**Overview**:

**Optimisation Methodology:**

1. The initial step was to find if any variables columns could be excluded from the model to improve accuracy.
   1. Drop columns from the application data frame.
   2. Run model using initial parameters *(See Appendix Figure 1)* to see if accuracy improved from initial test with all parameters included.
   3. Noted any parameters that increased accuracy of the model *(See appendix Figure 2)*.
2. Step two was implemented to find the ideal cutoff value for the application types and classification types. i.e. any classification types with a frequency that did not occur over a certain amount of times (threshold) was grouped together in the model and classified as ‘other’.
   1. Change the classification threshold for both application types and classification types and observe any increase in accuracy *(See appendix Figure 3)*.
3. Third the model was calibrated to optimal level from the two above steps then using experimenting with changing how many neurons were in each hidden layer as well as, the number of hidden layers and the activation function used in each hidden layer. The same number of Epochs was used during each experiment.
   1. Using a grid search method, we **Experiment with Different Activations** including combinations of activation functions like relu, tanh, and elu in various hidden layers to see which combination performs best (See Appendix Figure 4).
4. **Results**:

Data Preprocessing

* + The model target variable is the ‘IS\_SUCCESSFUL’ column. This variable indicates whether a charity donation request was successful (1) or not (0).
  + The features of the model are all columns from the data frame after dropping EIN', 'NAME', 'SPECIAL\_CONSIDERATIONS’ and IS\_SUCCESSFUL. This includes; Categorical variables converted to numeric through one-hot encoding and; Numerical variables (INCOME\_AMT, ASK\_AMT).
  + Model was found to increase accuracy when ‘SPECIAL\_CONSIDERATIONS’ column is dropped from the Data frame *(See Appendix Figure 2).*
  + It can be seen that the model accuracy is not impacted significantly by changing the classification bins (See Appendix Figure 3).
  + The greatest accuracy was found by introducing a application type cutoff threshold of 600 and a classification type cutoff threshold of 1000.

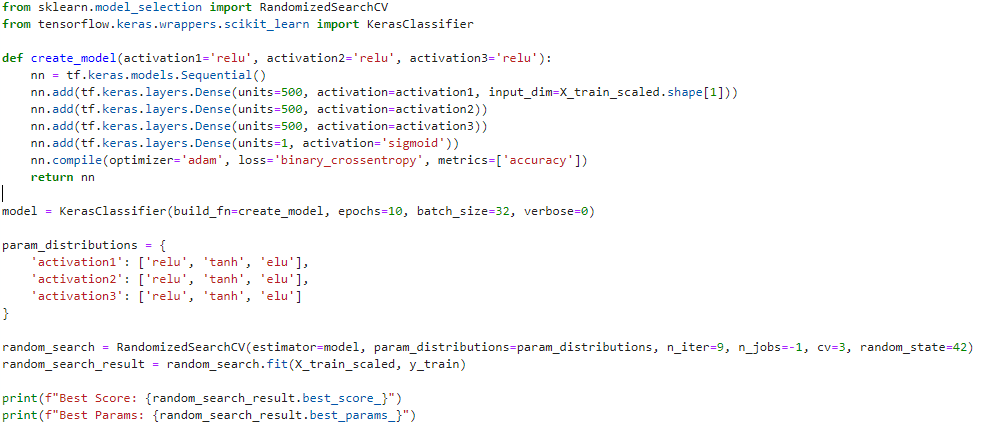
Compiling, Training, and Evaluating the Model

* + - How many neurons, layers, and activation functions did you select for your neural network model, and why?
  + By increasing the number of neurons in the first and second layer the accuracy was increased greatly.
  + By increasing the number of hidden layers, the accuracy of the model was also increased.
  + By increasing the number of epochs (iterations) the accuracy of the model was also increased.
  + **ReLU (Rectified Linear Unit)** was used as the activation function for each hidden layers as it helps the network learn complex patterns by introducing non-linearity, which is crucial for deep learning models. ReLU is computationally efficient and helps mitigate the vanishing gradient problem.
  + **Sigmoid** activation function is appropriate for the output layer in binary classification problems. It outputs a value between 0 and 1, which can be interpreted as a probability, making it suitable for binary classification.
  + Our final model optimisation was the following:

**Summary**:  
  
**BatchNormalization**: Normalizes the output of the previous layer.

**Dropout**: Randomly drops neurons during training to prevent overfitting.

**APPENDIX**

****

**Figure 4**