Date: July 6, 2025

From G. Pang , **File is BRCA1-Simulation3**

-------------------------------------------------------------------------------------------------------------------

**Dataset description:**

Training dataset: 2000 inputs, Ref vector is 1x1000, same for variant vector;

one binary output (0 or 1)

**4000 training samples: 2000 label 1, 200 label 0**

**Variant vector is obtained by randomly choosing 30 locations in Ref vector, and multiply that element by plus or minus [1.2, 1.5]**

**In one round, 40 label\_1 training samples, plus 40 label\_0 training samples are obtained.**

**ANN model:**

def baseline\_model2():

model = Sequential()

model.add(Dense(256, activation='relu', input\_dim = 2000))

model.add(BatchNormalization())

model.add(Dropout(0.3))

model.add(Dense(64, activation='relu'))

model.add(BatchNormalization())

model.add(Dropout(0.3))

model.add(Dense(16, activation='relu'))

model.add(BatchNormalization())

model.add(Dense(1, activation='sigmoid')) # Output layer for binary classification

model.compile(optimizer='adam',loss='binary\_crossentropy',metrics=['accuracy'])

return model

Model: "sequential\_12"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

dense\_45 (Dense) (None, 256) 512256

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batch\_normalization\_33 (Batc (None, 256) 1024

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_21 (Dropout) (None, 256) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_46 (Dense) (None, 64) 16448

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batch\_normalization\_34 (Batc (None, 64) 256

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_22 (Dropout) (None, 64) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_47 (Dense) (None, 16) 1040

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batch\_normalization\_35 (Batc (None, 16) 64

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_48 (Dense) (None, 1) 17

=================================================================

Total params: 531,105

Trainable params: 530,433

Non-trainable params: 672

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**MAIN function: generate\_dataset(…..)**

**num\_elements\_to\_change = 30**

**size\_ref\_vector = 1000**

**label\_1\_matrix = np.random.randint(0,size\_ref\_vector-1, size=(40,num\_elements\_to\_change) )**

**label\_0\_matrix = np.random.randint(0,size\_ref\_vector-1, size=(40,num\_elements\_to\_change) )**

**label\_1\_sign = np.random.choice([-1,1], size=(40,num\_elements\_to\_change) )**

**label\_0\_sign = np.random.choice([-1,1], size=(40,num\_elements\_to\_change) )**

**Nset = 50**

**# Each set of dataset is Nset x 80**

**train\_x, train\_y = generate\_dataset (Nset,size\_ref\_vector,label\_1\_matrix,label\_0\_matrix,label\_1\_sign,label\_0\_sign)**

**history = ANN\_model.fit(train\_x, train\_y, epochs=50, batch\_size=64, validation\_split=0.15,verbose = 2)**

54/54 - 0s - loss: 8.4680e-04 - accuracy: 1.0000 - val\_loss: 0.0021 - val\_accuracy: 0.9983

Execution time: 11.480006694793701

63/63 - 0s - loss: 1.2984e-04 - accuracy: 1.0000

Testing Accuracy = [0.00012984091881662607, 1.0]

**========================================================**

(2000, 2000)

(2000, 1)

\*\* CHECK Average BCE Loss for multiple samples: [0.0001]

Accuracy Score: 1.0

error0to1 = 0 ; label is 0

error1to0 = 0 ; label is 1

Testing total error = 0 percentError = 0.0

2000 test cases: 1000 label 0; 1000 label 1

test\_x.shape (2000, 2000)

test\_y.shape (2000, 1)

(2000, 1)

Compare between test\_y and y\_pred\_binary

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Sum of errors = 0

**------------------------------------------------------------------------------------------**

**Training cases:**

(4000, 2000)

(4000, 1)

(4000, 1)

\*\* CHECK Average BCE Loss for multiple samples: [0.0005]

Accuracy Score: 0.99975

error0to1 = 1 ; label is 0

error1to0 = 0 ; label is 1

Training total error = 1 percentError = 0.025

train cases: 4000