Christopher Maher

Self-Directed Class

5.5.2023

**Meta Cog Memo**

Looking back as I finish this self-directed topic, I cannot help but pause to realize all that I have had to learn this semester. Starting off I did break my learning into two different groups, and I want follow the pattern.

First, I started learning about data cleaning. Data cleaning is a massive field. I didn’t realize how important and how broad the styles, types and ways of cleaning are until I started learning more about it. Starting off, it’s important to identify the different ways that an error could occur. When I was learning this one of the top questions, I had was why is this important? I didn’t figure this out till later, but I consider it critical to cleaning the data. Data errors have very common patterns in how they occur. Because of this, if your project you’re coding requires for example a lot of user input from a form. This means that you should especially be checking and cleaning for spelling errors, correct data types, and formatting of the data. However, for example you probably won’t get any missing values if you set up the form correctly. Because of this if we can identify what type of errors are likely to occur in our process, we can focus more energy and time on creating methods for cleaning errors that are more likely to occur than not. Continuing with this I will quickly list out the different types of data errors that are common: Missing values, bad values, and duplicates. Each of these are common data errors you can find in any data set. Each of these data errors has multiple solutions to them. One of the solutions I used for missing values and bad values, for example is using a Data Schema. Python allows us to implement some seudo SQL and create schemas for DataFrames. When we do so it’ll treat the DataFrame under the strict rules of a SQL Schema. This means that it will immediately throw an error when a bad value is found or missing values. This is critical during the data cleaning stage since without cleaning wouldn’t be nearly as easy. Secondly.