

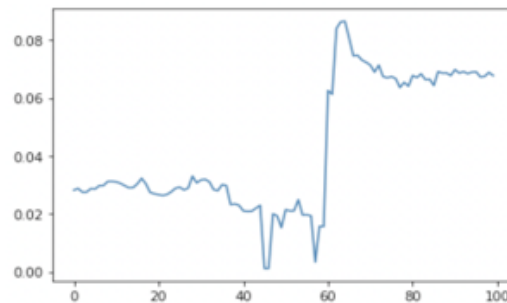
## Journal 25

Throughout this past week, I primarily focused on incorporating the back-end of my Machine Learning (ML) model with the ML portion of my GUI. As a refresher, the ML portion of my GUI consists of everything in the image below:

### Machine Learning Application

Enter Video Number Here

Output Display



Run ML Algorithm

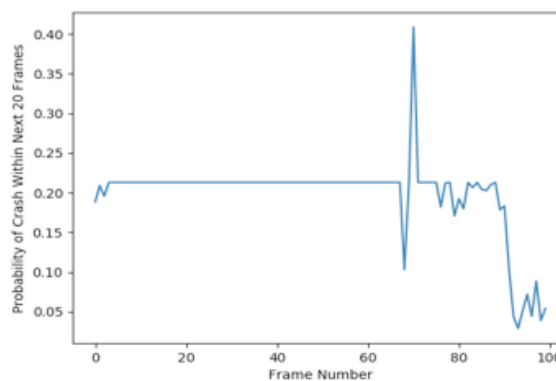
My progress consists of allowing the user to enter a video number in the text box and then click the “Run ML Algorithm” button. After the button is clicked, an output graph representing the predicted “Probability vs. Frame Number” is displayed below the “Output Display” label. The image below shows these added features in action:

### Machine Learning Application

Enter Video Number Here.  
Positive (456-620); Negative (830-1130).

473

Output Display



Run ML Algorithm

I will improve this interface for the user by displaying an (optional) preview of the test video and an image of the first frame in the video underneath the output graph.

In order to allow the user to actually use the ML model, I simply created a pipeline that would extract all of the frames from the video that the user chose and run my saved “.h5” weights file on those frames. The output of the ML model is a list of probabilities that provide information regarding the likelihood that there will be a vehicle crash within the next 20 frames of the current frame. The length of this list of probabilities is the number of frames in the video. Because I have set up the framework for implementing ML models into my GUI, I will dedicate the next week to refining my algorithm using more training videos and by restructuring my input data.