Preston Calderon

Nick Stapleton

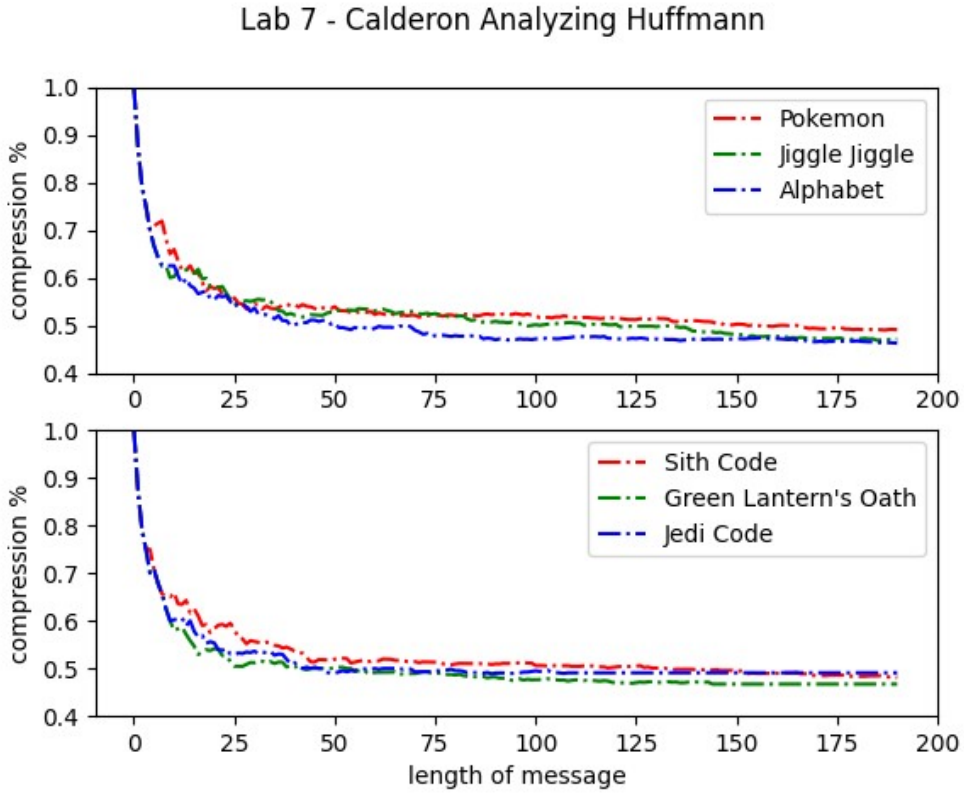
CSC 202

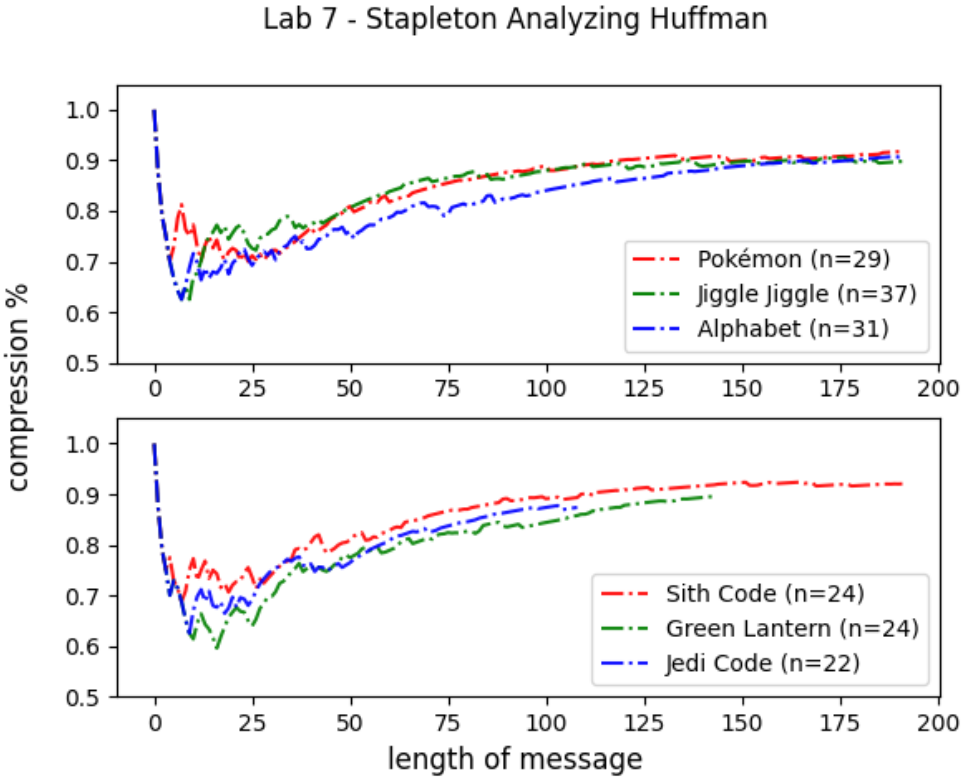
5 June 2023

Lab 7 Report

The compression efficiency of huffman encoding algorithms is dependent on the size of the input that needs to be compressed, and the probability in which the values within appear throughout the input. For example, a simple string ‘hello there’ has values or characters which have a similar probability of appearing within the string compared with each other. Strings that do not have many redundant values (i.e. strings that have all unique chracters) will result in a worse compression ratio compared with strings that have values of high redundancy (i.e. their values repeat many times.

The graph output from the lab exhibits the opposite behavior. First, small length of messages results in the best compression ratios. As the message size increases (redundancy increases), the compression ratios get worse. This is contrary to what appears in the professor's graphs. I believe these differences may lie in how our huffman code algorithms are written, and I believe there is further optimization that can be performed on the implementation I wrote to achieve more positive results.


Lab 7 - Calderon Analyzing Huffmann
compression %
Pokemon
Jiggle Jiggle
Alphabet
Sith Code
Green Lantern's Oath
Jedi Code
length of message


Lab 7 - Stapleton Analyzing Huffman
compression %
Pokémon (n=29)
Jiggle Jiggle (n=37)
Alphabet (n=31)
Sith Code (n=24)
Green Lantern (n=24)
Jedi Code (n=22)
length of message

As complexity of the input increases and the probabilities of its associated letters appearing in the messages decreases, Huffman Tree height increases, which can also be observed in the increases in bit string size. With an increase in redundancy of the letters in the messages, it is possible to have a more balanced tree, as all of the weights of the letters will be similar, resulting in a similar amount of nodes in each tree level. If probabilities of letters appearing are very similar, the values have similar weights, and will result in a more balanced Huffman tree.