SNS COLLEGE OF TECHNOLOGY

Coimbatore-35.

An Autonomous Institution

COURSE NAME : 19CST203 - DATA ANALYTICS

II YEAR /IV SEMESTER

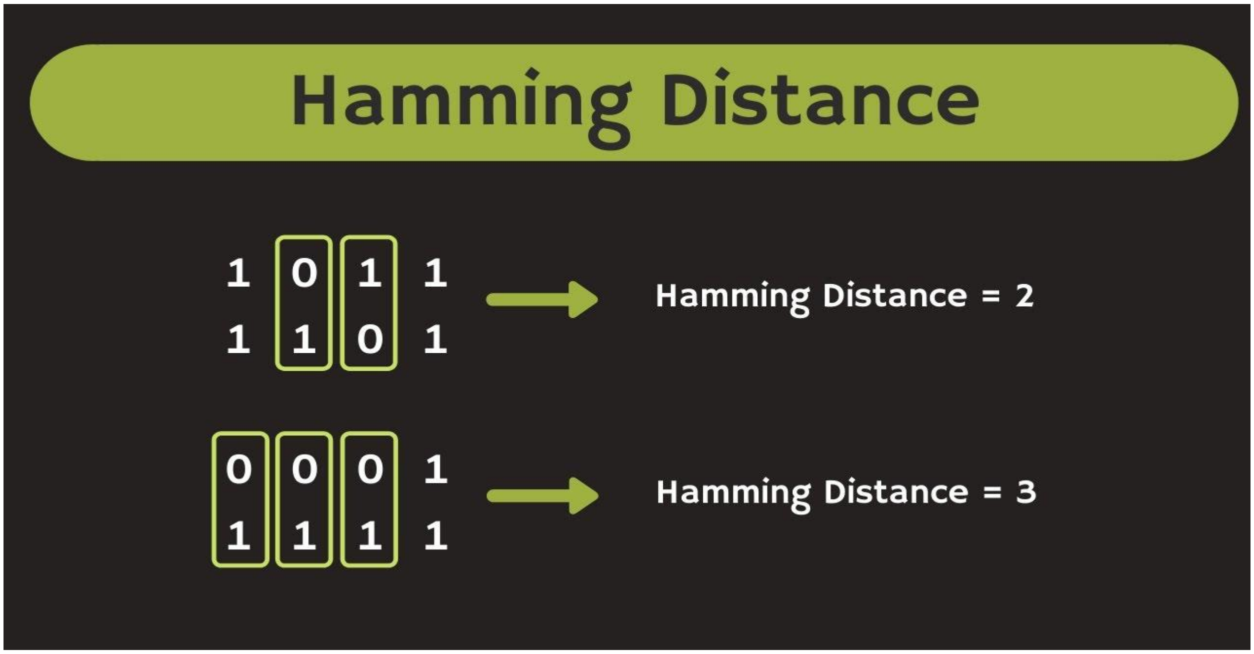
1.DISTANCE MEASURES FOR NON-CONVENTIONAL ATTRIBUTES

These attribute types, here termed “non-conventional’, include:

- biological sequences
- time series
- images
- sound
- video

The Hamming distance can be used for sequences of values and these values are usually characters or binary values. A binary value (or binary number) is either 1 or 0, meaning true or false, in general. The Hamming distance is the number of positions at which the corresponding characters or symbols in the two strings are different.

Hamming distance is a metric for comparing two binary data strings. While comparing two binary strings of equal length, Hamming distance is the number of bit positions in which the two bits are different. The Hamming distance between two strings, a and b is denoted as d(a,b).



For long texts, such as a text describing a product or a story, each text is converted into an integer vector using a method called the "bag of words". The bag of words method initially extracts a list of words, containing all the words appearing in the texts to be mined. Each text is converted into a quantitative vector, where each position is associated with one of the words found and its value is the number of times this word appeared in the text.

Let's understand this with an example. Suppose we wanted to vectorize the following:

- *the cat sat*
- *the cat sat in the hat*
- *the cat with the hat*

We'll refer to each of these as a text **document**.

Step 1: Determine the Vocabulary

We first define our **vocabulary**, which is the set of all words found in our document set. The only words that are found in the 3 documents above are: the, cat, sat, in, the, hat, and with.

Step 2: Count

To vectorize our documents, all we have to do is **count how many times each word appears**:

Document	**the**	**cat**	**sat**	**in**	**hat**	**with**
the cat sat	1	1	1	0	0	0
the cat sat in the hat	2	1	1	1	1	0
the cat with the hat	2	1	0	0	1	1

Now we have length-6 vectors for each document!

- *the cat sat*: [1, 1, 1, 0, 0, 0]
- *the cat sat in the hat*: [2, 1, 1, 1, 1, 0]
- *the cat with the hat*: [2, 1, 0, 0, 1, 1]

Notice that we lose contextual information, e.g. where in the document the word appeared, when we use BOW. It's like a literal **bag-**of-words: it only tells you *what* words occur in the document, not *where* they occurred.

To calculate the distance between images, two distinct approaches can be used. In the first, features associated with the application can be extracted from the images. For example, for a face recognition task, the distance between the eyes can be extracted. Ultimately, each image is represented by a vector of real numbers, where each element corresponds to one particular feature. In the second approach, each image is initially converted to a matrix of pixels, where the size of the matrix is associated with the granularity required for the image. Each pixel can then be converted into an integer value.