Brynn Dafoe
DSCI 510
Progress Report
3109669210
Project Scope Update:

- I originally planned on using the logistic regression model to make predictions for the top 10 finishers, but being that the model only gives 1 if it predicts a person will make the top 10 and 0 if not with no way of ordering those top 10 people in order of 1st place to 10th place, I've decided I am going to have the model make predictions for the top 5 so I can narrow down the possible winners. For my variables I was originally planning on including a skier's World Cup podiums, World Cup starts, and years since World Cup debut, but through scraping my data I found that the pages that hold this data are set up kind of oddly and I was having too difficult of a time gathering the data for these three variables, so I've decided to not use them (I might add this data in later on my own time when the class has ended). When choosing logistic regression as the model I wanted to use I did not have a full understanding yet of what the input and output would be, but I know now that the model only takes one CSV file as input. Because of this, I have decided that my variables will be: Name (identifier), FIS Code (identifier), Age, Nation, 2018 Olympic Cycle : [Average Final Score, Average Time Points, Average Jump Points, Average Turn Points, Standard Deviation of Final Scores], 2022 Olympic Cycle : [same variables as 2018], Olympic Year (to let model know which year the row is holding data points for), and Made Top 5 or Not (made top 5 or not for that particular Olympic cycle with a 0 or 1). For a skier that has been in both Olympic cycles, they will have two rows. Lastly, I was originally going to gather every single athlete from every single event. Realistically though, the top 25-30 finishers are pretty consistent with little variation in rankings for each event, so to narrow down the list of athletes I will need to gather URLs for for my Athlete_Data program, I am only going to take top 30 from each event. I am going to take the top 30 because the Olympic Results web pages only show the top 30, so I figure I should make it consistent.

Data Sources:

- All my data is coming from websites (my World Cup data is coming from web pages of downloaded PDF files). All of my data is being collected in CSV files. I have three groups of data currently: Athlete Data, World Cup Results Data, and Olympic Results Data. Each one has an identifier column, and the rest of the columns are data points that the logistic model will be able to use to create predictions. My Athlete Data includes: Name, FIS Code, Birthdate, Birth Year, Age, and Gender (although I am going to fix it later to only use Birth Year). My Olympic Results Data includes: Rank, Name, and Country. My World Cup Results Data includes: Rank, FIS Code, Name, Nation, Birth Year, Final Score, Time Points, Air Points, and Turn Points. I used different web-scraping tools for each program because each type of web page was set up differently. For Athlete_Data I was able to use BeautifulSoup to web scrape because all the data I wanted was in HTML format. For Olympic_Results I used json and regular expressions because the data I wanted was in a json embedded portion and I was having a hard time getting the data with BeautifulSoup. My World_Cup_Results uses pdfplumber because the data I want is only available via the downloadable PDFs.

Issues / Difficulties:

- Going into this I did not realize that the web pages I wanted to use were set up a little oddly so I had to do a lot of trial and error in order to find what would work to web scrape (hence why for each data scraping program I use a different method). My possible future difficulties: First, I will need to go through every single CSV file generated and clean the data. Second, I need to make at least three more programs. 1) A program that generates all the averages and standard deviations I want by parsing through my generated CSV files. 2) A program that will create one big CSV file as input for the logistic regression model. 3) The logistic regression model itself. I don’t think these things will necessarily be difficult, but tedious.