Section 4

Applications of Metric Spaces

Introduction

In this section we explore two applications of metric spaces.

The Hamming Metric

In our society, a great deal of information is communicated electronically. Bank transactions, television programs, military communications, cell phone calls, digital images, and almost any interchange one can think of either can be or is digitized and transmitted electronically. In many situations we need to compare one set of data to another (e.g., Internet searches for text strings or image matches, DNA strands), and metrics are often used for this purpose. Computers work in a binary system, that is they recognize only zeros and ones. So a digital text message is a string of zeros and ones. That is, a digital message is a collection of elements in the space X^n for some positive integer n, where $X = \{0, 1\}$. Each element in X^n is called a *word* - that is, a word is an element in X^n denoted in the form $(x_1, x_1, \ldots, x_n)$. Just like in the English language, where not every combination of letters corresponds to words that make sense, not every word is recognizable as part of an intelligible message. We might, for example, code the letters of the alphabet by assigning numbers 1-26 to the letters, then make them elements of X^n by converting to binary. The collection of all intelligible words is called a *code*. So a code is just some subset of X^n that all parties agree are sensible words. The words in a code are called *code words*. To deal with problems that occur in transmitting digital messages, like scrambling (*encoding*) messages, unscrambling (*decoding*) messages, and detecting and correcting errors in messages, it is useful to have a way to measure distance between words. One way is to use the Hamming metric.

Definition 4.1. Let $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ be words in X^n. The **Hamming distance** d_H between x and y is

$$d_H(x, y) = \sum_{i=1}^{n} |x_i - y_i|.$$

Recall that for each i both x_i and y_i are either 0 or 1. So

$$|x_i - y_i| = \begin{cases} 0 & \text{if } x_i = y_i \\ 1 & \text{if } x_i \neq y_i. \end{cases}$$

In other words, $d_H(x, y)$ counts the number of components at which x and y are different.

Activity 4.1.

(a) Explain why d_H is a metric.

(b) Suppose we create a code

$$C = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8\}$$

in X^6, where

$$\begin{array}{lll} c_1 = (0,0,0,0,0,0), & c_2 = (0,0,0,0,1,1), & c_3 = (0,0,0,1,0,1), \\ c_4 = (0,0,1,0,0,1), & c_5 = (0,0,0,1,1,0), & c_6 = (0,0,1,0,1,0), \\ c_7 = (0,0,1,1,0,0), & c_8 = (0,0,1,1,1,1). & \end{array}$$

That is, the words c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8 are the only words that can comprise a message.

Find $d_H(c_2, c_8)$.

(c) Suppose we are on the receiving end of the message

$$(0,0,0,1,1,1)\ (0,0,1,1,0,0)\ (1,0,0,0,0,0)\ (0,0,0,0,1,1)\ (0,0,1,0,0,1).$$

i. How do we know that an error has occurred in transmission of the message we received?

ii. To correct the errors in this received message, we replace the incorrect words with the code word(s) in C closest to them. Correct this message. (Note that there may be more than one possible substitution. Find all of the possibilities.)

The Levenshtein Metric

The Levenshtein metric is one measure of distance that researchers use to understand DNA. DNA is composed of double chains of nucleotides, which wind together to form a double helix. The nucleotides come in four types: adenine (A), cytosine (C), guanine (G), and thymine (T). The nucleotides in the two chains of a DNA strand pair together, (A with T, and C with G), so the nucleotides in one chain determine the nucleotides in the other. Therefore, we can represent a DNA strand with a string of letters from the alphabet $\{A, C, G, T\}$. One problem DNA researchers have is how to compare two strands of DNA, and the Levenshtein metric is one way that the distance between strands can be measured. Other metrics could be used, but the Levenshtein metric is appropriate to the task for several reasons. During evolution, changes in DNA sequences arise due to nucleotide substitution, or the insertion or deletion of nucleotides. These evolutionary changes

can be modeled by the operations that determine the Levenshtein distance better than other metrics. In addition, the Levenshtein metric can be used to calculate distances between strings of different lengths. The Levenshtein metric also has applications in spell checkers, speech recognition, and automated plagiarism detection. To understand how the Levenshtein metric is calculated, consider the question of how far apart the words "green" and "grease" are.

To compare these words, we have to be able to change letters, and add or delete letters. If $x = x_1x_2 \cdots x_n$ is a string of letters, we allow the following operations:

a deletion: replace x with $x_1 \cdots x_{i-1}x_{i+1} \cdots x_n$ for some i,

an insertion: replace x with $x_1 \cdots x_i y x_{i+1} \cdots x_n$, where y is an allowable letter and $0 \leq i \leq n$,

a substitution: replace x with $x_1 \cdots x_{i-1} y x_{i+1} \cdots x_n$, where y is an allowable letter and $1 \leq i \leq n$.

Activity 4.2.

(a) Using the allowable operations, change the word "green" into the word "grease". Specifically identify each operation you use. (Note: the intermediate strings of letters do not have to form recognizable words.) How many operations did you use?

(b) If it took three operations to transform "green" into "grease", we could say that the distance between "green" and "grease" is at most 3. However, it may be possible to transform "green" into "grease" in fewer than 3 operations, which might change our opinion of the distance between these words.

In general, to define the Levenshtein distance d_L between a sting x and a string y, let m_d denote the number of deletions, m_i the number of insertions, and m_s the number of substitutions we use to get from x to y. There may be many different combinations of m_d, m_i, and m_s that get us from x to y, so we want the smallest number.

Definition 4.2. The **Levenshtein distance** $d_L(x, y)$ between strings x and y is

$$d_L(x, y) = \min\{m_d + m_i + m_s\}.$$

Prove that the Levenshtein distance function is really a metric on the set of all possible words (sensical or nonsensical).

(c) A spell checker corrects the misspelled word "tupotagry". Using the Levenshtein metric, which word would the spell checker use as the closest to "tupotagry"? Why?

"topography" "topology" "tautology"