UNIVERSITY OF MICHIGAN-DEARBORN

**CIS-579**
**Winter 2023**
**Prof Khalid Khattan**

# Steering maneuver recognition


Presented By:

**Axel Aguilar**
**Sachin Lade**
**Harshit Barde**

# Motivation/Introduction

- Steering maneuver recognition can be used to determine the current driving status.
- Steering maneuver recognition can be used to take decisions or alert the driver to prevent accidents.
- Steering maneuver recognition can be used to improve driving experiences overall..
- Here we are using Supervised learning.


# Real Data or Simulation ?

- Equipment is expensive.
- Equipment requires license.
- Knowledge is required.

# Parameters [Update]

Parameters for the model in our Experiment will be

- Yaw Rate
-Steering angle
-Speed

We are excluding Time as a Parameter, we will use deltas instead

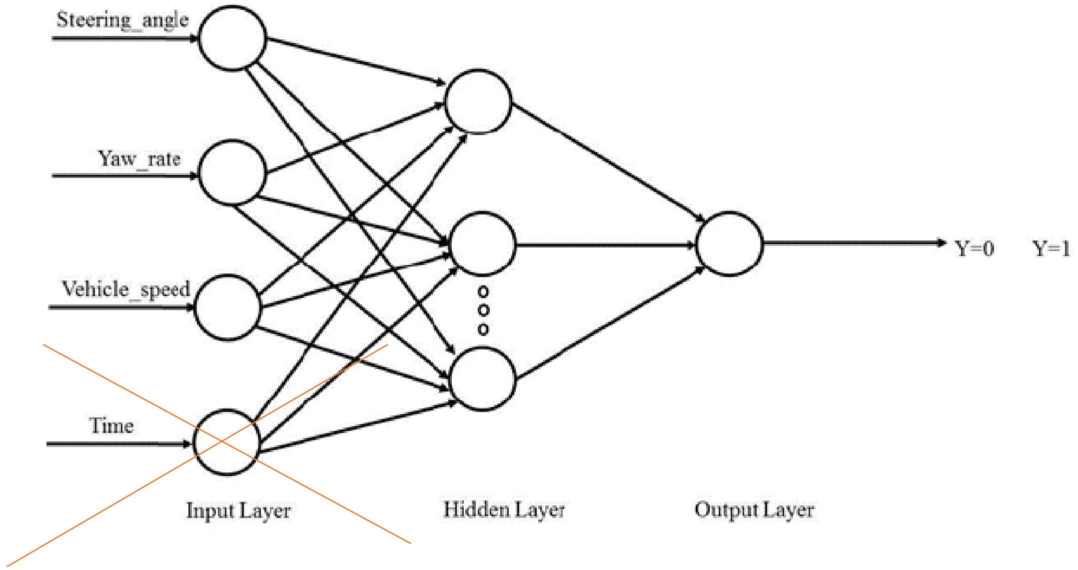
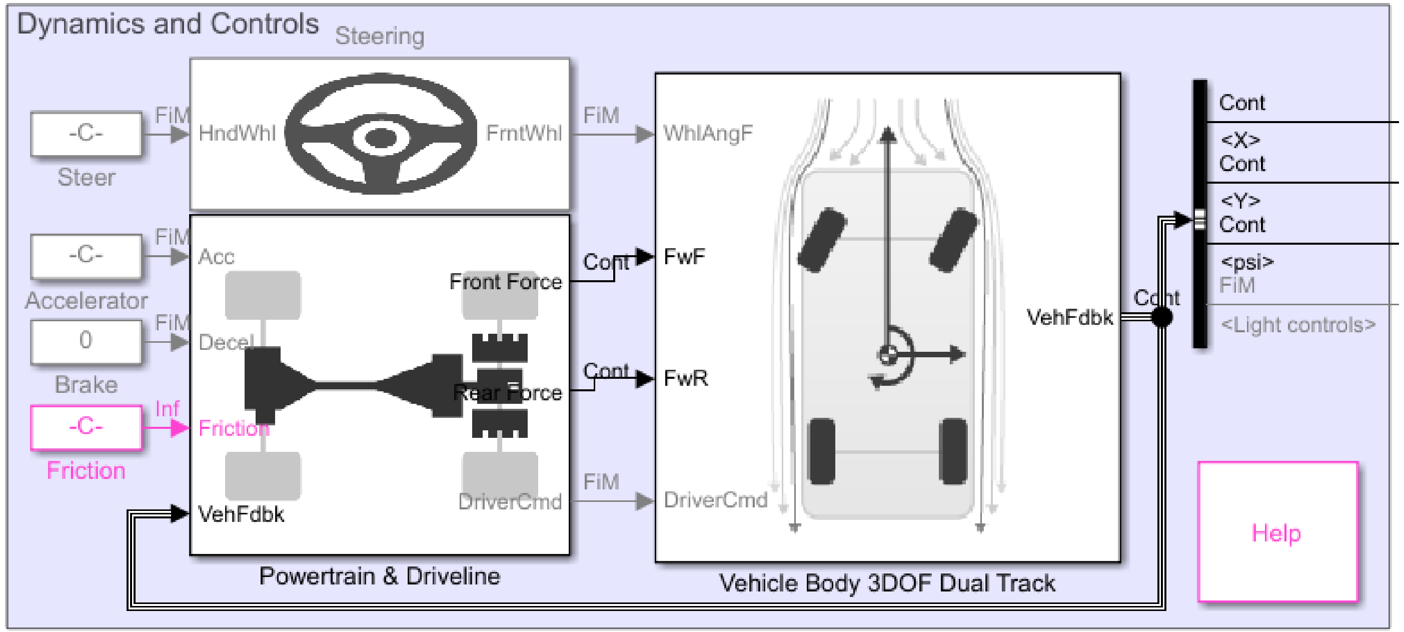



Figure 4: Artificial Neural Network architecture

# Simulation [MATLAB]

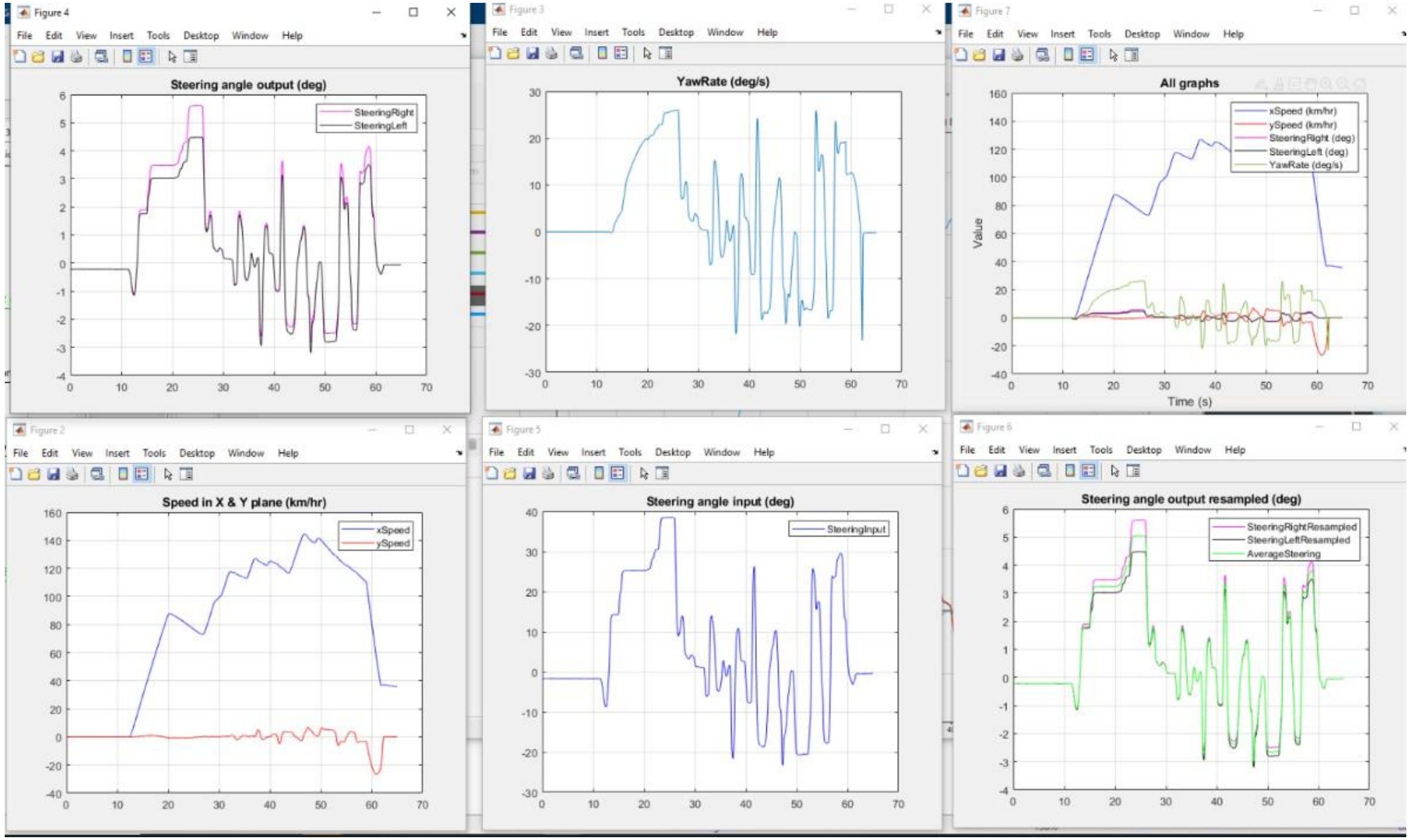
- **To acquire data equipment is needed to measure the parameters of interest, either using car sensors or external standalone sensors that also has logging capabilities.**

- Such equipment is usually expensive and requires license as well as knowledge on how to get the data



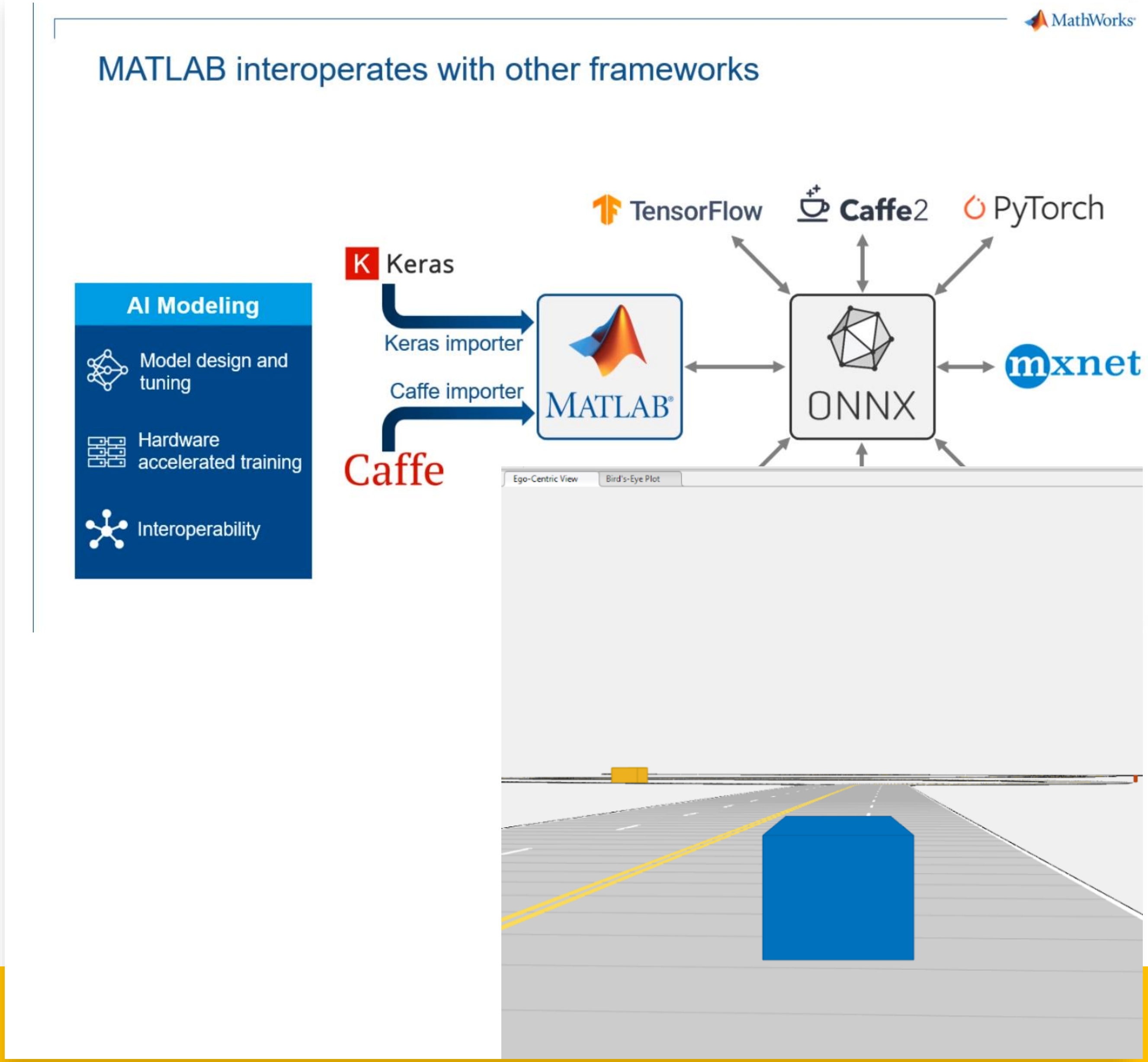
# Data Set Collection

- Time series data is a set of data points that are ordered by time.
- Time series data can be collected from a variety of sources, such as sensors, databases, and web services.
- MATLAB provides a variety of tools for importing, preprocessing, visualizing, and analyzing time series data.



# Framework

•MATLAB provides a comprehensive framework to integrate simulation and training

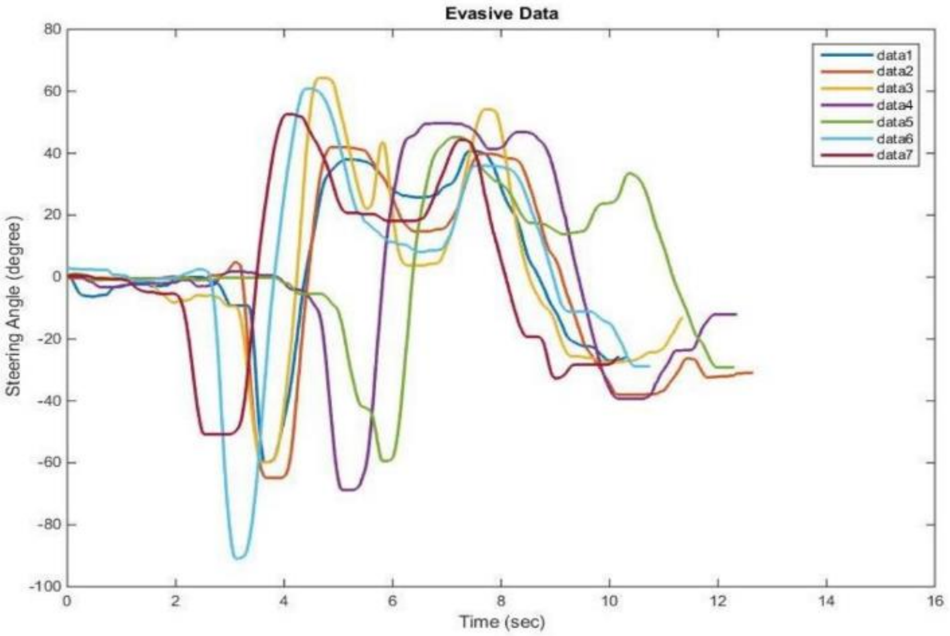


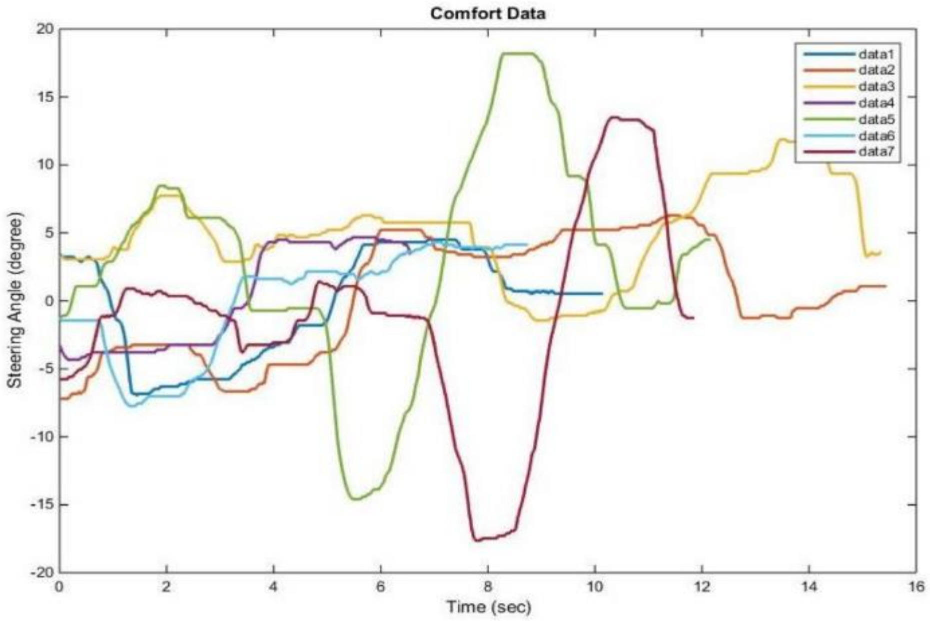
# Algorithm

- ANNs are good at recognizing patterns.
- ANNs can be trained on a variety of data.
- ANNs can be used to make predictions

- Explain in detail model from code

- Developed Evasive driving label logic based on the Steering Angle.



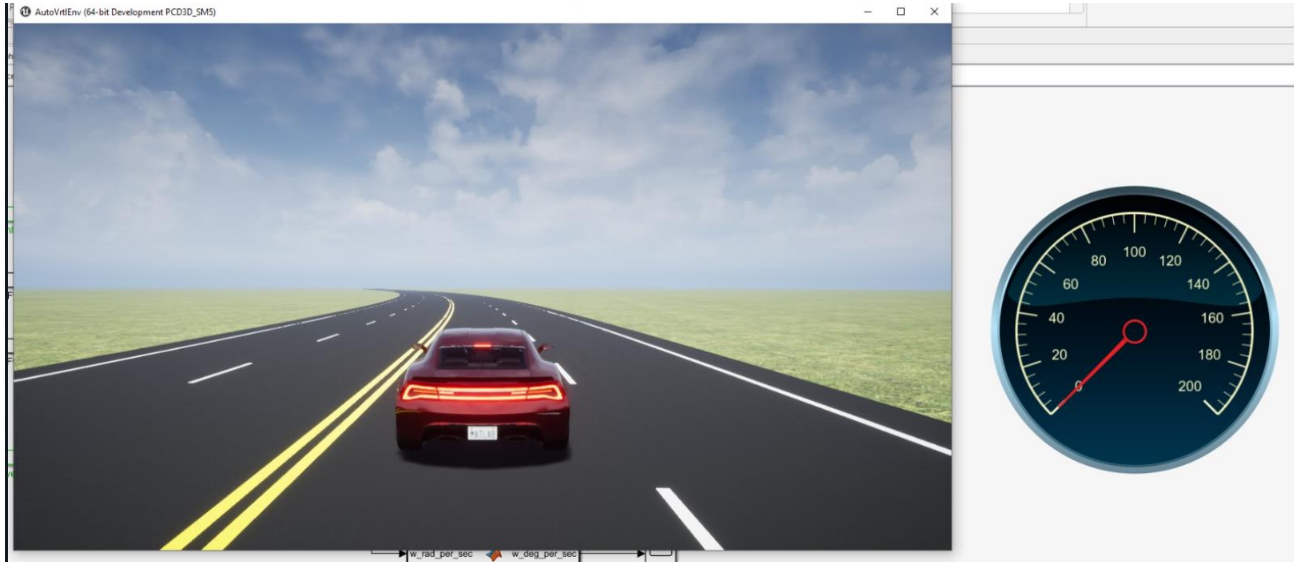
# Algorithm [Update]

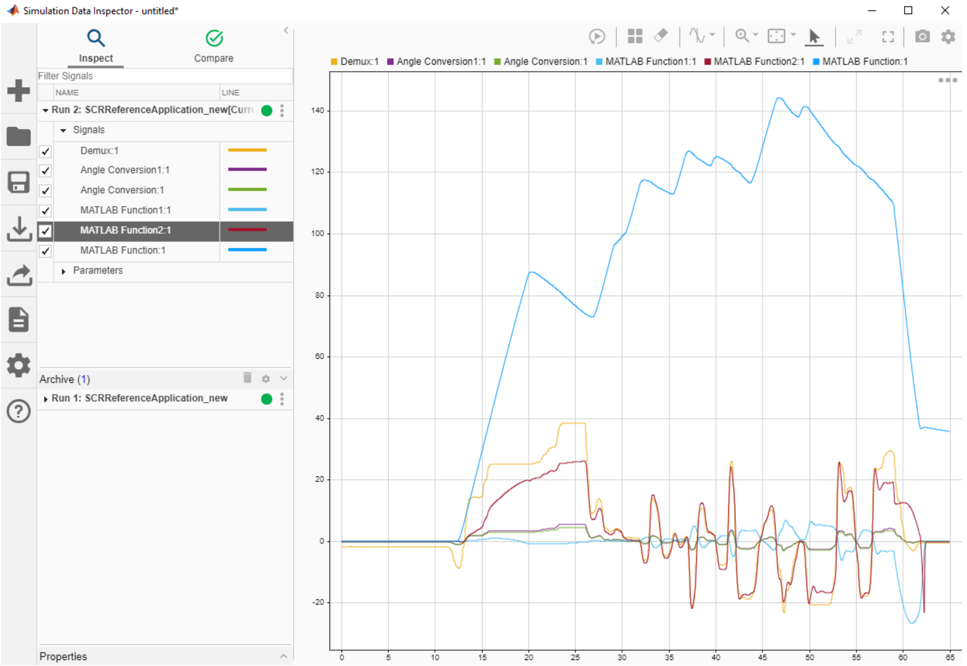
```
model = Sequential()
model.add(Dense(64, input_shape=(3,), activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(2, activation='softmax'))
```

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_45 (Dense)            (None, 64)                256

 dense_46 (Dense)            (None, 32)                2080

 dense_47 (Dense)            (None, 2)                 66

=================================================================
Total params: 2,402
Trainable params: 2,402
Non-trainable params: 0
```

# Experiment Result:

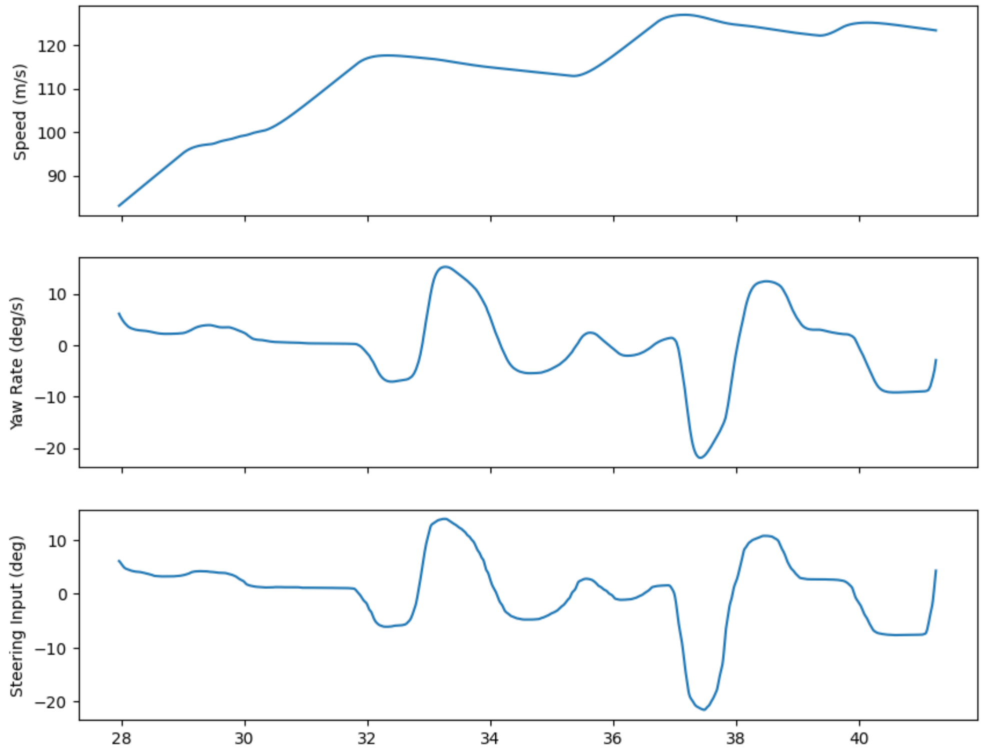
- Understanding of the predictive model through code.
- Steering angle for deriving Evasive or not.
- Simulations using MATLAB



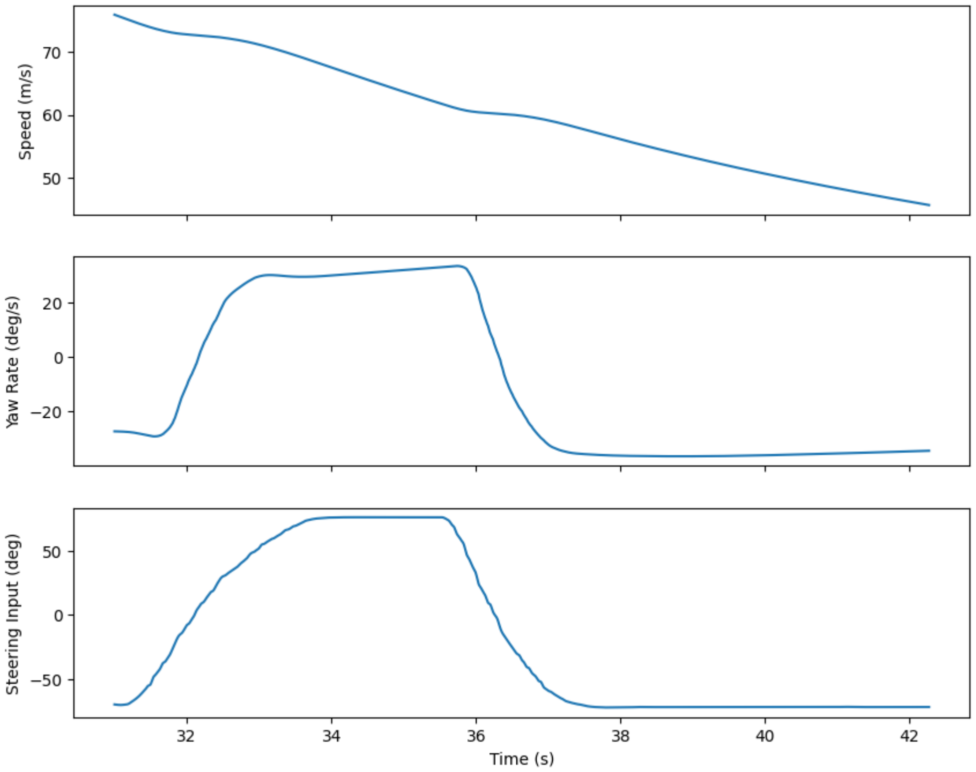



# Experiment Result [Update]

- The Training and Testing datasets were passed to train and test the model in the Google Collab project

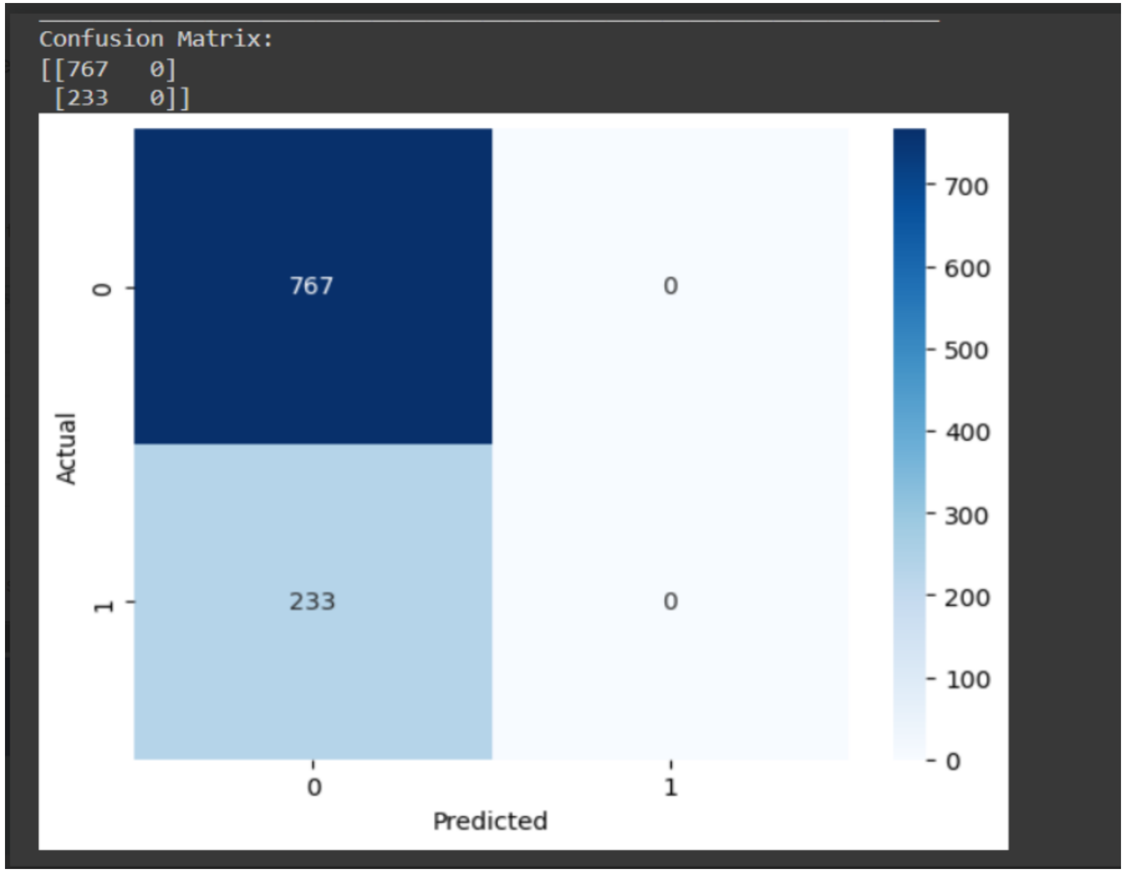


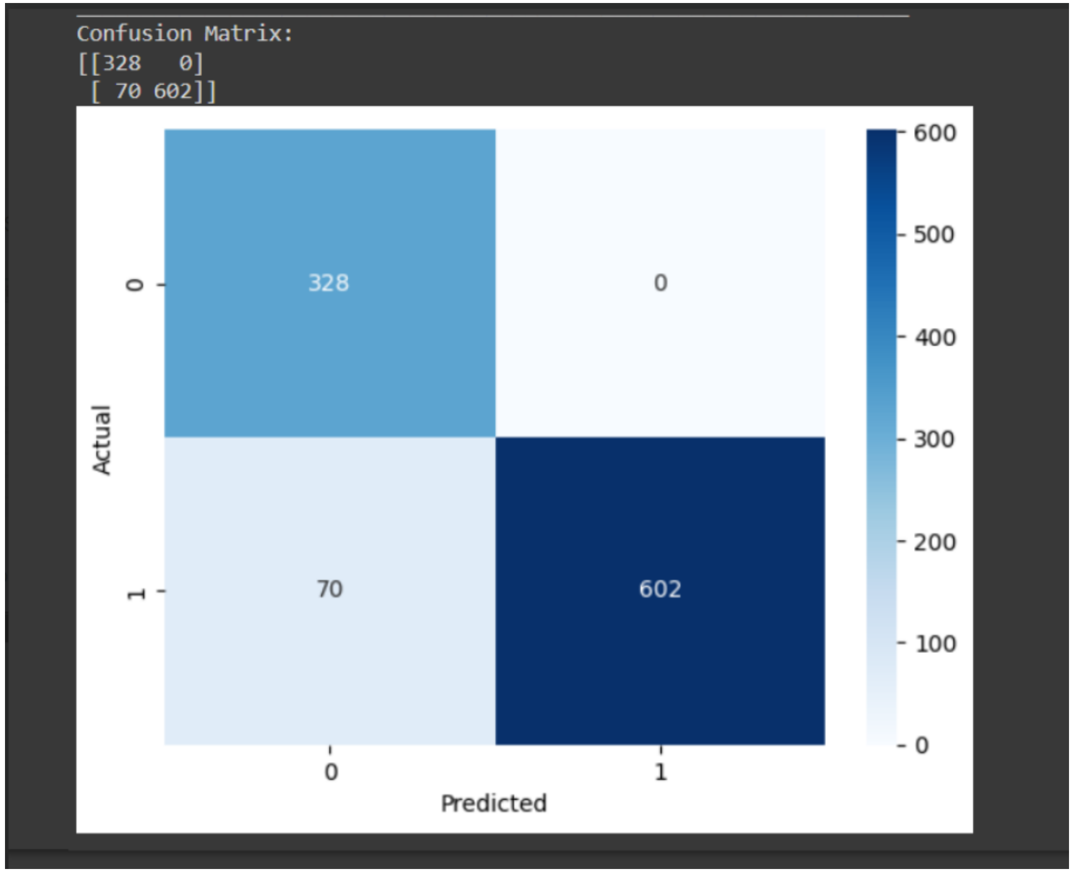

Train set




TEST Set

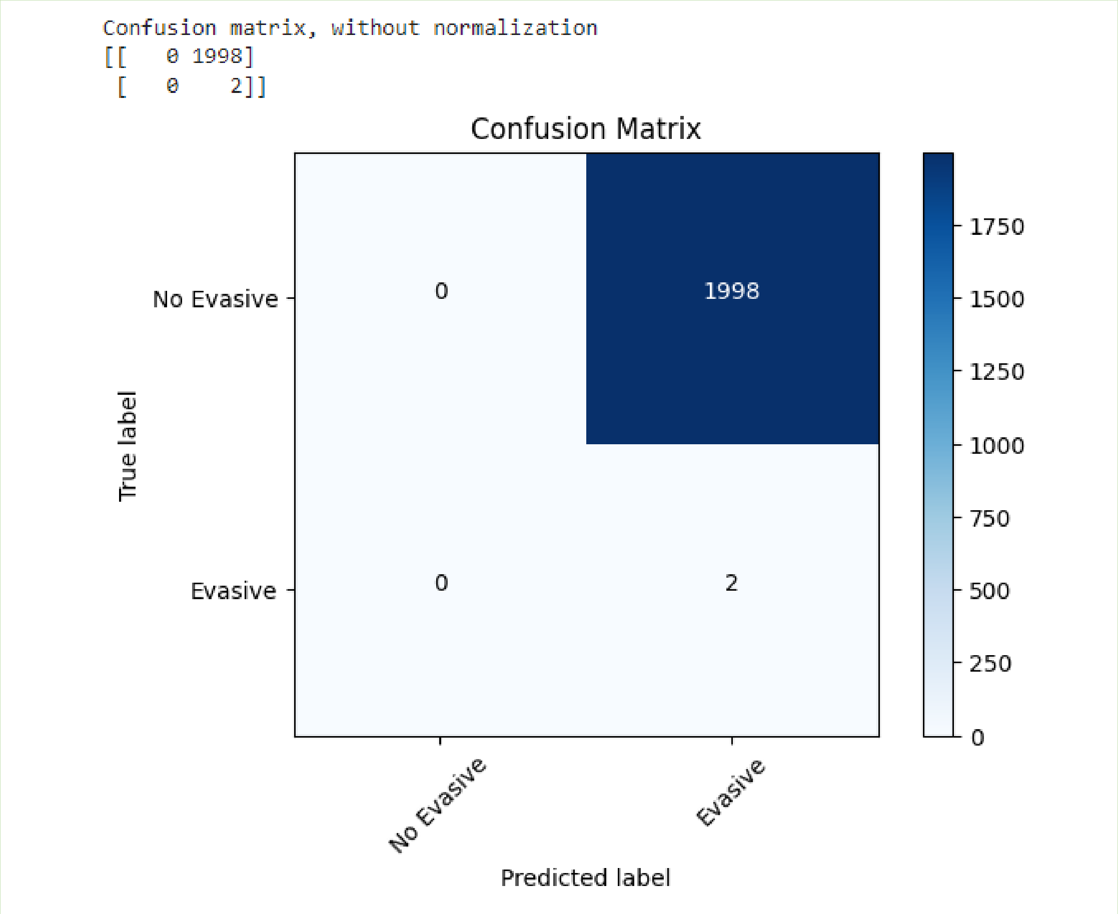
# Comparison of few Confusion Matrix



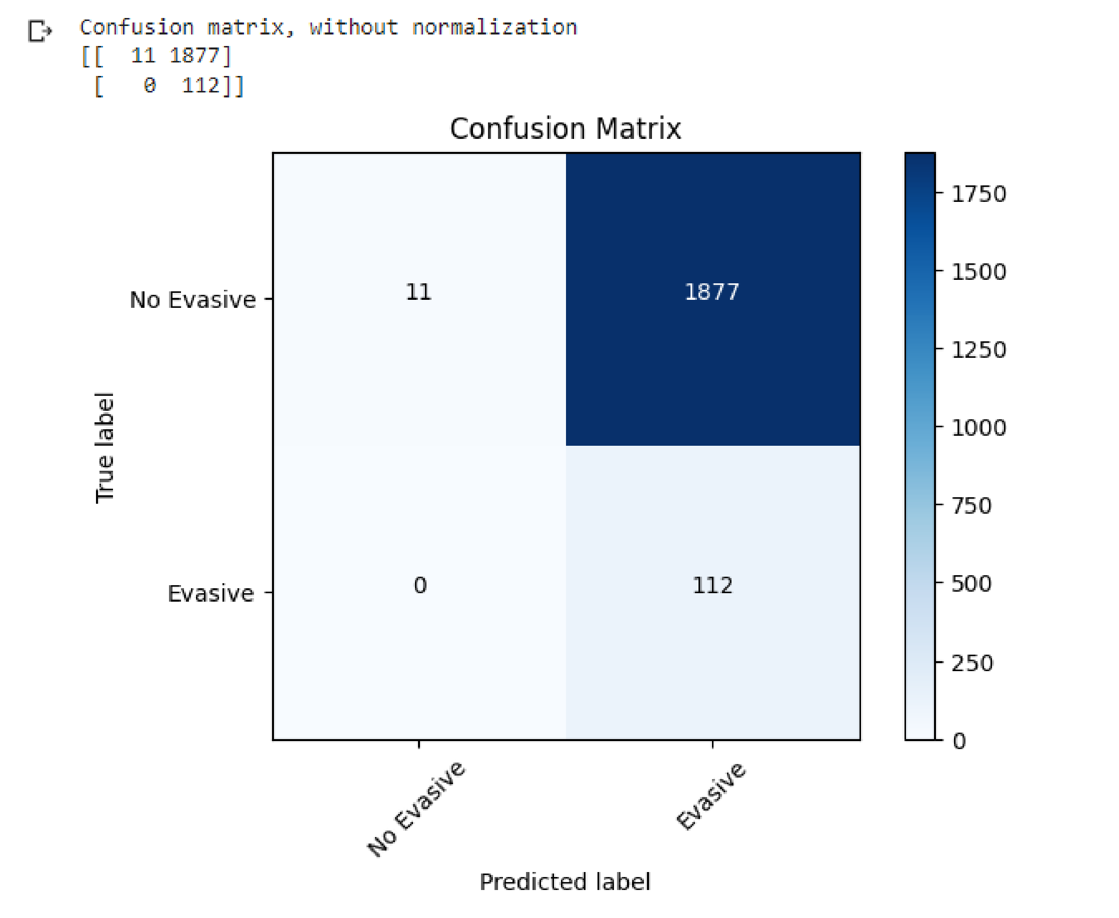
# Results 1 [Update]

```
if delta[i][2] > 0.25:                # Steering input delta greater than
     if delta[i][1] > 0.01:              # Yaw Rate greater than
       if delta[i][0] > 0.01:            # Speed delta greater than
              EVASIVE = TRUE
```



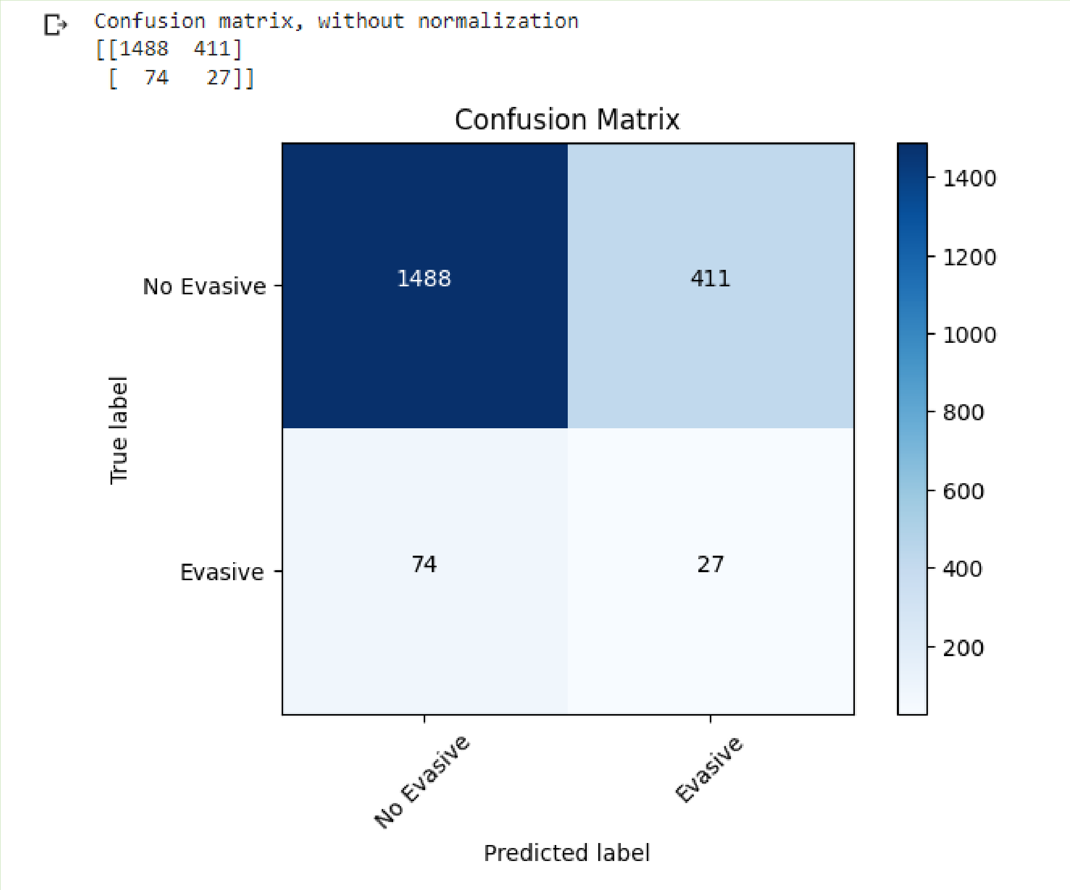
# Results 2 [Update]

```
if delta[i][2] > 0.01:                 # Steering input delta greater than
     if delta[i][1] > 0.001:               # Yaw Rate greater than
       if delta[i][0] > 0.001:             # Speed delta greater than
              EVASIVE = TRUE
```



# Results 3 [Update]

```
if delta[i][2] > 0.01:                  # Steering input delta greater than
    if delta[i][1] > 0.01:                  # Yaw Rate greater than
      if delta[i][0] > 0.001:                 # Speed delta greater than
              EVASIVE = TRUE
```

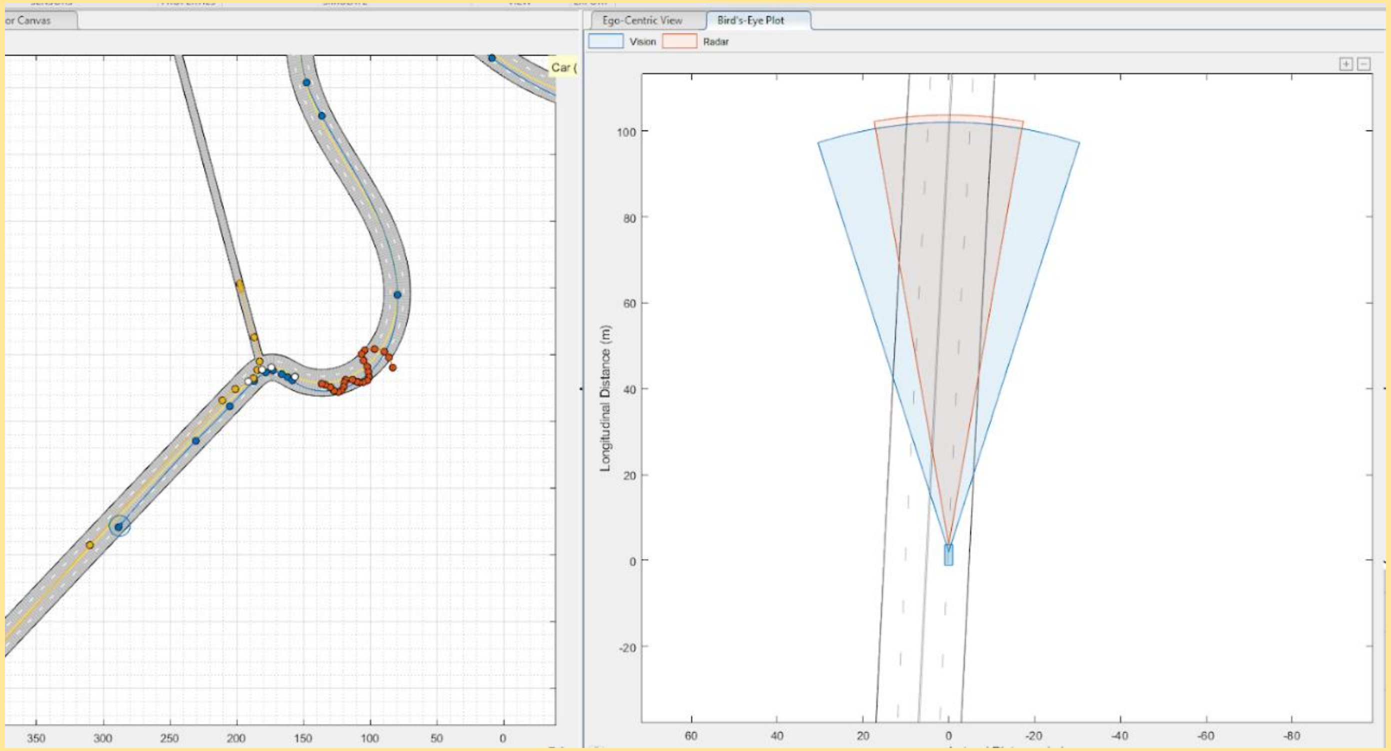


# Conclusions:

- **Defining Labels on Supervised Learning is KEY factor, finding parameters to tune the training is critical**
- The ANN model was able to recognize lane change behavior with a high degree of accuracy, both for trained and untrained data.
- The model was able to classify lane change behavior into two states: normal and evasive.
- The authors suggest that future work could focus on improving the accuracy of the model by considering additional potential inputs, such as the behavior of the driver's visual system.
- Future Scope an aggressive lane change or maneuver can be predicted through our model using more inputs from sensory data

# Future plans:

- Calculate and use Delta values to calculate differences in between actual and previous value
- Define what are the rules and regulations for evasive and non-evasive
- Create different instances of the ANN model optimizing which parameters will be used
- Add sensors to get feedback from the road to increase the accuracy of the predictions



# Sources:

**Lane Change Behaviour Recognition Using Neural Network**
*N. J. Zakaria1,2, H. Zamzuri1,2,*, M. H. Mohamed Ariff1, M. Z. Azmi2,*
*N. Hassan2*

**Unobtrusive drowsiness detection by neural network**
**learning of driver steering**
**R Sayed** and **A Eskandarian***
Center for Intelligent System Research, The George Washington Transportation Research Institute, Ashburn, Virginia, USA

***Actual Project in Google Collab:***
*https://colab.research.google.com/drive/1rekKKv2kxPV2GSYd_ViQ9ut8QnEkaFHr#scrollTo=-cQAqeOjBccb*

*Vehicle Dynamics model*
https://www.mathworks.com/help/vdynblks/ug/scene-interrogation.html

# Thank You