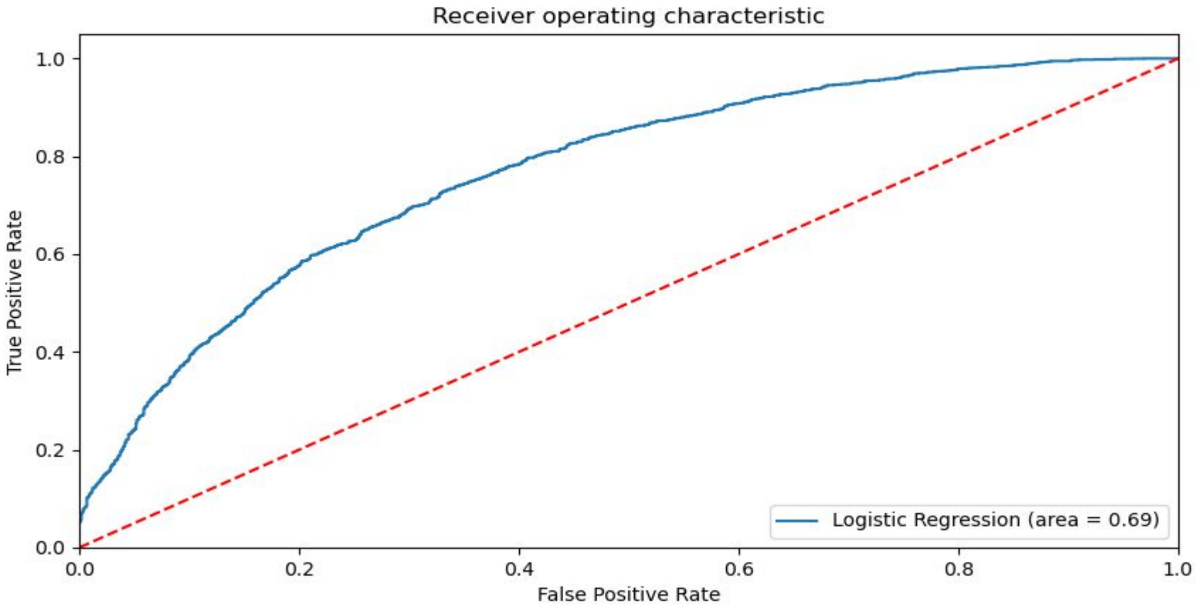
Final Presentation of Mouse

Zening Ye, Qiannan Shen

Recall

- Randomly Select Mouse # 608102_414 from Zero Maze experiment
- Using logistic regression as the baseline model
- Conducted a Recurrent Neural Network(GRU)
- All 70% training, 20% validation, 10% testing

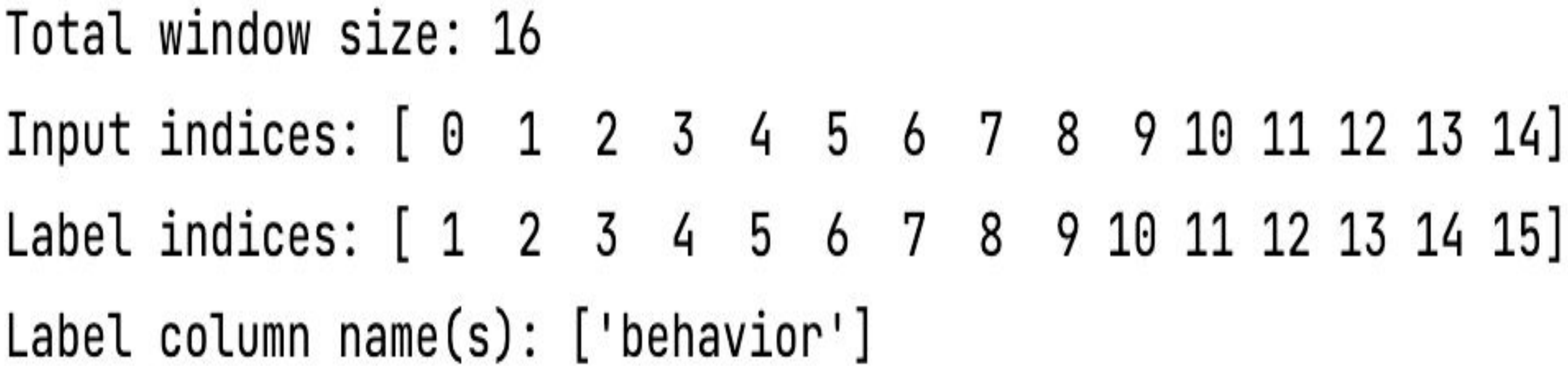
Baseline Model (Logistic)



```
Accuracy score: 0.711677
Confusion Matrix :
 [[1965  450]
 [ 703  881]]
```

Window Generator

- Single Step Window
- 15 inputs, step equal 1

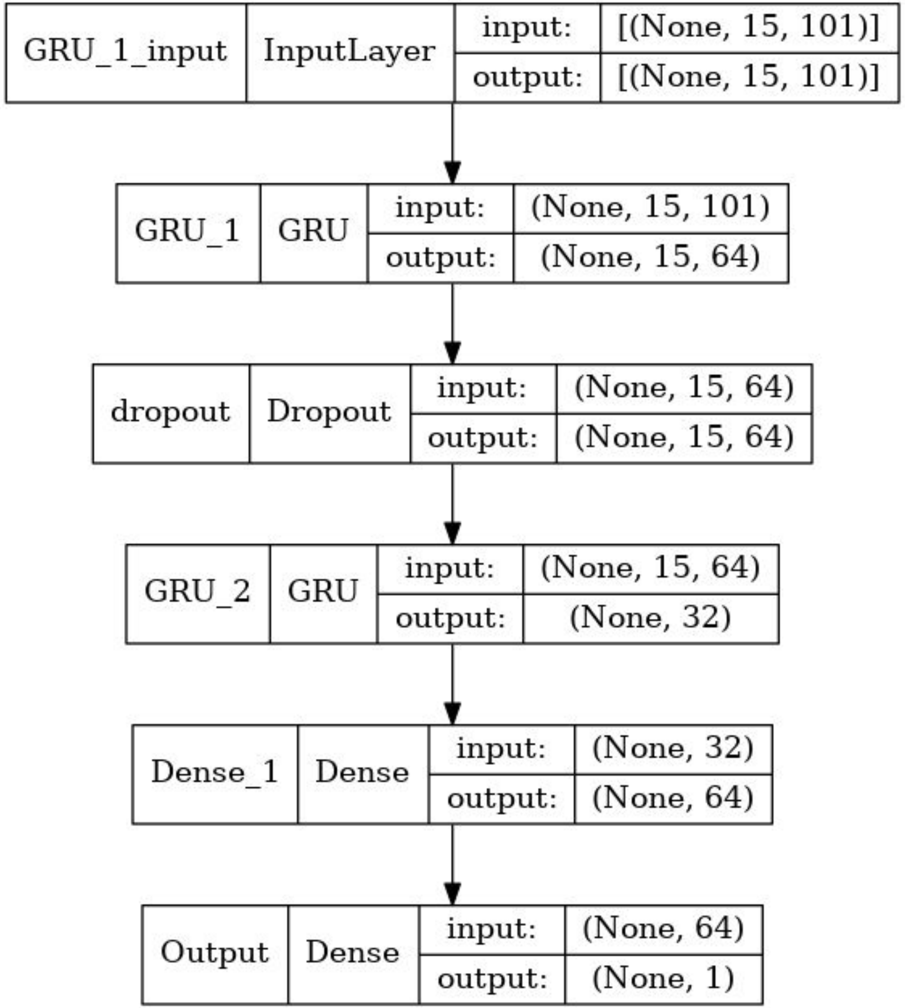


```
Total window size: 16
Input indices: [ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14]
Label indices: [ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15]
Label column name(s): ['behavior']
```

Recurrent Neural Network

Single Step RNN

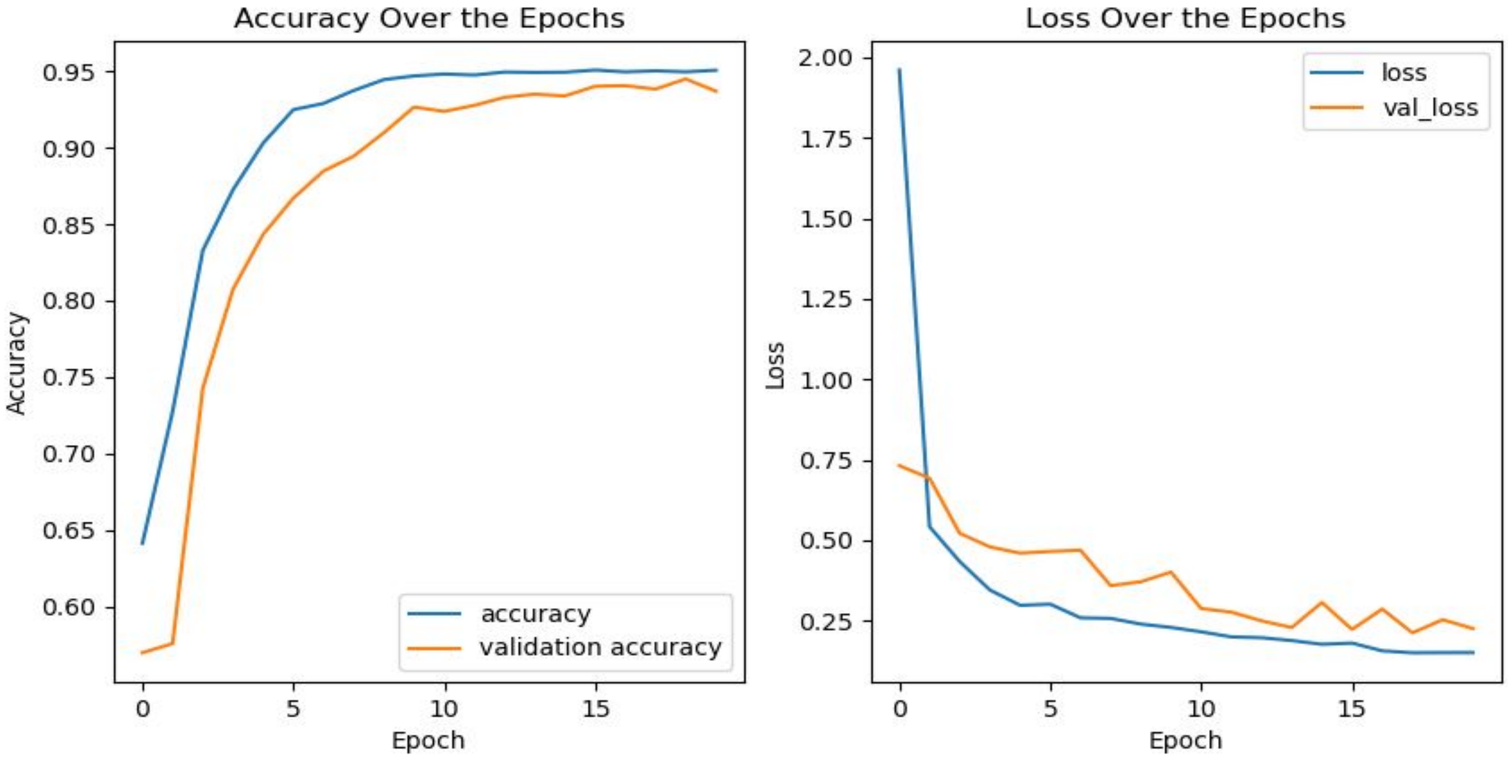
- 2 GRU layers with ReLU activation function
- 1 fully connected layer with ReLU activation function
- 1 Dropout layer with level 0.1
- 1 fully connected output layer


GRU_1_input | InputLayer | input: [(None, 15, 101)] | output: [(None, 15, 101)]
GRU_1 | GRU | input: (None, 15, 101) | output: (None, 15, 64)
dropout | Dropout | input: (None, 15, 64) | output: (None, 15, 64)
GRU_2 | GRU | input: (None, 15, 64) | output: (None, 32)
Dense_1 | Dense | input: (None, 32) | output: (None, 64)
Output | Dense | input: (None, 64) | output: (None, 1)

Hyperparameters and Loss/Optimizer

- Learning rate: 0.0001
- Epochs: 20
- batch_size : 11
- Loss: Binary Cross Entropy
- Optimizer: RMSprop(Root Mean Squared Propagation)

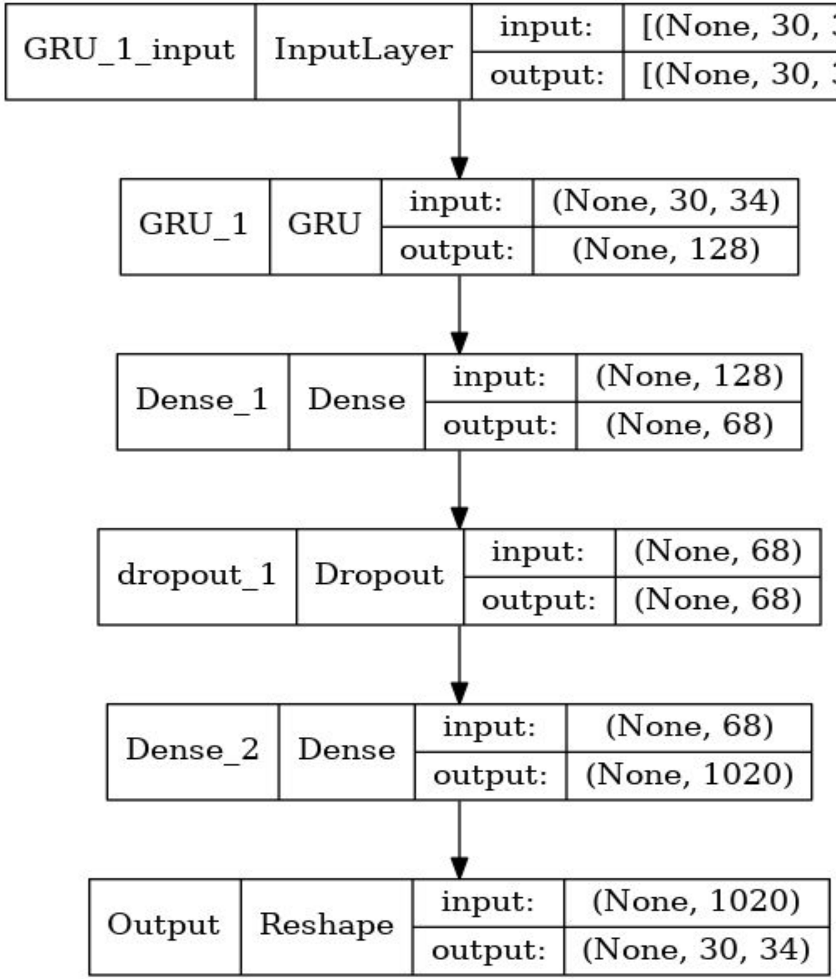
Accuracy and Loss



Another Thought

Multi-step RNN

- 20 Inputs, step equal 20
- 1 GRU layer with ReLU activation function
- 1 fully connected layer with ReLU activation function
- 1 fully connected layer with Sigmoid activation function
- 1 fully connected output layer


GRU_1_input
InputLayer
input:
[(None, 30, 34)]
output:
[(None, 30, 34)]
GRU_1
GRU
input:
(None, 30, 34)
output:
(None, 128)
Dense_1
Dense
input:
(None, 128)
output:
(None, 68)
dropout_1
Dropout
input:
(None, 68)
output:
(None, 68)
Dense_2
Dense
input:
(None, 68)
output:
(None, 1020)
Output
Reshape
input:
(None, 1020)
output:
(None, 30, 34)

Result

- Using single-step RNN came up a well shape prediction for the behavior
- Single Step Test Loss: 0.0854, Accuracy: 0.9575
- Unable to predict the behavior using multi-step at this moment

Improvement & Suggestion

- Using multiple samples of data to train the network

- Using Autoregression for multi-step prediction