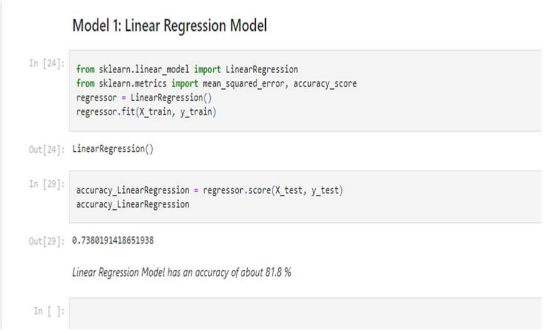
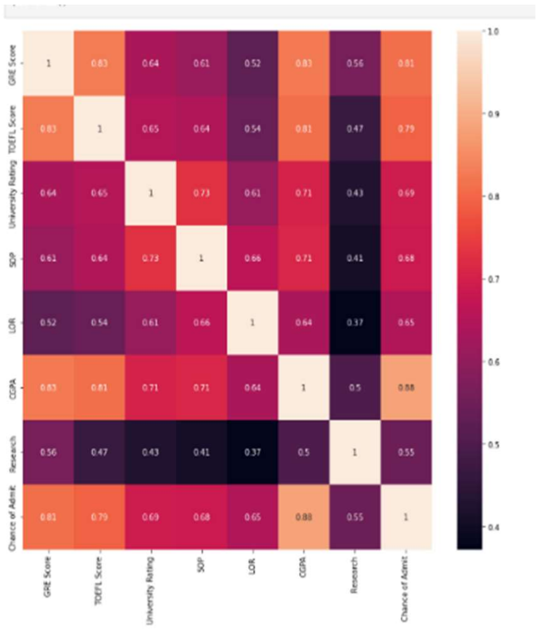
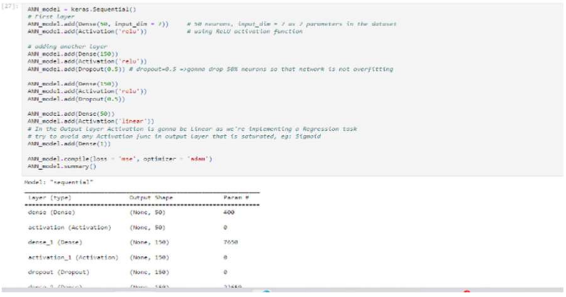
Project Development Phase
Model Performance Test

Date	03 November 2022
Team ID	PNT2022TMD48455
Project Name	University Admit Eligibility Predictor
Maximum Marks	10 Marks

S.No.	Parameter	Values	Screenshot
1.	Metrics	Linear Regression Model *Linear Regression Model has an accuracy of about 81.8 %*	 
2.	Tune the Model	Artificial Neural Network (ANN) Model	

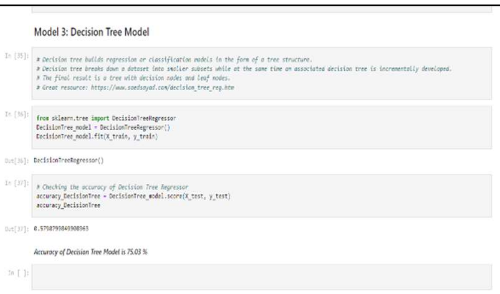
Model 1: Linear Regression Model

```
In [24]: from sklearn.linear_model import LinearRegression
         from sklearn.metrics import mean_squared_error, accuracy_score
         regressor = LinearRegression()
         regressor.fit(X_train, y_train)

Out[24]: LinearRegression()

In [29]: accuracy_LinearRegression = regressor.score(X_test, y_test)
         accuracy_LinearRegression

Out[29]: 0.7380191418651938
```

Linear Regression Model has an accuracy of about 81.8 %

```
In [ ]:
```

```
[27]: ANN_model = keras.Sequential()
      # First Layer
      ANN_model.add(Dense(50, input_dim = 7))     # 50 neurons, input_dim = 7 as 7 parameters in the dataset
      ANN_model.add(Activation('relu'))           # using ReLU activation function

      # adding another layer
      ANN_model.add(Dense(150))
      ANN_model.add(Activation('relu'))
      ANN_model.add(Dropout(0.5)) # dropout=0.5 =>gonna drop 50% neurons so that network is not overfitting

      ANN_model.add(Dense(150))
      ANN_model.add(Activation('relu'))
      ANN_model.add(Dropout(0.5))

      ANN_model.add(Dense(50))
      ANN_model.add(Activation('linear'))
      # In the Output Layer Activation is gonna be Linear as we're implementing a Regression task
      # try to avoid any Activation func in output layer that is saturated, eg: Sigmoid
      ANN_model.add(Dense(1))

      ANN_model.compile(loss = 'mse', optimizer = 'adam')
      ANN_model.summary()

Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 50)                400

activation (Activation)      (None, 50)                0

dense_1 (Dense)              (None, 150)               7650

activation_1 (Activation)    (None, 150)               0

dropout (Dropout)            (None, 150)               0
```

3.	Tune the Model	Decision Tree Model	

Model 3: Decision Tree Model

```
In [35]: # Decision tree builds regression or classification models in the form of a tree structure.
         # Decision tree breaks down a dataset into smaller subsets while at the same time an associated decision tree is incrementally developed.
         # The final result is a tree with decision nodes and leaf nodes.
         # Great resource: https://www.saedsayad.com/decision_tree_reg.htm

In [36]: from sklearn.tree import DecisionTreeRegressor
         DecisionTree_model = DecisionTreeRegressor()
         DecisionTree_model.fit(X_train, y_train)

Out[36]: DecisionTreeRegressor()

In [37]: # Checking the accuracy of Decision Tree Regressor
         accuracy_DecisionTree = DecisionTree_model.score(X_test, y_test)
         accuracy_DecisionTree

Out[37]: 0.5758793849908963
```

Accuracy of Decision Tree Model is 75.03 %

```
In [ ]:
```