Fathan Ananta Nur
236-D8749

44135 Applied Information Theory
Summary

1. What did I learn in this course?

In the first class, we focused on chaos theory, starting with the logistic and skew Bernoulli maps. By adjusting the parameters, we observed significant changes in graph plot of real-valued series. Even minor differences in the initial value, led to vastly different mid-series graph plot, demonstrating the sensitivity of the parameter. Although the graph plot seemed always different, they were uniformly distributed for all initial values and c value. In this class, we also studied bit-shift errors, which occur when bits in a digital data stream are shifted left or right, leading to misalignment and possible data corruption. This phenomenon can be illustrated by a graph where the values abruptly stop and return to zero.

In the second class, we examined the sequence distribution and the impact of c values on graph plot. We created memoryless information using skew Bernoulli maps, comparing theoretical and calculated probabilities. When c value equaled the threshold t, the values aligned closely. We also explored the relationship between entropy and compression ratio when binary sequences from skew tent maps were saved and compressed. As c value approached 0.5, entropy increased and compression ratio decreased.

The third class involved generating Markov binary sequences using piecewise linear chaotic mapping (PLM3). Depending on initial values $p1$ and $p2$, the function's complexity increased, yet the calculated probabilities remained close to theoretical values. We also compared entropy and compression ratios for Markov sequences, finding that higher $p1$ and $p2$ increased entropy and decreased compression ratio.

In the fourth class, we compared undetected error probabilities for different bit error rates p and initial values in Markov errors using (4-3) single parity check code. For the same p, memoryless errors had lower undetected error probabilities up to $p = 0.25$, but beyond that, Markov errors showed lower probabilities, depending on $p1$ and $p2$. Thus, undetected error probabilities varied with initial values for both error types.

In the last class, we compared memoryless and Markov-type errors using a (7,4)-Hamming code at various bit error probabilities. For p is lower around 0.2 and 0.25, memoryless before decoding had a lower error probability, while for p value is higher than that, error probability after decoding will rise higher. In contrast, the error probability for Markov-type decoding varied with the initial values of $p1$ and $p2$. Additionally, decoding reduced the bit error probability for Markov-type error across p value.

2. What are interesting and/or difficult points?

Before taking the class, I had some understanding of chaos theory, particularly the concept of the "butterfly effect," popularized by *internet memes* (e.g., "if you moved a chair in the past, Taylor Swift would have invented the theory of relativity"). However, through the course and the assignment report, I uncovered several fascinating aspects. For example, examining the graph plots of output values showed an almost uniform distribution despite the non-linear and complex transitions. Even slight differences in initial values caused significant changes in transitions, underscoring the sensitivity of the skew Bernoulli map and linear chaos map to initial conditions. These findings, validated by our individual experiments, were captivating.

Despite this, I found it difficult to understand how chaos theory applies to real-world phenomena or specific fields. However, I learned that file compression tasks on computers depend on entropy levels, with higher entropy indicating greater "chaos" in the data set. This is one application of chaos theory. While intriguing, deepening my understanding of chaos theory remains challenging due to the extensive knowledge and mathematical expertise required.

3. Impression

At first, I was worried about understanding the class content since the lectures focused solely on theoretical explanations without practical problems. However, the slides included the algorithms for the assignments, making coding in Python relatively easy and engaging, and allowing me to explore the results with curiosity. Additionally, I was pleased to help other students with the coding assignments by explaining the code algorithms, even though I did not explain the theoretical material.