Charity Funding Predictor

Objective :

The objective of the model is to predict the outcome of a charity funding application.

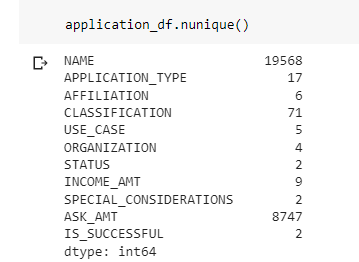
This will help enable charities submitting applications to predict their chances of getting the funding. That will help charities to temper their expectations and also modify their program to meet their objectives and the funding goals.

Overview:

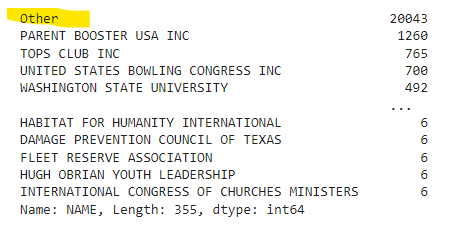
A neural network will be used to create a model using the tensor flow platform. The data from the csv file will be read and cleaned. It will be put through a scaling process and then finally sent to the tensorflow model. Various tuning methods will be used along the way such as dimensionality reduction, domain reduction, removing invalid values etc. The various tuning parameters such as number of layers, number of neurons and the various activation functions will be tried to optimize and tune the model

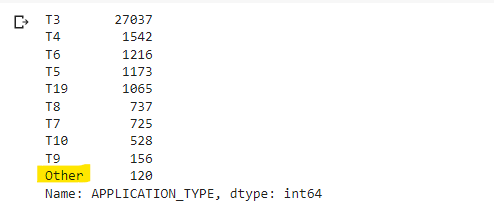
Results:

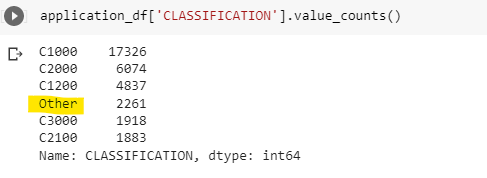
Dimensionality reduction is performed by removing the unneeded columns such as NAME and EIN



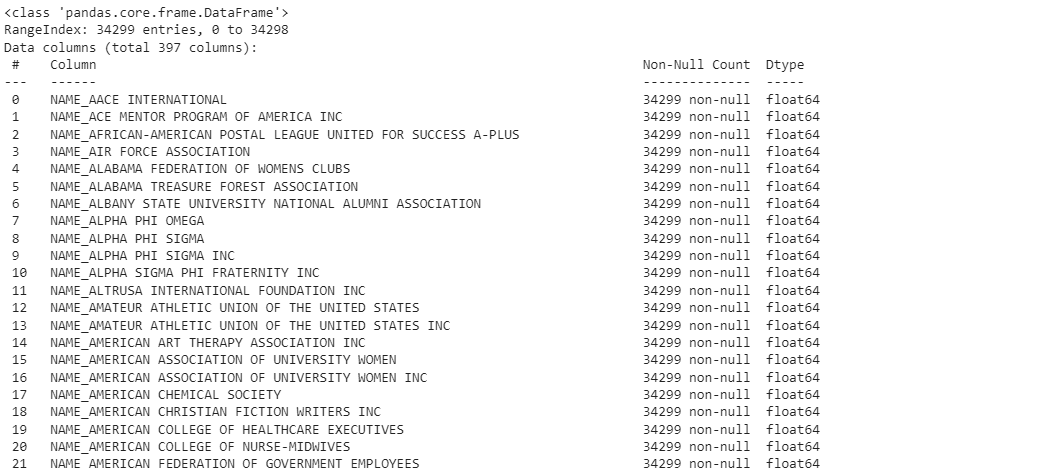
The domain size for various features us reduced by clubbing values with low counts to an “Others” bin



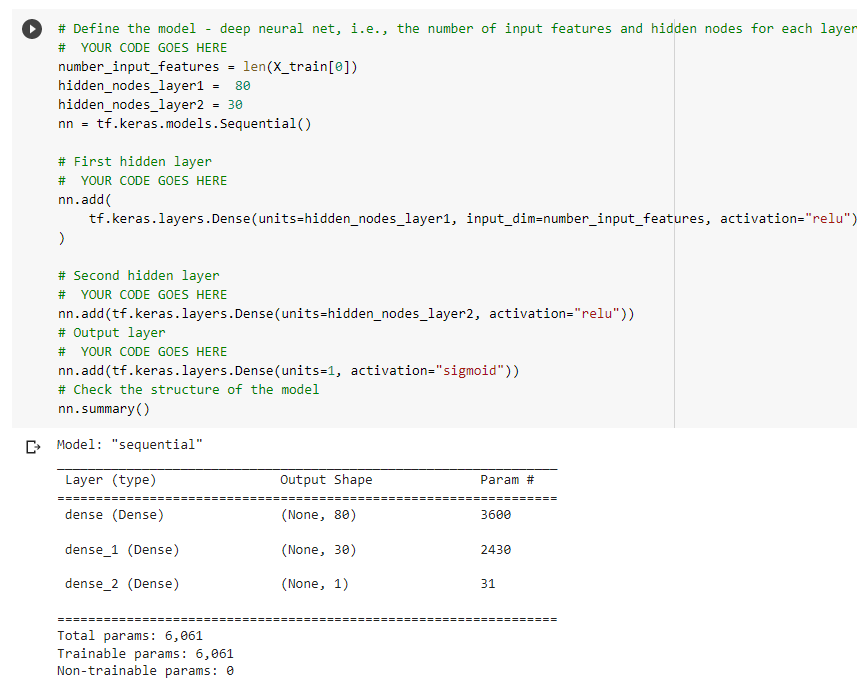




The categorical columns are then one-hot encoded

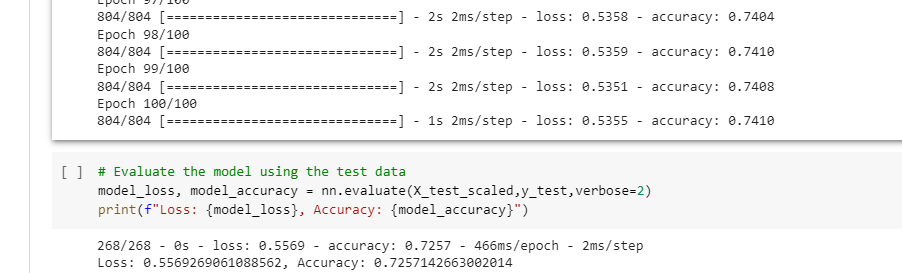


The features and the target are split and then scaled. The data is then split between training and test data and fed to the tensor flow model.

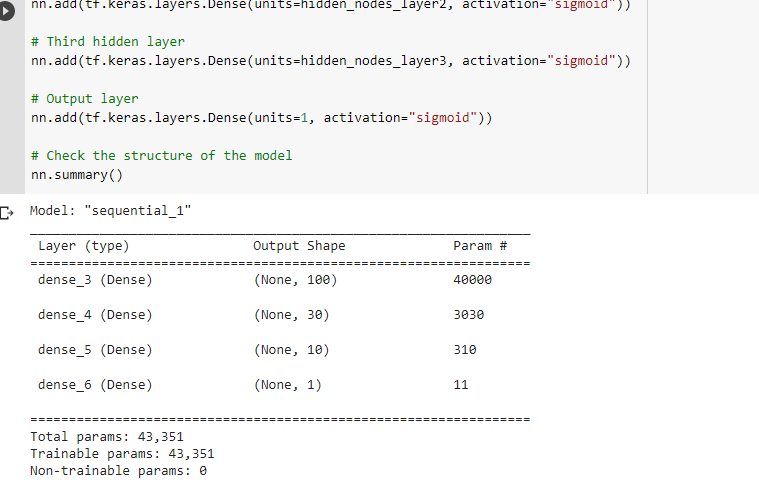


The first attempt at the neural network model gives a training accuracy of .74 but scores a bit lower on test at .72.

This attempt is made by eliminating the NAME and EIN columns.



The next attempt was made by adding the NAME gesture back but by binning it to reduce the domain values.



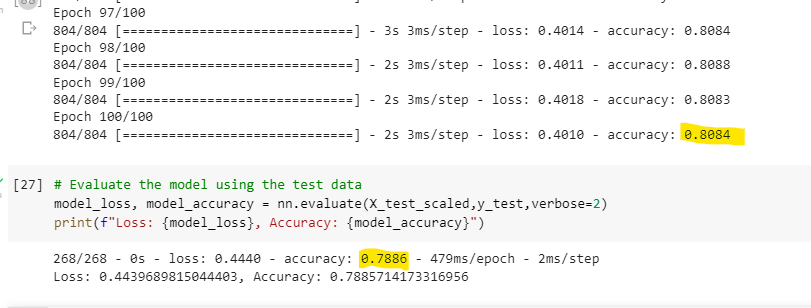
This model had a high number of input features due to the addition of the NAMES feature

Also a high number of neurons and multiple layers were used to improve the accuracy.

This model scores high on the training set at .8084.

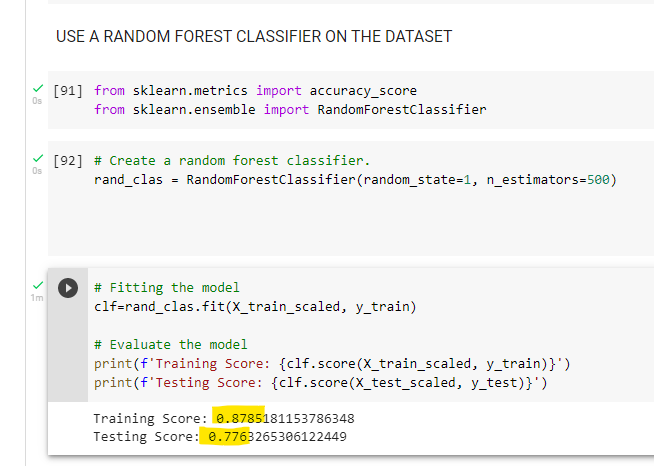
However the test accuracy is a bit lower at .788 pointing to some overfitting.

Also the execution time is increased and the resources used are higher

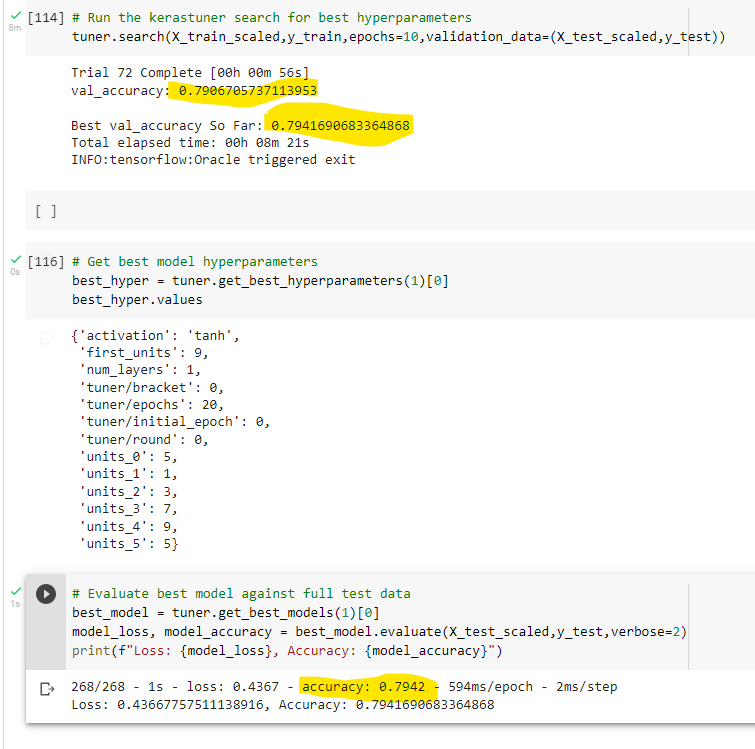


Another attempt to optimize the model using random forest was made

This attempt was not very successful . It overfits as can be seen with a high training score but a low test score of .77



A hypertuner was then used using the kerastuner library.



This provides a much improved training and test score.

The low spread between training and test show its reliability and the best model hyperparameters are shown.