	4
	A	O	U	I	Q	A	V	Y	E	G	L	X	B	E

Notice that the ciphertext appears as random as the key. This is basically where the security of the scheme comes from.

Now that we know how to encrypt a message using a one-time pad, we need to know how to decrypt it as well. To decrypt Alice's message, Bob also needs a copy of the secret key. Since the assumption is that Alice and Bob are communicating over an insecure channel, they need to have shared this key in advance. In any case, once Bob has the key, he just needs to subtract each letter in the key from the corresponding letter in the cipertext. In this case, we do the same letter-to-number conversion as before, but now whenever a number is less than 0 we add 26. So for example $A - E = 0 - 4 = -4 = 22 = W$.

Since subtraction "undoes" addition, it's intuitive that this method will undo the encryption and we'll get back the original message. In fact, you'll notice that in the encryption we added the letter E from the key to the letter W from the message and got the ciphertext letter A and we also just saw that subtracting E from A gets back W.

If used correctly, the one-time pad achieves the highest possible level of security for an encryption scheme, a concept known as "perfect secrecy". Unfortunately, it comes with some major drawbacks: you have to generate a truly random key the length of your message for each message you want to send and you have to share those keys before exchanging messages. The scheme is fine for sending short messages, but is impractically inefficient for encrypting large amounts of data.

2 Modular Arithmetic

In the interest of trying to come up with a method of encryption that deals with the shortcomings of the one-time pad, let's examine the mathematical system created by our adding and subtracting of letters. Notice we wrote some funny things earlier, like $30 = 4$. That's because, in our system, both were ways of representing the letter E. In essence, we described a system where there are only 26 different numbers, or in other words, every number falls into one of 26 different classes.

After a little bit of poking and consideration, it might come to mind, that there are 26 possible remainders when dividing by 26. That is, every integer, k, can be written as $k = 26q + r$ where q and r are integers and $0 \leq r \leq 25$. We call q the *quotient* and r the *remainder*. In fact, instead of 26, we could use any positive integer

n. That is to say, if k is any integer and n is a positive integer, then there exist integers q and r with $0 \leq r \leq n-1$ such that $k = qn + r$.

If k has remainder r when divided by n we say that $k \bmod n = r$. If a and b have the same remainder when divided by n, we say that they are *congruent* mod n and write $a \equiv b \pmod{n}$. It is easy to verify that if $a \equiv b \pmod{n}$ and $c \equiv d \pmod{n}$, then $a + c \equiv b + d \pmod{n}$. This allows us to do addition mod n in a very natural way. For example, we can see that $40 \bmod 26 = 14$ and that $36 \bmod 26 = 10$ and so we can say that $40 + 36 \equiv 10 + 14 \equiv 24 \pmod{26}$. This gives us a method to do addition mod n.

Similarly, if $a \equiv b \pmod{n}$ and $c \equiv d \pmod{n}$, then $ac \equiv bd \pmod{n}$. For example,

$$40(36) \equiv 14(10) \equiv 140 \equiv 10 + 26(5) \equiv 10 + 0(5) \equiv 10 \pmod{26}.$$

(Notice we used the fact that $n \equiv 0 \pmod{n}$ here. This is in a certain sense the defining property of the number system mod n. It's what makes it so that there are only n distinct numbers mod n.) Thus, we can do multiplication mod n as well.

One thing you might have noticed in the above, is we can do addition and multiplication normally and then do reductions mod n or do reductions before these operations and we get the same result. That is to say we can do either $7(5) \equiv 35 \equiv 3 \pmod{4}$ or $7(5) \equiv 3(1) \equiv 3 \pmod{4}$. Hence, we can say that $(-1)k \equiv -k \pmod{n}$. Since we can do addition and multiplication mod n, this means we can do subtraction mod n as well. We can write $m - k \equiv m + (-k) \equiv m + (-1)k \mod n$. Anytime a number system includes addition, multiplication, and some concept of "-1", subtraction can be done in that system as well.

The question remains: "What about division?" For various reasons, it makes sense to view division as "inverse multiplication" in the context of modular arithmetic. In other words, we want to figure out what number to multiply k by to get 1 mod n. Does such a number even exist? Well, suppose we want to solve this for $k = 3$ and $n = 26$. Notice that $3(9) \equiv 27 \equiv 1 \pmod{26}$ and so we can say that 3^{-1} (three inverse) is 9 mod 26. What if we instead set $k = 2$? In this case notice that a solution to the congruence $2x \equiv 1 \pmod{26}$ means that there is an integer solution to the equation $2x + 26y = 1$. But we know that an integer solution to such an equation cannot exist, because the lefthand side is even for every choice of integers x and y, while the righthand side is odd. Therefore we conclude that 2 does not have an inverse mod 26. Since, in general, not all (non-zero) numbers have multiplicative inverses we cannot do division mod n.

3 Problem Set

1. Use a one-time pad system to:

   (a) Encrypt the message "UWMATHCIRCLES" with the key "BYTJPHZQIHXNB."

   (b) Decrypt the message "DCNWFBPIGTRAEU" with the key "OYWRBZWQCRAWCW."

2. Find the following remainders:

   (a) 16 mod 5

   (b) -20 mod 11

   (c) $-37(50) + 23(18)$ mod 8

   (d) 55145973736535916503 mod 10

   (e) 2^{2014} mod 3

3. A number is a sequence of digits $k = a_n a_{n-1} \cdots a_1 a_0$. (For example, the number 145 has digits $a_0 = 5$, $a_1 = 4$, and $a_2 = 1$.) Hence, k can be written as $k = a_n \cdot 10^n + a_{n-1} \cdot 10^{n-1} + \cdots + a_1 \cdot 10^1 + a_0 \cdot 10^0$. So $145 = 1 \cdot 10^2 + 4 \cdot 10 + 5$. Use this fact to prove that if a number, $k = a_n a_{n-1} \cdots a_1 a_0$, is divisible by 9 (i.e. has remainder 0 when divided by 9), then the sum of its digits is also divisible by 9.

4. (a) Suppose you know a message $m = m_0 m_1 \cdots m_n$ (each m_i is a letter) was encrypted using a one-time pad to the ciphertext $c = c_0 c_1 \cdots c_n$. Explain how to find the key $k = k_0 k_1 \cdots k_n$ used to encrypt the message.

   (b) Using your method in part (a), find the key used to encrypt "COMPLETELYRANDOM" to "PCFPCEGHZKTHNQQQ."

   (c) In parts (a) and (b), you saw that for every message, m, and every ciphertext of the same length c, there is a key that encrypts m to c. Explain why this means that given only a ciphertext that you know was encrypted with a one-time pad, you have no information about the original message (other than its length). (Perfect secrecy is the property of the ciphertext giving no information about the message.)

5. This exercise examines what happens when a key is re-used and why the two-time pad is insecure.

   (a) Suppose you know that two ciphertexts, c_1 and c_2, have been encrypted using the same one-time pad. Suppose also that you eventually learn the plaintext message, m_2, corresponding to c_2. Explain how to decrypt c_1.

(b) Suppose c_1 = "KQKZRO", c_2 = "NBWDCH", and m_2 = "DEFEND". Use your method from part (a) to decrypt c_1.

(c) Even if you don't know m_2, knowing that two messages were encrypted with the same key can give a lot of information. Why might that be the case? What operation can you perform on the two ciphertexts to make them easier to analyze? As a hint, you want to somehow remove the randomness added by the key.

(d) [Challenge Problem] Suppose c_1 = "DYXMHRDMCLVY" and c_2 = "NPCTIAJBJSOJ" have been encrypted using the same one-time pad and further you know that the plaintext messages are 12-letter words. Decrypt the two ciphertexts.