***Prediction Model:***

***Libraries Used: Pandas, Numpy, Seaborn, scikit-learn, Tensorflow 2, Keras***

**I opted for a 6 layer Artificial Neural Network for this model.**

***Why a Neural Network?***

-Regression Prediction models aren't that efficient as compared to these ANNs.

-Efficiency can be tweaked and can be increased with these types of Neural Networks as shown in this model.

After tweaking with a few types of Neural Networks, I found out that a 6 layer, 21 Neurons Network output is the most efficient with about 96% - 97% efficiency.

**Activation used: Rectified Linear Unit**

**All Observations:**

1st Observation: 3 layer, 9 Neurons model - 91% efficiency.

2nd Observation: 4 layer, 13 Neurons model - 93% efficiency.

3rd Observation: 5 layer, 17 Neurons model - 95% efficiency.

4th Observation: 6 layer, 21 Neurons Model - 96-97% efficiency. (Selected this model)

5th Observation: 7 layer, 25 Neurons Model - 93-95% efficiency with increased losses.

**The neural network:**



Please check this google colab file for step wise explanation of the code: <https://bit.ly/3z88cnQ>. I have also calculated losses along with efficiency in the file.

To run the code please upload a csv file to the colab env and rename the following csv file name to the uploaded file name.



To predict a value please change the RPM and TPS values at the end of the colab file.



The predicted value is shown as an array:



The file generates a .h5 file which can be further used for predictions.

**The folder:**



Prediction model.py is the main file with the code.

HiPER.h5 is the generated prediction model saved by the prediction\_model.py file.

To reuse the saved model, the following code should be used.

