Case Study 4

Moral Machine Part 2

Due: XXX

**Moral Machine Part 2**

Congratulations! It is your second week on the job at Messla and your quick thinking creating the Trolley Problem algorithm for us last week impressed the big bosses at the top. They want to implement your solution ASAP. In order to do this, they will need to use the function that you created that determines who to save and extrapolate it so it can be used with new data that they haven't seen yet.

**Problem 1 (45 points) Get This Function Working With Our Historical Data**

First things first, we need to get your function operating with the data that we have available. Please write a function that takes in a filename (you can expect the file to be in the same format that we saw before - e.g., rowNum, age, gender, country, fault).

Your function should loop through this file and for every "incident" (every 2 rows) call your function that you created in Part 1: *whoDoesTheCarSave*.

The person that you choose to save should be added to a list of people to save.

Your function should **return** a list of who you have chosen to save for each scenario. For example, your return list might look something like:

[Driver, Driver, Pedestrian, Pedestrian, Pedestrian, Driver, Pedestrian, etc....]

Your code should use the csv.reader() method to open and parse the “trolley-problem.csv” file.

Use the following function header:

**def messlasMoralMachine(***fileName****)*:**

**Problem 2 (45 points) The Final Step**

Wonderfully done! Now as the final step to finish out your job for us, we will need to be able to send this data that you have created to the self-driving car so that it can run your function and know what to do! For this last step, you will need to write a function that takes in an input file name and an output file name, then calls the function you just wrote *messlasMoralMachine* and **writes** to a csv file with the following columns:

original Row Number (from the input file), person (Driver or Pedestrian), who was saved (Driver or Pedestrian)

Make sure to include a header for rowNumber and Saved so we know what we are looking at!

For example, your output csv might look something like:

rowNumber,person,saved

0,Pedestrian,Driver

1,Driver,Driver

2,Pedestrian,Pedestrian

3,Driver,Pedestrian

4,Pedestrian,Pedestrian

5,Driver,Pedestrian

6,Pedestrian,Driver

7,Driver,Driver

etc...

Amazing work on our Trolley Problem dilemma! On behalf of everyone at Messla, we are thrilled to have you onboard and we can't wait to code more moral machines with you!

**Problem 3 (10 points) Reflection**

Now that you have officially written your own “moral machine” algorithm, take a few moments to reflect on what the process was like. How did it feel to be in charge of the design decisions and criteria to select who to save? What do you think you could have done differently or better? What would you have done if you were working at a real self-driving car company and were assigned to create this algorithm?