Vol.9.Issue.3.2021 (July-Sept)
©KY PUBLICATIONS

http://www.bomsr.com
Email:editorbomsr@gmail.com

RESEARCH ARTICLE

BULLETIN OF MATHEMATICS AND STATISTICS RESEARCH

A Peer Reviewed International Research Journal

INTERNATIONAL STANDARD SERIAL NUMBER 2348-0580

MATHEMATICS OF ERROR CORRECTING CODES

Dr. R. SIVARAMAN
Associate Professor, Department of Mathematics
Dwaraka Doss Goverdhan Doss Vaishnav College, Chennai, India
National Awardee for Popularizing Mathematics among masses
Email: rsivaraman1729@yahoo.co.in
DOI:10.33329/bomsr.9.3.61

ABSTRACT
Transmission of messages and details are usually done using codes. In this paper, my focus will be on words using the alphabets from the set $S = \{0, 1\}$, the letters of which are usually called bits. I will prove the distance between binary words form a Metric Space and prove an important theorem concerning perfect codes. These methods will provide a way of correcting the errors of the codes during transmission through efficient manner.
Keywords: Binary Words, Code, Hamming Distance, Metric Space, Ball, Perfect Codes.

1. Introduction

One of the most crucial aspects of discovery in today's digital world is concerned about efficiency and safety of transmission of data. Usually the data is transmitted through codes of different kinds. In this paper, I will discuss about binary codes, meaning the codes are formed only using the digits 0 and 1. Assuming that the binary codes are of length n, I will define the distance between two binary codes and prove that it forms a metric. In this paper, I prove a theorem related to code being perfect and discuss some examples related to these concepts. The mathematics used for this purpose enables one to correct the errors in codes significantly better.

2. Definitions

2.1 A binary word is a word comprising only of two digits 0 and 1. An n-bit word is also known as a binary word of length n. For example, 101 is a 3–bit binary word of length 3, where as 0011010 is a binary word of length 8.

2.2 The set of all binary words of length *n* is denoted by S^n.

The total number of binary words of length *n* is 2^n because each bit of length 1 can be filled with either 0 or 1, so the total number of binary words of length *n* by multiplication theorem of counting = $2\times 2\times 2\times\cdots\times 2 = 2^n$. Thus, $|S^n| = 2^n \quad (2.1)$

3. Distance between binary words

3.1 Suppose *x* and *y* are two binary words of length *n*. The distance between *x* and *y* denoted by $d(x,y)$ is defined as the number of places in which the bits (digits) of *x* and *y* differ. This distance is called Hamming Distance.

For example, if *x* = 101, *y* = 110 then $d(101,110) = 2$ since *x* and *y* differ in second and third bits.

I now prove an important theorem regarding Hamming distance between two binary words.

3.2 Theorem 1

(S^n, d) is a metric space (3.1)

Proof: First, we note from (2.1) that S^n contains 2^n binary words each of length *n*. To prove (3.1), I will prove that the Hamming distance two binary words $d(x,y)$ satisfies the axioms of a metric defined for a metric space.

(i) If $x, y \in S^n$ are two binary words of length *n*, then by definition 3.1, $d(x,y)$ is the number of places in which the bits of *x* and *y* differ. If all the bits of x and y are equal then $d(x,y)$ = 0, otherwise $d(x,y) > 0$. In fact, $0 \le d(x,y) \le n$ (3.2) Hence $d(x,y) \ge 0$ for all $x, y \in S^n$.

(ii) If *x* = *y* then all the bits of *x* and *y* will be equal. In this case, $d(x,y) = 0$. If $x \ne y$ then bits of *x* and *y* will differ at least in one place. Hence in this case, we get $d(x,y) > 0$.

(iii) By definition the Hamming distance $d(x,y)$ represents the number of places in which the bits (digits) of *x* and *y* differ. But the number of places in which the bits of *x* and *y* differ will be same as that of number of places that *y* and *x* differs. Hence, $d(x,y) = d(y,x)$ for all $x, y \in S^n$.

(jv) Let $x, y, z \in S^n$ be any three binary words. The binary words *x* and *z* cannot differ from each other in a place where neither of them differs from *y*. Being binary words, they also cannot differ from each other in a place where both of them differ from *y*. It follows that $d(x,z)$ is the sum of the number of places where *x* differs from *y* but *z* does not, and the number of places where *z* differs from *y* but *x* does not. Now the first term in this sum is at most $d(x,y)$, the number of places where *x* differs from *y*, and the second is at most $d(y,z)$, the number of places where *y* differs from *z*. Hence we get $d(x,z) \le d(x,y) + d(y,z)$. This proves the triangle inequality property.

Hence, we see that the Hamming distance *d* defined on S^n is a metric. Therefore (S^n, d) becomes a metric space. This completes the proof.

4. Error Correcting Codes

4.1 Definition

The triplet (n,k,d) is defined as a code, in which, there are *k* binary words each of length *n* and *d* is the minimum distance between any pair of distinct words. We denote a code by $C=(n,k,d)$.

We note that any code *C* is a subset of S^n, since $k \le 2^n$. Also from (3.2) we know that $0 \le d \le n$

4.2 As an example, for the code $C=\{00000,11101,10011,01110\}$ (4.1) we have *n* = 5, *k* = 4. According to the definition 4.1, *d* is the minimum Hamming distance between any pair of binary words among the four given words in *C*. Finding the distance between each pair of words from *C* we get $d(00000,11101)=4, d(00000,10011)=3, d(00000,01110)=3, d(11101,10011)=3,$ $d(11101,01110)=3, d(10011,01110)=4$. Since the minimum Hamming distance is 3, we find that for the code *C* in (4.1), *d* = 3. Hence, the code in (4.1) is given by $C=(5,4,3)$.

4.3 Nearest – Neighbor Decoding

Nearest-neighbor decoding refers to a process by which an erroneous binary word *x* is corrected to a legitimate codeword *c* in a way that minimizes $d(x,c)$.

4.4 Definition

An *r* – error correcting code is defined as a code for which nearest neighbor decoding reliably corrects as many as *r* errors.

We now see a theorem which gives necessary and sufficient condition for a code being *r* – error correcting code.

4.5 Theorem 2

A code $C=(n,k,d)$ is an *r* – correcting code if and only if $2r+1 \le d$

Proof: First, we note that a code $C=(n,k,d)$ can reliably detect as many as *d* – 1 errors. To determine how many errors the code *C* can correct reliably, we consider the possibility that, for some erroneous binary word *x*, there is a tie for the codeword nearest to *x*. That is, $d(c,x) \ge r$ for every $c \in C=(n,k,d)$, such that $d(c_1,x)=r, d(c_2,x)=r$. That is, we assume that there is a tie between the code words c_1, c_2. By triangle inequality property of the Hamming distance *d* we get $d(c_1,c_2) \le d(c_1,x)+d(x,c_2)=r+r=2r$. Since, *d* is the minimum distance between any pair of words, $d(c_1,c_2) \le 2r$ guarantees that no such situation occur when $2r<d$. Thus the code will be an *r* – correcting code if and only if $2r+1 \le d$ This completes the proof.

4.6 We note that theorem 2, helps us to determine the value of *r* for a code *C* to be an *r* – correcting code. For example, let us consider the code $C=(5,4,3)$ given in (4.1). This will be an *r* – correcting code such that $2r+1 \le 3$. Hence *r* = 1 making the code $C=(5,4,3)$ a 1 – correcting code. We also note that the code $C=(5,4,3)$ is not a 2 – error correcting code because $2r+1=2(2)+1=5>d=3$.

Similarly, the code $C=\{00000,11111\}$ has *d* = 5. To know how many errors it can correct, we need to find *r* such that $2r+1\le 5$ giving *r* = 2. It is easy to see that it is not a 3 – correcting code because $2r+1=2(3)+1=7>d=5$. So the code is a 2 – correcting code but not a 3 – correcting code.

5. Bound for number of code words

5.1 Let *x* be a binary word of length *n*. The ball of radius *r*, centered at *x*, is defined by

$$B_r(x)=\{y\in S^n / d(x,y)\le r\} \quad (5.1)$$

That is, $B_r(x)$ consists of set of all binary words which differ from *x* in at most *r* bits.

We now discuss an important theorem regarding the bound for number of code words *k* in a *r* – correcting code of the form $C=(n,k,d)$.

5.2 Theorem 3

Let $C=(n,k,d)$ be an *r* – correcting code. The number of code words in the code *C* cannot be more than $\frac{2^n}{\sum_{m=0}^{r}\binom{n}{m}}$ where $\binom{n}{m}$ is the binomial coefficient. That is, $k\le\frac{2^n}{\sum_{m=0}^{r}\binom{n}{m}}$ (5.2)

Proof: Let $C=(c_1,c_2,...,c_k)$ be an *r* – correcting code. Let $B_r(c_i)$ be an ball centered at the codeword c_i and radius *r*, where $i=1,2,3,...,k$. Now if c_i,c_j are different code words in *C*, and if we assume that $x\in B_r(c_i)\cap B_r(c_j)$ then by (5.1), we have $d(c_i,x)\le r$ and $d(c_j,x)\le r$. Hence by triangle inequality property we get $d(c_i,c_j)\le d(c_i,x)+d(x,c_j)\le r+r=2r<2r+1\le d$. Thus the distance between the code words c_i,c_j is less than *d*, contradicting the fact that *d* is the minimum distance between any pair of code words in *C*. Hence if $c_i\ne c_j$ then $B_r(c_i)\cap B_r(c_j)\ne\phi$

Let $c_i\in C$ be any codeword of length *n*. Then there will be $\binom{n}{m}, m=0,1,2,3,...,r$ places in which a binary word can differ from c_i in exactly *m* places.

Thus the ball $B_r(c_i)$ will have $\binom{n}{0}+\binom{n}{1}+\cdots+\binom{n}{r}=\sum_{m=0}^{r}\binom{n}{m}$ binary code words where $1\le i\le k$

Now the number of different binary words of length *n* contained in the union of length *n* contained in the union of *k* balls is $k\times\sum_{m=0}^{r}\binom{n}{m}$. But this collection of different binary words cannot exceed the total number of binary words of length *n* in S^n. By (2.1), we know that $|S^n|=2^n$. Hence we have $k\times\sum_{m=0}^{r}\binom{n}{m}\le 2^n$ proving (5.2). This completes the proof.

6. Perfect Codes

6.1 Definition

A code $C=(n,k,d)$ is said to be perfect if $k\times\sum_{m=0}^{r}\binom{n}{m}=2^n$ where $r=\left\lfloor\frac{d-1}{2}\right\rfloor$ (6.1)

From (6.1), we see that the code *C* is perfect if the number of binary words obtained through *k* balls is equal to the total number of binary words of length *n* namely 2^n.

The following theorem provides the necessary and sufficient conditions for a code to be a perfect code.

6.2 Theorem 4

Let $C=(n,k,d)$ be an code such that $r=\left\lfloor\frac{d-1}{2}\right\rfloor=1$. Then *C* is a perfect code if and only if there exists an integer $s\geq 2$ such that $n=2^s-1$ and $k=2^{n-s}$ (6.2)

Proof: If *C* is a perfect code, then by (6.1), we have $k\times\sum_{m=0}^{1}\binom{n}{m}=2^n$. This gives $2^n=k(1+n)$.

Hence $k=\frac{2^n}{n+1}$. Since *k* is an integer, $n+1$must divide 2^n. This is possible only if $n+1=2^s$ for some positive integer $s\leq n$. Thus $n=2^s-1$and $k=\frac{2^n}{2^s}=2^{n-s}$.

If $n=2^s-1$and $k=2^{n-s}$then $k\times\sum_{m=0}^{1}\binom{n}{m}=2^{n-s}(1+n)=2^{n-s}\times 2^s=2^n$. Hence by definition, *C* is a perfect code. This completes the proof.

6.3 Illustration

Using theorem 3 in 6.2, we can show that the code $C=(7,16,3)$ is perfect by choosing *s* = 3. In fact, for *s* = 3 and we get $n=2^3-1=7,k=2^{7-3}=2^4=16$, satisfying the conditions in (6.2).

7. Conclusion

By considering binary code words of length *n* through the set S^n and defining Hamming distance between the words, I proved that (S^n,d) is a metric space in theorem of section 3.2. Theorem 2 of section 4.5 provides a nice condition for a code $C=(n,k,d)$ to become an $r-$correcting code. Theorem 3 of section 5.2 provides an upper bound for number of code words of length *n*. In theorem 4 of section 6.2, I proved that the upper bound obtained in theorem 3, will be obtained if the code is a perfect code. Illustrations and remarks were provided wherever necessary for better understanding.

Though the ideas discussed in this paper are quite significant, still there is lot of scope to discuss about the codes especially if *n* is quite large. The idea of forming efficient codes will make the digital transaction safe and secure. This paper offer chance to understanding the behavior of such codes and provide an opportunity to explore further.

REFERENCES

[1]. F. J. MacWilliams and N. J. A. Sloane, The Theory of Error-Correcting Codes, North-Holland, New York, 1977.

[2]. N. J. A. Sloane, A Short Course on Error Correcting Codes, Springer-Verlag, New York, 1975.

[3]. R. Sivaraman, Equivalence Relations and Bell Numbers, International Journal of Psychosocial Rehabilitation, Vol 24, Issue 04, 2020, pp. 10639 – 10647.

[4]. R. Sivaraman, Enumeration of Relations and Functions, Journal of Xi'an Shiyou University, Natural Science Edition,, Volume 16, Issue 9, 2020, pp. 314 – 319.

[5]. D. G. Hoffman, D. A. Leonard, C. C. Lindner, C. A. Rodger, and J. R.Wall, Algebraic Coding Theory, Charles Babbage Research Centre, Winnipeg, 1987.

[6]. P. A. MacMahon, Combinatory Analysis, Chelsea, New York, 1960.