**Using ChatGPT for Working with GPX Data**

The Powers lab uses a boat with an integrated GPS logger and sonar device that both take continuous measurements while the boat is in motion. This boat is frequently driven on Lake Waco, so we have multiple gpx files, with lots of boat tracks, from multiple dates, spanning months to years apart. I am interested in developing a workflow to convert the gpx files into a more readable tabulated format, and use the data to create a bathymetric map of Lake Waco. The goal of this project was to prompt ChatGPT to help create this workflow, and to do as little manual coding and debugging as possible.

Since I had never worked with gpx files before, I began by doing a bit of research into how to work with them in R. I found a small chunk of code that was on the right track for what I needed and ran it with my gpx file of boat track data. Although this code did not immediately give me what I wanted, I fed the code and output to ChatGPT and asked it to tell me more about the structure of gpx files and the contents of my file. This helped me learn more about what I needed to do, and how to prompt ChatGPT to accomplish certain tasks. I then proceeded through a few iterations of ChatGPT providing code, and me providing outputs and error codes. After a handful of iterations and tweaks, I arrived at a functional script that would read my gpx file into R and generate a data frame with the relevant data from the gpx file.

Now that I had some clean tabulated data, I described the bathymetric map I wanted to create and asked ChatGPT about methods that could be used to generate that kind of plot. One of the methods it suggested was to make use of the mapping features included in the ggplot2 package. I had used a very similar approach for another project (in 5202), so I quickly repurposed the code I had previously written to generate the water depth and lake bed elevation plots. I wanted to see if ChatGPT could also help me generate a raster map that took the average of multiple data points to create a smoother bathymetric map. ChatGPT provided essentially functional code after only one prompt and was able to generate a very satisfying product after just a few more prompts to tweak to plot formatting.

For this project, I decided to type all my prompts for ChatGPT in a single conversation thread, rather than making different requests in multiple threads. I think this helped give ChatGPT more context, and a better idea of what I wanted. Showing ChatGPT functional code and examples of outputs in addition to error codes also seemed to be key for getting good responses. Finally, asking ChatGPT conceptual questions about the topic helped me understand what I needed to do, and helped me provide better feedback to ChatGPT. I would consider this project to be very successful because it resulted in the product I was hoping for, and I was able to accomplish it with ChatGPT faster than I would have on my own.