**PROJECT REPORT**

Our task was to classify our data into either of Musk(1) or non-Musk(0). For this task we had dataset with 166 numerical features.

My approach for the solution was done is following steps:

* Remove 3 unnecessary features like molecule name,ID,conformation name and seprate our class feature from rest of the data
* Divide the dataset into Train and validation set with ration 80:20
* For preprocessing, Normalization is done to make all the values lie between 0 and 1.Although normalization is not actually neceassy in Neural network but it's better to normalize value because it gives better result. Why? because when input value is multiplied by weights the it gives smaller value which result in better gradient calculation.
* Model architecture is simple, it has 1 input layer,2 dense layer with 2048,512 units. Why larger unit is used before and smaller later? Because i allowed the model to learn larger range of features at first dense layer then second dense layer having smaller units will force the model to learn only important features. Dropout layer is added as regularizer incase model starts to overfit and finally output layer with 1 unit for giving out class probability between 0-1 due to sigmoid function.
* Loss function is Binary Crossentropy and optimizer is Adam with default learning rate (0.01). Model checkpoint is used to save best model i.e., Model with lowest validation loss.
* For evaluation, Confusion metrix,roc score,auc score,roc curve plotting is used apart from normal accuracy metric.
* Eval Result: Accuracy of model was around 96% while AUC score was 0.99 apart from that confusion matrix showed that model generatlization is very good.
* Other Matrics scores:

1. Precision Score: 0.984
2. Recall Score: 0.952
3. F1 Score : 0.968

The model was trained for 30 epochs and here are the final results:

