How can we build-in ways of finding and fixing transmission errors when using binary codewords?

Definitions:

Codes composed of sequences of binary digits are called ____________________. A __________________________ is an extra digit appended to a message for error-detecting.

Example: Determine the parity check digit that should be appended to each block so that the total number of 1's is even.

(a) 01000111

(b) 10111001

(c) 10111010

Key Assumptions and Formula:

1. The probability of changing a 0 to a 1 and of changing a 1 to a 0 is the same.
2. The probability of an error in each digit is the same and is independent of whether there are errors in other digits (i.e., in probability terms, the transmission of any two digits are independent events).
3. The probability of an error in any digit is small, so the probability of the correct transmission of a block is greater than the probability of a single error in the block and the probability of a single error is greater than the probability of two or more errors.

Formula: $\mathrm{C}(n,k)p^k(1-p)^{n-k}$, where p is the probability of error in a single digit, k is the number of errors, and n is the length of the message

Why should this formula make sense?

Example: Suppose the probability of error in transmission of a single digit is .01. What is the probability of having exactly 1 error in a message of length 9?

Example: Suppose the probability of error in transmission of a single digit is .01. What is the probability of having exactly 0 errors in a message of length 9?

Example: Suppose the probability of error in transmission of a single digit is .01. What is the probability of having exactly 2 errors in a message of length 9?

Definitions:

Suppose we want to transmit information in blocks of k binary digits. Each block is a __________________ and k is the ____________. We transmit ____________ that include more digits to permit error-detection. If each message word has length k and each codeword has length n, the coding scheme is a __________________. The ____________ of a (k, n)-block code is the ratio k/n.

Example: Suppose we want to transmit messages using ASCII, which uses 8 digit names for each symbol, and we will include a parity check digit for error detection. What are k and n and what is the efficiency of this (k, n)-block code?

Example: Suppose we have 8 digit message words and we repeat the message for error detection (e.g., message 00000001 00100000 is sent as 00000001 00000001 00100000 00100000). What are k and n and what is the efficiency of this (k, n)-block code?

Example: Suppose we have 8 digit message words and we repeat the message twice for error detection (e.g., message 00000001 00100000 is sent as 00000001 00000001 00000001 00100000 00100000 00100000). What are k and n and what is the efficiency of this (k, n)-block code?

Definitions:

For this coding scheme, we need a one-to-one function that ____________ (call this function E) message words as codewords and an inverse function that ____________ (call it D). Thus for message $w_1, w_2, ..., w_m$, we transmit $E(w_1), E(w_2) ..., E(w_m)$ and regain the original message by taking $D(E(w_1)) = w_1$, etc. If someone receives a word z that is not a codeword, they know an error happened. Usually the receiver would decode z as the codeword that differs from z by the fewest digits. This is called ________________________.

Example: While less efficient, the third option (e.g., message 00000001 00100000 is sent as 00000001 00000001 00000001 00100000 00100000 00100000) permits some error-correction. How could we use the three copy version of the message to determine the likely intended message?

Definition:

For two codewords c_1 and c_2 of the same length, the __________________ between c_1 and c_2 is defined to be the number of digits in which c_1 and c_2 differ, denoted $d(c_1, c_2)$.

Example: What is the Hamming distance between the following codewords?

(a) $d(01000111, 01010101)$

(b) $d(10111001, 10111011)$

(c) $d(00000000, 11111111)$

Example: If you add two codewords over $\mathbb{Z}_2$ for each digit, what will happen?

Triangle Inequality (Theorem 3.6):

If c_1, c_2, and c_3 are any codewords of the same length, then $d(c_1, c_3) \leq d(c_1, c_2) + d(c_2, c_3)$.

Why should this make sense?

Theorem 3.7:

Consider a block code in which m is the minimal Hamming distance between distinct codewords.

(a) This coding scheme can detect r or fewer errors if and only if $m \geq r + 1$.

(b) This coding scheme can correct r or fewer errors if and only if $m \geq 2r + 1$.

Why should this make sense?

Example: Suppose the minimal Hamming distance between codewords in a certain block code is 4. What is the maximum number of errors that can be detected and what is the maximum number of errors that can be corrected?

Example: Suppose the minimal Hamming distance between codewords in a certain block code is 16. What is the maximum number of errors that can be detected and what is the maximum number of errors that can be corrected?

Example: Suppose the following set comprises all the possible codewords. What is the minimal Hamming distance between codewords for the set?

(a) $\{01011, 00110, 00111, 11000, 10101\}$

(b) $\{101011, 001101, 100111, 110001, 000001\}$

Example: Suppose we used the parity check digit method as before (length 8 message words based in ASCII, length 9 codewords that must have an even number of 1s). What is the minimal Hamming distance in this context? What does this suggest about error-detection and error-correction?

Example: If a set of codewords contains the codeword in which each digit is 0, what can be said about the minimal Hamming distance between two codewords?

Rachel Rupnow. Compiled: November 13, 2022