hyperparameterized_model

May 27, 2020

[25]:
```
import itertools
import numpy as np
import pandas as pd
# for data scaling and splitting
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
# for neural net
from tensorflow.keras.models import Sequential, load_model
from tensorflow.keras.layers import Dense, Dropout
from tensorflow.keras.wrappers.scikit_learn import KerasClassifier
# for evaluation
from sklearn.model_selection import KFold, cross_val_score, GridSearchCV
from sklearn.metrics import classification_report, confusion_matrix
```

[12]:
```
data = pd.read_csv("data/combined_expression.csv")
data.head()
data.shape
```

[12]:
```
(642, 16383)
```

[13]:
```
selected_genes = pd.read_csv('cleaned/boruta-99-25-0.01.csv')
selected_genes = selected_genes.values.tolist()
selected_genes = list(itertools.chain(*selected_genes))
```

[14]:
```
# retrieving proper columns
X = data.loc[:, selected_genes]
y = data['classification'].values

# scaling the data
scalar = MinMaxScaler()
x_scaled = scalar.fit_transform(X)

# splitting data (20% test, 80% train)
X_train, X_test, y_train, y_test = train_test_split(x_scaled, y, test_size=0.2,␣
↪random_state=0)
```

1 Gridsearch for Input and Output Layer

[15]:
```
def create_model(optimizer='rmsprop',init='glorot_uniform', dropout=0.3):
    model = Sequential()
    # adding layers and adding droplayers to avoid overfitting
    hidden_layers = len(selected_genes)
    model.add(Dense(hidden_layers, activation='relu'))
    model.add(Dropout(dropout))

    model.add(Dense((hidden_layers*0.5), activation='relu'))
    model.add(Dropout(dropout))

    model.add(Dense((hidden_layers*0.25), activation='relu'))
    model.add(Dropout(dropout))

    model.add(Dense((hidden_layers*0.125), activation='relu'))
    model.add(Dropout(dropout))

    model.add(Dense(1, activation='sigmoid'))
    # compiling
    model.compile(optimizer=optimizer, loss='binary_crossentropy',␣
 ↪metrics=['accuracy'])
    return model
```

[10]:
```
model = KerasClassifier(build_fn=create_model)
epochs = [50, 75, 100, 150]
batches = [16, 32, 64, 128]
optimizers = ['SGD', 'RMSprop', 'Adagrad', 'Adam', 'Adamax']
init = ['glorot_uniform', 'normal', 'uniform']
param_grid = dict(epochs=epochs,␣
 ↪batch_size=batches,optimizer=optimizers,init=init)
grid = GridSearchCV(estimator=model, param_grid=param_grid, cv=3, verbose=1,␣
 ↪n_jobs=-1)
grid_result = grid.fit(X_train, y_train)
```

```
Fitting 3 folds for each of 240 candidates, totalling 720 fits

[Parallel(n_jobs=-1)]: Using backend LokyBackend with 12 concurrent workers.
[Parallel(n_jobs=-1)]: Done  26 tasks      | elapsed:  1.3min
[Parallel(n_jobs=-1)]: Done 176 tasks      | elapsed: 11.4min
[Parallel(n_jobs=-1)]: Done 426 tasks      | elapsed: 20.1min
[Parallel(n_jobs=-1)]: Done 720 out of 720 | elapsed: 27.8min finished

Train on 513 samples
Epoch 1/50
513/513 [==============================] - 2s 4ms/sample - loss: 0.6588 -
accuracy: 0.6335
Epoch 2/50
```

```
513/513 [==============================] - 0s 179us/sample - loss: 0.6636 -
accuracy: 0.6374
Epoch 3/50
513/513 [==============================] - 0s 178us/sample - loss: 0.6325 -
accuracy: 0.6394
Epoch 4/50
513/513 [==============================] - 0s 176us/sample - loss: 0.6414 -
accuracy: 0.6530
Epoch 5/50
513/513 [==============================] - 0s 176us/sample - loss: 0.6293 -
accuracy: 0.6550
Epoch 6/50
513/513 [==============================] - 0s 176us/sample - loss: 0.6375 -
accuracy: 0.6433
Epoch 7/50
513/513 [==============================] - 0s 178us/sample - loss: 0.6292 -
accuracy: 0.6589
Epoch 8/50
513/513 [==============================] - 0s 173us/sample - loss: 0.6473 -
accuracy: 0.6472
Epoch 9/50
513/513 [==============================] - 0s 173us/sample - loss: 0.6206 -
accuracy: 0.6589
Epoch 10/50
513/513 [==============================] - 0s 177us/sample - loss: 0.6240 -
accuracy: 0.6472
Epoch 11/50
513/513 [==============================] - 0s 177us/sample - loss: 0.6012 -
accuracy: 0.6667
Epoch 12/50
513/513 [==============================] - 0s 177us/sample - loss: 0.6204 -
accuracy: 0.6628
Epoch 13/50
513/513 [==============================] - 0s 175us/sample - loss: 0.6229 -
accuracy: 0.6667
Epoch 14/50
513/513 [==============================] - 0s 176us/sample - loss: 0.6255 -
accuracy: 0.6764
Epoch 15/50
513/513 [==============================] - 0s 172us/sample - loss: 0.6140 -
accuracy: 0.6628
Epoch 16/50
513/513 [==============================] - 0s 172us/sample - loss: 0.6146 -
accuracy: 0.6628
Epoch 17/50
513/513 [==============================] - 0s 173us/sample - loss: 0.6230 -
accuracy: 0.6530
Epoch 18/50
```

```
513/513 [==============================] - 0s 182us/sample - loss: 0.5989 -
accuracy: 0.6667
Epoch 19/50
513/513 [==============================] - 0s 177us/sample - loss: 0.6198 -
accuracy: 0.6530
Epoch 20/50
513/513 [==============================] - 0s 177us/sample - loss: 0.6154 -
accuracy: 0.6569
Epoch 21/50
513/513 [==============================] - 0s 173us/sample - loss: 0.6041 -
accuracy: 0.6667
Epoch 22/50
513/513 [==============================] - 0s 172us/sample - loss: 0.6220 -
accuracy: 0.6550
Epoch 23/50
513/513 [==============================] - 0s 172us/sample - loss: 0.5984 -
accuracy: 0.6920
Epoch 24/50
513/513 [==============================] - 0s 169us/sample - loss: 0.6129 -
accuracy: 0.6647
Epoch 25/50
513/513 [==============================] - 0s 172us/sample - loss: 0.6167 -
accuracy: 0.6686
Epoch 26/50
513/513 [==============================] - 0s 171us/sample - loss: 0.6044 -
accuracy: 0.6842
Epoch 27/50
513/513 [==============================] - 0s 174us/sample - loss: 0.6051 -
accuracy: 0.6706
Epoch 28/50
513/513 [==============================] - 0s 172us/sample - loss: 0.6022 -
accuracy: 0.6706
Epoch 29/50
513/513 [==============================] - 0s 173us/sample - loss: 0.5895 -
accuracy: 0.6725
Epoch 30/50
513/513 [==============================] - 0s 173us/sample - loss: 0.6016 -
accuracy: 0.6667
Epoch 31/50
513/513 [==============================] - 0s 173us/sample - loss: 0.5876 -
accuracy: 0.6784
Epoch 32/50
513/513 [==============================] - 0s 171us/sample - loss: 0.6001 -
accuracy: 0.6764
Epoch 33/50
513/513 [==============================] - 0s 173us/sample - loss: 0.6041 -
accuracy: 0.6725
Epoch 34/50
```

```
513/513 [==============================] - 0s 176us/sample - loss: 0.5970 -
accuracy: 0.6725
Epoch 35/50
513/513 [==============================] - 0s 174us/sample - loss: 0.5950 -
accuracy: 0.6901
Epoch 36/50
513/513 [==============================] - 0s 178us/sample - loss: 0.6035 -
accuracy: 0.6745
Epoch 37/50
513/513 [==============================] - 0s 178us/sample - loss: 0.6135 -
accuracy: 0.6784
Epoch 38/50
513/513 [==============================] - 0s 177us/sample - loss: 0.5890 -
accuracy: 0.6628
Epoch 39/50
513/513 [==============================] - 0s 176us/sample - loss: 0.5914 -
accuracy: 0.6901
Epoch 40/50
513/513 [==============================] - 0s 180us/sample - loss: 0.5867 -
accuracy: 0.6842
Epoch 41/50
513/513 [==============================] - 0s 174us/sample - loss: 0.5979 -
accuracy: 0.6842
Epoch 42/50
513/513 [==============================] - 0s 174us/sample - loss: 0.5809 -
accuracy: 0.6725
Epoch 43/50
513/513 [==============================] - 0s 179us/sample - loss: 0.5979 -
accuracy: 0.6803
Epoch 44/50
513/513 [==============================] - 0s 177us/sample - loss: 0.5719 -
accuracy: 0.7154
Epoch 45/50
513/513 [==============================] - 0s 174us/sample - loss: 0.5838 -
accuracy: 0.6784
Epoch 46/50
513/513 [==============================] - 0s 174us/sample - loss: 0.5855 -
accuracy: 0.6764
Epoch 47/50
513/513 [==============================] - 0s 174us/sample - loss: 0.5863 -
accuracy: 0.6920
Epoch 48/50
513/513 [==============================] - 0s 177us/sample - loss: 0.6031 -
accuracy: 0.6823
Epoch 49/50
513/513 [==============================] - 0s 176us/sample - loss: 0.5945 -
accuracy: 0.6803
Epoch 50/50
```

```
513/513 [==============================] - 0s 179us/sample - loss: 0.5980 -
accuracy: 0.7115
```

[11]:
```
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
```

```
Best: 0.748538 using {'batch_size': 64, 'epochs': 50, 'init': 'uniform',
'optimizer': 'Adagrad'}
```

[12]:
```
grid_result.cv_results_
```

[12]:
```
{'mean_fit_time': array([22.87440864, 29.32623951, 27.11062702, 24.87289794,
20.84396259,
        18.22900732, 26.27965927, 24.82301545, 21.94619497, 23.58998831,
        21.23158709, 28.46431748, 26.44677734, 23.70279694, 25.34283153,
        30.35232393, 40.48846563, 36.20284955, 34.93643999, 34.72135162,
        26.96035504, 38.37487062, 35.27403617, 31.93025732, 33.35870632,
        30.03306031, 38.8240184 , 35.94747686, 33.71194688, 34.08766039,
        38.66021053, 54.42402458, 51.30228074, 47.13256033, 43.75787473,
        32.6987474 , 47.30237158, 46.35322007, 49.22904126, 51.82448411,
        39.05789129, 49.82315334, 47.01785088, 45.01280864, 44.63239392,
        54.73623848, 77.51178463, 72.62631536, 69.90273364, 67.29407962,
        58.81166538, 74.67216094, 70.26548068, 62.4366796 , 63.05991546,
        56.51756843, 72.38511475, 69.1286943 , 59.17332363, 60.26967128,
        19.44160636, 24.24690302, 19.6103344 , 15.49522948, 15.72469044,
        12.50570965, 17.16207361, 15.35686795, 14.43419957, 15.44468164,
        12.44476716, 17.07122095, 15.68642243, 16.03344806, 15.46375672,
        16.32414953, 23.51906927, 22.06611125, 20.95646683, 20.70602965,
        17.24162038, 22.50595037, 20.65463424, 21.01154367, 20.04171356,
        17.44929043, 24.40032999, 22.81567963, 22.34642585, 22.10524066,
        23.03069107, 31.85798264, 29.24490197, 28.24667676, 28.08741816,
        24.78644212, 32.92416827, 30.90960598, 28.50491134, 25.71532464,
        23.15834395, 32.5767796 , 30.32733591, 27.9823641 , 29.27016306,
        37.85635034, 44.30354905, 41.44908174, 38.65052501, 39.54783026,
        33.91771499, 42.63098025, 40.0758841 , 35.47434219, 38.71947861,
        34.93426355, 43.14829199, 41.05317911, 38.26613673, 38.19240991,
        15.22887961, 16.51566299, 14.36152895, 12.69561672, 11.78815667,
        10.36032232, 14.04511364, 12.62887263, 11.80364203, 12.07971247,
        10.47299401, 14.05826354, 12.74309023, 11.76599034, 11.93240436,
        11.71949959, 17.69823996, 16.10278273, 14.88679703, 15.84473101,
        12.95154309, 17.12429929, 15.58696206, 15.80541404, 15.67891089,
        13.9608264 , 17.04978863, 16.72527115, 14.59462976, 15.51814953,
        15.55159998, 22.6125559 , 20.09535901, 19.74343165, 20.49685907,
        15.69533769, 22.59175293, 20.74570441, 19.42902907, 19.6816953 ,
        16.67598756, 21.33367896, 19.67512814, 19.95726482, 20.72803768,
        23.36156996, 30.83866994, 29.17078273, 27.06641833, 27.6473283 ,
        23.30621688, 30.16569734, 27.88467741, 27.63531113, 26.98511394,
        22.32135431, 30.85097329, 28.93699106, 27.13127001, 25.43222801,
```

```
         9.26536576, 12.44886533, 11.30464427, 10.00841188,  8.84670965,
         7.97416496, 10.99503517, 10.1353031 ,  9.14568806,  8.92570233,
         8.77535939, 11.47268701, 10.62823939,  9.13388173,  9.50283734,
         8.87389   , 13.85570431, 12.441185  , 11.89486106, 11.434189  ,
         9.31972067, 13.35448607, 11.6709187 , 11.35297283, 10.7570111 ,
         9.74133666, 12.95398394, 12.14878718, 11.83271893, 13.25841387,
        15.46069217, 17.64109842, 15.55333455, 14.16656597, 14.77583194,
        11.35559559, 16.33277965, 14.74954184, 14.08646433, 13.39434997,
        11.69443361, 15.80208373, 14.46729422, 13.54168558, 14.25913811,
        14.85359661, 22.25664139, 20.24019265, 19.65864333, 19.78275871,
        16.50987864, 22.13029504, 20.61098083, 19.62201794, 18.46829756,
        16.832551  , 21.99949471, 19.620936  , 17.72649582, 16.16154067]),
 'std_fit_time': array([0.33019486, 0.07865309, 0.17124742, 0.03938971,
0.07091081,
        0.1263222 , 0.16193855, 0.21674429, 0.0583721 , 0.17204818,
        0.14335187, 0.16884901, 0.19294588, 0.1475078 , 0.16951218,
        0.14771833, 0.32244545, 0.31161945, 0.3806765 , 0.26576285,
        0.22708828, 0.19985474, 0.46840717, 0.64089358, 0.15876269,
        0.1792995 , 0.12744854, 0.32710575, 0.13193886, 0.45186857,
        0.10106198, 0.60128026, 0.23333074, 0.19216929, 0.38502442,
        0.18979099, 0.54416834, 2.24359833, 0.19712976, 0.07771243,
        0.73016061, 0.10510226, 0.76723204, 0.2533471 , 0.14203489,
        0.23628772, 0.76057416, 0.67790089, 0.32627233, 0.50967448,
        0.11459554, 1.03719458, 0.37674116, 0.35799261, 0.48406259,
        0.10325176, 1.09746097, 0.74368236, 0.43454413, 0.3123531 ,
        0.14932637, 0.12094659, 0.31392001, 0.01409023, 0.13533163,
        0.09064797, 0.24828752, 0.19891607, 0.35074699, 0.02477984,
        0.1438869 , 0.17921055, 0.51592391, 0.09914755, 0.23362501,
        0.19607435, 0.12602488, 0.18284008, 0.53763557, 0.06190507,
        0.16501883, 0.09109509, 0.59599928, 0.04828951, 0.19111704,
        0.18412604, 0.17373907, 0.77689589, 0.17749684, 0.16061851,
        0.07288733, 0.09181358, 0.40446302, 0.08469715, 0.46565696,
        0.19278512, 0.09237274, 0.09274412, 0.19633748, 0.11800316,
        0.25002796, 0.33319673, 0.22616244, 0.13821156, 0.32839281,
        0.08946832, 0.2933276 , 0.28768008, 0.44079004, 0.34879914,
        0.20684613, 0.35238453, 0.1066214 , 0.71425666, 0.17676895,
        0.11049241, 0.78383236, 0.44414218, 0.02718805, 0.30300546,
        0.09442857, 0.03533213, 0.12998143, 0.04573322, 0.03925825,
        0.14370603, 0.11025858, 0.02716431, 0.24019892, 0.03926459,
        0.06680516, 0.14114222, 0.31717808, 0.21785644, 0.39720337,
        0.09020461, 0.11777351, 0.05325872, 0.14959202, 0.15075828,
        0.0341632 , 0.0348954 , 0.13795212, 0.14740068, 0.07543113,
        0.1029088 , 0.15080837, 0.09982778, 0.18708848, 0.12063233,
        0.3225472 , 0.04068368, 0.15662822, 0.11182152, 0.03101806,
        0.03865702, 0.05339251, 0.2529298 , 0.148307  , 0.44829989,
        0.09593749, 0.17219947, 0.08930755, 0.29991291, 0.17389551,
        0.09827105, 0.16594123, 0.40858081, 0.17279645, 0.12944211,
```

```
        0.19821183, 0.36284867, 0.05605068, 0.11861258, 0.20383077,
        0.30130814, 0.06937366, 0.26910582, 0.2804785 , 0.07959193,
        0.24805213, 0.14730129, 0.16453798, 0.09584305, 0.11681823,
        0.03214616, 0.09492294, 0.08749925, 0.04577994, 0.05395681,
        0.1082882 , 0.03687504, 0.14975631, 0.10004558, 0.1632394 ,
        0.13273458, 0.08856438, 0.16730778, 0.06073153, 0.30465424,
        0.10493963, 0.11785952, 0.04458314, 0.08946303, 0.01545883,
        0.09164002, 0.16360754, 0.42971097, 0.12372104, 0.25047002,
        0.27347663, 0.13212515, 0.23319038, 0.16645111, 0.11933378,
        0.3170786 , 0.06602604, 0.05457419, 0.09659605, 0.28155583,
        0.11658653, 0.12554197, 0.1024081 , 0.11492282, 0.0849855 ,
        0.07734944, 0.40003469, 0.12463412, 0.16457195, 0.04869092,
        0.29313124, 0.07303873, 0.21152297, 0.00256393, 0.33798474,
        0.38979819, 0.17104688, 0.05175599, 0.14992718, 0.44161538]),
 'mean_score_time': array([2.99050331, 1.12851667, 1.42845456, 2.22466874,
2.13489135,
        2.72949203, 1.46295404, 1.30072618, 2.06867774, 1.8451515 ,
        1.34138624, 2.30361009, 1.70512509, 1.69976648, 1.15262802,
        3.04432702, 1.78927565, 2.6543777 , 1.70730305, 2.60676161,
        2.3252329 , 2.3496743 , 2.00878541, 1.82502198, 2.09912054,
        2.05081034, 2.72654565, 2.22021111, 1.70328665, 2.17869401,
        3.15548833, 3.10922511, 2.19433395, 1.66304445, 4.34767906,
        3.37934661, 3.99165757, 2.57081906, 2.61740843, 2.22544901,
        2.1217653 , 2.49619667, 2.53336771, 1.91762137, 1.68815287,
        2.91737119, 2.79854266, 2.28299379, 1.61239203, 2.06874394,
        2.61418096, 2.42468945, 2.59505995, 1.98240749, 2.65264416,
        2.05096428, 2.9744912 , 2.26328103, 2.03621697, 1.71798944,
        2.45509442, 1.54786364, 1.71349343, 1.42505638, 1.21050541,
        1.54978172, 1.79151495, 1.45835725, 1.33372696, 1.20553827,
        1.89011033, 1.63317561, 1.50689197, 1.38034527, 1.24220904,
        2.03164307, 1.87745444, 1.76954103, 1.35282564, 1.56580981,
        1.76289829, 1.77293881, 1.71136888, 1.52891835, 1.49231434,
        1.89170639, 1.95874373, 1.98093907, 1.64456948, 1.34551183,
        2.41046659, 2.1116755 , 1.85663295, 1.37758795, 2.7805078 ,
        2.61478257, 1.85311151, 1.6229674 , 2.24588354, 2.42576734,
        1.7210935 , 2.86644697, 2.78483621, 1.61910764, 1.32046707,
        1.98472794, 2.18792526, 2.08632723, 1.67821002, 2.12379042,
        2.01914374, 1.56084506, 1.50521461, 1.80931886, 1.74276304,
        1.99147662, 2.39944998, 2.2638243 , 1.31851061, 1.68069665,
        2.03898231, 1.5664258 , 1.40972662, 1.32812341, 1.35900998,
        1.29499292, 1.47922532, 1.68595695, 1.31357455, 1.15716402,
        1.26510533, 1.61491744, 1.37387037, 1.30138675, 1.17863003,
        1.62639308, 1.4315683 , 1.64004469, 1.38236992, 1.59978374,
        1.67023118, 1.20779999, 1.36765297, 1.51811687, 1.36713394,
        1.32382528, 1.28009907, 1.83596341, 1.4911267 , 1.10191909,
        1.32038267, 1.44641376, 2.04136642, 1.3817567 , 1.30693698,
        2.32628028, 1.43552478, 1.52692572, 1.26457699, 1.47448913,
```

```
        1.81358306, 1.24579763, 1.31886999, 2.47118084, 1.56897942,
        1.7882967 , 1.70306738, 1.53617676, 1.43045425, 1.48373842,
        1.65633512, 1.83493574, 2.0704937 , 1.41519189, 1.17228262,
        2.19211268, 1.97399211, 1.54640428, 1.31417902, 1.39348753,
        1.27906656, 1.28531162, 1.28162257, 1.20448629, 1.34439286,
        1.36832333, 1.06853382, 1.41855105, 1.22563537, 1.17683331,
        1.04344233, 1.64794715, 1.27071118, 1.36307057, 1.11771806,
        1.65620168, 1.22446434, 1.27875638, 1.26823902, 1.42996685,
        1.69518328, 1.14888692, 1.24790565, 1.65017311, 1.27139227,
        1.28461305, 1.27301749, 2.53801433, 2.01444308, 1.92872063,
        1.45379933, 1.4011306 , 1.55573996, 1.25063276, 1.47068405,
        1.56700643, 1.31768274, 1.34561261, 1.48178943, 1.35609166,
        1.29476674, 1.64790026, 1.65236139, 1.34973637, 1.17115164,
        1.77469834, 2.02950708, 1.94838103, 1.422484  , 1.34535567,
        1.8378911 , 1.3829511 , 1.43214774, 1.46042053, 1.79368504,
        1.64961068, 1.50192769, 1.30953964, 1.23223392, 0.36245386]),
 'std_score_time': array([0.13933433, 0.01609753, 0.03582326, 0.04239366,
0.18157524,
        0.10365876, 0.04524786, 0.0022447 , 0.03888824, 0.11195833,
        0.03679681, 0.07261847, 0.0341697 , 0.08646597, 0.06974961,
        0.18357697, 0.12365764, 0.15195587, 0.25202107, 0.22968891,
        0.13691681, 0.08094172, 0.01149788, 0.3129698 , 0.04456323,
        0.07371229, 0.02456735, 0.02057917, 0.12291137, 0.18104598,
        0.25862654, 0.17108055, 0.09321445, 0.1875548 , 0.1104794 ,
        0.37941863, 0.11737841, 0.99216837, 0.09072397, 0.07014607,
        0.07219676, 0.14913634, 0.18932625, 0.09889311, 0.04677085,
        0.20600764, 0.1358235 , 0.25570042, 0.06494318, 0.21276214,
        0.13724551, 0.44491098, 0.19024061, 0.24534253, 0.1608724 ,
        0.02735487, 0.04748548, 0.34535501, 0.14822322, 0.053534  ,
        0.03663492, 0.16522295, 0.0280038 , 0.03885323, 0.08364897,
        0.04017865, 0.2593094 , 0.0774934 , 0.09039595, 0.05107402,
        0.12611659, 0.05896532, 0.14445115, 0.00663814, 0.01610788,
        0.07035092, 0.051173  , 0.09341209, 0.15604281, 0.09135676,
        0.02841795, 0.02137805, 0.02356881, 0.13258097, 0.14861626,
        0.09569826, 0.07349671, 0.10636846, 0.15401675, 0.04132063,
        0.14067875, 0.14617444, 0.12804508, 0.11924521, 0.44481887,
        0.23960096, 0.2367634 , 0.06121038, 0.15198516, 0.27416463,
        0.1094949 , 0.22340671, 0.28749683, 0.13441945, 0.13477433,
        0.08580831, 0.14263929, 0.27075156, 0.13527477, 0.07709445,
        0.06361435, 0.10722404, 0.1807296 , 0.22393818, 0.15436482,
        0.05077525, 0.30970114, 0.25959451, 0.02067148, 0.08483031,
        0.34653469, 0.10307704, 0.08716669, 0.0364383 , 0.03171151,
        0.00660629, 0.11907937, 0.02523423, 0.14121684, 0.02636406,
        0.06170431, 0.03815904, 0.16539103, 0.09919661, 0.02096508,
        0.05170649, 0.07451482, 0.03012373, 0.1092553 , 0.05252085,
        0.01395632, 0.02157766, 0.10109216, 0.07985474, 0.06689039,
        0.06855037, 0.12225438, 0.01740979, 0.10447036, 0.05601143,
```

```
        0.12771525, 0.07664966, 0.11591054, 0.0770554 , 0.0916462 ,
        0.1011595 , 0.12592754, 0.19526287, 0.0387718 , 0.15532528,
        0.04356143, 0.01622972, 0.07300192, 0.25672516, 0.0636804 ,
        0.14120952, 0.0472283 , 0.15407084, 0.04475494, 0.02192799,
        0.17567671, 0.07586659, 0.07964361, 0.03068509, 0.05200486,
        0.166137  , 0.32643276, 0.11495931, 0.04557455, 0.17647404,
        0.01559424, 0.06465666, 0.08186963, 0.0761501 , 0.01904008,
        0.00706275, 0.02653112, 0.19280941, 0.0327531 , 0.03096175,
        0.02994727, 0.13908217, 0.01589563, 0.01886542, 0.03934065,
        0.05657509, 0.07860436, 0.07751538, 0.04886834, 0.1001748 ,
        0.04396843, 0.02609564, 0.03478522, 0.11679248, 0.06181542,
        0.02432464, 0.01870086, 0.07532448, 0.04705603, 0.16525173,
        0.10580254, 0.17657347, 0.20067814, 0.0042675 , 0.13006212,
        0.10105194, 0.02759059, 0.04378754, 0.05484496, 0.02997985,
        0.02453713, 0.05588239, 0.06112155, 0.0790724 , 0.02109911,
        0.12599505, 0.07442514, 0.14716945, 0.03392381, 0.07707901,
        0.09429186, 0.02835542, 0.12252095, 0.03073166, 0.06597582,
        0.14176165, 0.05939394, 0.08514205, 0.51751809, 0.03711157]),
 'param_batch_size': masked_array(data=[16, 16, 16, 16, 16, 16, 16, 16, 16, 16,
16, 16, 16, 16,
                   16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16,
                   16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16,
                   16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16,
                   16, 16, 16, 16, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
                   32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
                   32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
                   32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
                   32, 32, 32, 32, 32, 32, 32, 32, 64, 64, 64, 64, 64, 64,
                   64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64,
                   64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64,
                   64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64,
                   64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 128,
                   128, 128, 128, 128, 128, 128, 128, 128, 128, 128, 128,
                   128, 128, 128, 128, 128, 128, 128, 128, 128, 128, 128,
                   128, 128, 128, 128, 128, 128, 128, 128, 128, 128, 128,
                   128, 128, 128, 128, 128, 128, 128, 128, 128, 128, 128,
                   128, 128, 128, 128, 128, 128, 128, 128, 128, 128, 128,
                   128, 128, 128, 128],
             mask=[False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
```

```
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False],
        fill_value='?',
             dtype=object),
 'param_epochs': masked_array(data=[50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
50, 50, 50,
                  50, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75,
                  75, 75, 100, 100, 100, 100, 100, 100, 100, 100, 100,
                  100, 100, 100, 100, 100, 100, 150, 150, 150, 150, 150,
                  150, 150, 150, 150, 150, 150, 150, 150, 150, 150, 50,
                  50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
                  75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75,
                  75, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100,
                  100, 100, 100, 100, 100, 150, 150, 150, 150, 150, 150,
                  150, 150, 150, 150, 150, 150, 150, 150, 150, 50, 50,
                  50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 75,
                  75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75,
                  100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100,
                  100, 100, 100, 100, 150, 150, 150, 150, 150, 150, 150,
                  150, 150, 150, 150, 150, 150, 150, 150, 50, 50, 50, 50,
                  50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 75, 75, 75,
                  75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 100,
                  100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100,
                  100, 100, 100, 150, 150, 150, 150, 150, 150, 150, 150,
                  150, 150, 150, 150, 150, 150, 150],
            mask=[False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
```

```
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False,
                    False, False, False, False, False, False, False, False],
            fill_value='?',
                 dtype=object),
 'param_init': masked_array(data=['glorot_uniform', 'glorot_uniform',
'glorot_uniform',
                    'glorot_uniform', 'glorot_uniform', 'normal', 'normal',
                    'normal', 'normal', 'normal', 'uniform', 'uniform',
                    'uniform', 'uniform', 'uniform', 'glorot_uniform',
                    'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
                    'glorot_uniform', 'normal', 'normal', 'normal',
                    'normal', 'normal', 'uniform', 'uniform', 'uniform',
                    'uniform', 'uniform', 'glorot_uniform',
                    'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
                    'glorot_uniform', 'normal', 'normal', 'normal',
                    'normal', 'normal', 'uniform', 'uniform', 'uniform',
                    'uniform', 'uniform', 'glorot_uniform',
                    'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
                    'glorot_uniform', 'normal', 'normal', 'normal',
                    'normal', 'normal', 'uniform', 'uniform', 'uniform',
                    'uniform', 'uniform', 'glorot_uniform',
                    'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
```

```
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform', 'glorot_uniform',
'glorot_uniform', 'glorot_uniform', 'glorot_uniform',
'glorot_uniform', 'normal', 'normal', 'normal',
'normal', 'normal', 'uniform', 'uniform', 'uniform',
'uniform', 'uniform'],
```

```
             mask=[False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False,
                   False, False, False, False, False, False, False, False],
       fill_value='?',
            dtype=object),
 'param_optimizer': masked_array(data=['SGD', 'RMSprop', 'Adagrad', 'Adam',
'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
                   'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
```

```
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax', 'SGD',
          'RMSprop', 'Adagrad', 'Adam', 'Adamax'],
mask=[False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
      False, False, False, False, False, False, False, False,
```

```
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False,
                  False, False, False, False, False, False, False, False],
      fill_value='?',
           dtype=object),
'params': [{'batch_size': 16,
  'epochs': 50,
  'init': 'glorot_uniform',
  'optimizer': 'SGD'},
 {'batch_size': 16,
  'epochs': 50,
  'init': 'glorot_uniform',
  'optimizer': 'RMSprop'},
 {'batch_size': 16,
  'epochs': 50,
  'init': 'glorot_uniform',
  'optimizer': 'Adagrad'},
 {'batch_size': 16,
  'epochs': 50,
  'init': 'glorot_uniform',
  'optimizer': 'Adam'},
 {'batch_size': 16,
  'epochs': 50,
  'init': 'glorot_uniform',
  'optimizer': 'Adamax'},
 {'batch_size': 16, 'epochs': 50, 'init': 'normal', 'optimizer': 'SGD'},
 {'batch_size': 16, 'epochs': 50, 'init': 'normal', 'optimizer': 'RMSprop'},
 {'batch_size': 16, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adagrad'},
 {'batch_size': 16, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adam'},
 {'batch_size': 16, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adamax'},
 {'batch_size': 16, 'epochs': 50, 'init': 'uniform', 'optimizer': 'SGD'},
 {'batch_size': 16, 'epochs': 50, 'init': 'uniform', 'optimizer': 'RMSprop'},
 {'batch_size': 16, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adagrad'},
```

```
{'batch_size': 16, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 16, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 16,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 16,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 16,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 16,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 16,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 16, 'epochs': 75, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 16, 'epochs': 75, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 16, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 16, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 16, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 16, 'epochs': 75, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 16, 'epochs': 75, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 16, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 16, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 16, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 16,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 16,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 16,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 16,
 'epochs': 100,
 'init': 'glorot_uniform',
```

```
 'optimizer': 'Adam'},
{'batch_size': 16,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 16, 'epochs': 100, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 16, 'epochs': 100, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 16, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 16, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 16, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 16, 'epochs': 100, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 16, 'epochs': 100, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 16, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 16, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 16, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 16,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 16,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 16,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 16,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 16,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 16, 'epochs': 150, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 16, 'epochs': 150, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 16, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 16, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 16, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 16, 'epochs': 150, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 16, 'epochs': 150, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 16, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 16, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 16, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 32,
 'epochs': 50,
```

```
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 32,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 32,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 32,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 32,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 32, 'epochs': 50, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 32, 'epochs': 50, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 32, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 32, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 32, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 32, 'epochs': 50, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 32, 'epochs': 50, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 32, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 32, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 32, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 32,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 32,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 32,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 32,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 32,
 'epochs': 75,
 'init': 'glorot_uniform',
```

```
 'optimizer': 'Adamax'},
{'batch_size': 32, 'epochs': 75, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 32, 'epochs': 75, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 32, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 32, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 32, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 32, 'epochs': 75, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 32, 'epochs': 75, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 32, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 32, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 32, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 32,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 32,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 32,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 32,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 32,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 32, 'epochs': 100, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 32, 'epochs': 100, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 32, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 32, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 32, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 32, 'epochs': 100, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 32, 'epochs': 100, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 32, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 32, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 32, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 32,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 32,
 'epochs': 150,
```

```
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 32,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 32,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 32,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 32, 'epochs': 150, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 32, 'epochs': 150, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 32, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 32, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 32, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 32, 'epochs': 150, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 32, 'epochs': 150, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 32, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 32, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 32, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 64,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 64,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 64,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 64,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 64,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 64, 'epochs': 50, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 64, 'epochs': 50, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 64, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adagrad'},
```

```
{'batch_size': 64, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 64, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 64, 'epochs': 50, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 64, 'epochs': 50, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 64, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 64, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 64, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 64,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 64,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 64,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 64,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 64,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 64, 'epochs': 75, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 64, 'epochs': 75, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 64, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 64, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 64, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 64, 'epochs': 75, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 64, 'epochs': 75, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 64, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 64, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 64, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 64,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 64,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 64,
 'epochs': 100,
```

```
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 64,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 64,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 64, 'epochs': 100, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 64, 'epochs': 100, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 64, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 64, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 64, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 64, 'epochs': 100, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 64, 'epochs': 100, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 64, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 64, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 64, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 64,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 64,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 64,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 64,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 64,
 'epochs': 150,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 64, 'epochs': 150, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 64, 'epochs': 150, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 64, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 64, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 64, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 64, 'epochs': 150, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 64, 'epochs': 150, 'init': 'uniform', 'optimizer': 'RMSprop'},
```

```
{'batch_size': 64, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 64, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 64, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 128,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 128,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 128,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 128,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 128,
 'epochs': 50,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 128, 'epochs': 50, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 128, 'epochs': 50, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 128, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 128, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 128, 'epochs': 50, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 128, 'epochs': 50, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 128, 'epochs': 50, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 128, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 128, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 128, 'epochs': 50, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 128,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 128,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 128,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 128,
 'epochs': 75,
```

```
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 128,
 'epochs': 75,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 128, 'epochs': 75, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 128, 'epochs': 75, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 128, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 128, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 128, 'epochs': 75, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 128, 'epochs': 75, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 128, 'epochs': 75, 'init': 'uniform', 'optimizer': 'RMSprop'},
{'batch_size': 128, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adagrad'},
{'batch_size': 128, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adam'},
{'batch_size': 128, 'epochs': 75, 'init': 'uniform', 'optimizer': 'Adamax'},
{'batch_size': 128,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'SGD'},
{'batch_size': 128,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 128,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adagrad'},
{'batch_size': 128,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adam'},
{'batch_size': 128,
 'epochs': 100,
 'init': 'glorot_uniform',
 'optimizer': 'Adamax'},
{'batch_size': 128, 'epochs': 100, 'init': 'normal', 'optimizer': 'SGD'},
{'batch_size': 128, 'epochs': 100, 'init': 'normal', 'optimizer': 'RMSprop'},
{'batch_size': 128, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adagrad'},
{'batch_size': 128, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adam'},
{'batch_size': 128, 'epochs': 100, 'init': 'normal', 'optimizer': 'Adamax'},
{'batch_size': 128, 'epochs': 100, 'init': 'uniform', 'optimizer': 'SGD'},
{'batch_size': 128,
 'epochs': 100,
 'init': 'uniform',
 'optimizer': 'RMSprop'},
{'batch_size': 128,
```

```
   'epochs': 100,
   'init': 'uniform',
   'optimizer': 'Adagrad'},
  {'batch_size': 128, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adam'},
  {'batch_size': 128, 'epochs': 100, 'init': 'uniform', 'optimizer': 'Adamax'},
  {'batch_size': 128,
   'epochs': 150,
   'init': 'glorot_uniform',
   'optimizer': 'SGD'},
  {'batch_size': 128,
   'epochs': 150,
   'init': 'glorot_uniform',
   'optimizer': 'RMSprop'},
  {'batch_size': 128,
   'epochs': 150,
   'init': 'glorot_uniform',
   'optimizer': 'Adagrad'},
  {'batch_size': 128,
   'epochs': 150,
   'init': 'glorot_uniform',
   'optimizer': 'Adam'},
  {'batch_size': 128,
   'epochs': 150,
   'init': 'glorot_uniform',
   'optimizer': 'Adamax'},
  {'batch_size': 128, 'epochs': 150, 'init': 'normal', 'optimizer': 'SGD'},
  {'batch_size': 128, 'epochs': 150, 'init': 'normal', 'optimizer': 'RMSprop'},
  {'batch_size': 128, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adagrad'},
  {'batch_size': 128, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adam'},
  {'batch_size': 128, 'epochs': 150, 'init': 'normal', 'optimizer': 'Adamax'},
  {'batch_size': 128, 'epochs': 150, 'init': 'uniform', 'optimizer': 'SGD'},
  {'batch_size': 128,
   'epochs': 150,
   'init': 'uniform',
   'optimizer': 'RMSprop'},
  {'batch_size': 128,
   'epochs': 150,
   'init': 'uniform',
   'optimizer': 'Adagrad'},
  {'batch_size': 128, 'epochs': 150, 'init': 'uniform', 'optimizer': 'Adam'},
  {'batch_size': 128,
   'epochs': 150,
   'init': 'uniform',
   'optimizer': 'Adamax'}],
 'split0_test_score': array([0.71345031, 0.70760232, 0.71929824, 0.69005847,
0.73099416,
        0.74269009, 0.74269009, 0.73099416, 0.66666669, 0.74269009,
```

```
       0.71345031, 0.67836255, 0.71345031, 0.69005847, 0.72514617,
       0.7368421 , 0.69590646, 0.73099416, 0.71929824, 0.73099416,
       0.73099416, 0.59649122, 0.73099416, 0.71929824, 0.71345031,
       0.75438595, 0.69005847, 0.74269009, 0.67251462, 0.72514617,
       0.71929824, 0.71929824, 0.74269009, 0.66666669, 0.70175439,
       0.75438595, 0.61988306, 0.74269009, 0.68421054, 0.71345031,
       0.69005847, 0.69005847, 0.7368421 , 0.67836255, 0.72514617,
       0.72514617, 0.70175439, 0.7368421 , 0.69590646, 0.74853802,
       0.69590646, 0.67251462, 0.75438595, 0.70175439, 0.69005847,
       0.74853802, 0.67836255, 0.74853802, 0.70760232, 0.66666669,
       0.66666669, 0.6608187 , 0.72514617, 0.70175439, 0.72514617,
       0.68421054, 0.7368421 , 0.70760232, 0.71929824, 0.72514617,
       0.66666669, 0.69005847, 0.69590646, 0.71345031, 0.73099416,
       0.70175439, 0.74269009, 0.75438595, 0.68421054, 0.68421054,
       0.69005847, 0.66666669, 0.7368421 , 0.69590646, 0.71345031,
       0.69590646, 0.69005847, 0.74853802, 0.72514617, 0.69590646,
       0.75438595, 0.67251462, 0.74269009, 0.70175439, 0.69005847,
       0.72514617, 0.70175439, 0.74269009, 0.71345031, 0.66666669,
       0.74269009, 0.6608187 , 0.74853802, 0.70175439, 0.69005847,
       0.76023394, 0.72514617, 0.74853802, 0.67251462, 0.71345031,
       0.75438595, 0.69590646, 0.75438595, 0.69005847, 0.74269009,
       0.73099416, 0.70175439, 0.75438595, 0.69005847, 0.72514617,
       0.66666669, 0.66666669, 0.72514617, 0.71345031, 0.70760232,
       0.66666669, 0.67251462, 0.70760232, 0.73099416, 0.71345031,
       0.66666669, 0.72514617, 0.74853802, 0.69590646, 0.76023394,
       0.67251462, 0.71345031, 0.74853802, 0.63742691, 0.74269009,
       0.69590646, 0.71345031, 0.75438595, 0.71345031, 0.73099416,
       0.66666669, 0.70760232, 0.71929824, 0.71929824, 0.72514617,
       0.68421054, 0.72514617, 0.74853802, 0.64912283, 0.73099416,
       0.66666669, 0.54970759, 0.75438595, 0.70175439, 0.7368421 ,
       0.67251462, 0.70175439, 0.76023394, 0.70760232, 0.70760232,
       0.71345031, 0.7368421 , 0.7719298 , 0.70760232, 0.68421054,
       0.72514617, 0.70760232, 0.74853802, 0.69005847, 0.71929824,
       0.72514617, 0.71929824, 0.71345031, 0.70760232, 0.69590646,
       0.66666669, 0.70760232, 0.71929824, 0.71345031, 0.7719298 ,
       0.66666669, 0.67251462, 0.75438595, 0.72514617, 0.75438595,
       0.66666669, 0.68421054, 0.71345031, 0.72514617, 0.74269009,
       0.66666669, 0.69005847, 0.73099416, 0.71345031, 0.70760232,
       0.66666669, 0.69590646, 0.73099416, 0.72514617, 0.70175439,
       0.66666669, 0.69005847, 0.7368421 , 0.70175439, 0.7368421 ,
       0.66666669, 0.74269009, 0.74853802, 0.74853802, 0.73099416,
       0.66666669, 0.72514617, 0.7368421 , 0.67251462, 0.71929824,
       0.66666669, 0.69005847, 0.74269009, 0.63157892, 0.75438595,
       0.66666669, 0.7368421 , 0.7368421 , 0.68421054, 0.69005847,
       0.66666669, 0.56140351, 0.7368421 , 0.73099416, 0.70175439,
       0.67251462, 0.72514617, 0.70760232, 0.71929824, 0.72514617]),
'split1_test_score': array([0.62573099, 0.64912283, 0.67836255, 0.68421054,
```

```
0.7368421 ,
        0.64327484, 0.67251462, 0.69005847, 0.72514617, 0.70175439,
        0.69005847, 0.71929824, 0.71345031, 0.70175439, 0.7368421 ,
        0.69590646, 0.71345031, 0.70175439, 0.73099416, 0.67836255,
        0.70175439, 0.68421054, 0.70175439, 0.71929824, 0.69590646,
        0.71929824, 0.70760232, 0.68421054, 0.7368421 , 0.71929824,
        0.71345031, 0.69590646, 0.71929824, 0.74269009, 0.69590646,
        0.71929824, 0.67836255, 0.72514617, 0.66666669, 0.69590646,
        0.70175439, 0.70175439, 0.69590646, 0.73099416, 0.71345031,
        0.70760232, 0.72514617, 0.73099416, 0.67836255, 0.71929824,
        0.71929824, 0.73099416, 0.71929824, 0.69590646, 0.71345031,
        0.70175439, 0.70175439, 0.73099416, 0.65497077, 0.70760232,
        0.63157892, 0.70175439, 0.69590646, 0.70175439, 0.72514617,
        0.62573099, 0.70175439, 0.71345031, 0.70175439, 0.71929824,
        0.62573099, 0.72514617, 0.66666669, 0.70760232, 0.73099416,
        0.63157892, 0.68421054, 0.69590646, 0.70175439, 0.69590646,
        0.63157892, 0.67251462, 0.73099416, 0.69005847, 0.72514617,
        0.64327484, 0.69590646, 0.70175439, 0.71929824, 0.69005847,
        0.70175439, 0.74853802, 0.70760232, 0.73099416, 0.73099416,
        0.69590646, 0.68421054, 0.70760232, 0.69590646, 0.69590646,
        0.70175439, 0.69590646, 0.71345031, 0.70760232, 0.69590646,
        0.70760232, 0.73099416, 0.74269009, 0.65497077, 0.73099416,
        0.71929824, 0.70760232, 0.7368421 , 0.73099416, 0.69590646,
        0.69590646, 0.69005847, 0.73099416, 0.67836255, 0.7368421 ,
        0.62573099, 0.7368421 , 0.67836255, 0.7368421 , 0.74853802,
        0.62573099, 0.70175439, 0.69005847, 0.72514617, 0.75438595,
        0.62573099, 0.71929824, 0.7368421 , 0.71345031, 0.71345031,
        0.62573099, 0.66666669, 0.73099416, 0.70760232, 0.73099416,
        0.62573099, 0.70760232, 0.70175439, 0.67251462, 0.74853802,
        0.62573099, 0.70175439, 0.71345031, 0.7368421 , 0.71929824,
        0.64327484, 0.70175439, 0.7368421 , 0.70175439, 0.73099416,
        0.62573099, 0.70760232, 0.74269009, 0.75438595, 0.70175439,
        0.62573099, 0.72514617, 0.69590646, 0.67251462, 0.71345031,
        0.67836255, 0.69590646, 0.71345031, 0.69005847, 0.71345031,
        0.63742691, 0.67836255, 0.72514617, 0.69590646, 0.72514617,
        0.71345031, 0.73099416, 0.7368421 , 0.69590646, 0.70760232,
        0.62573099, 0.71929824, 0.71345031, 0.70175439, 0.71929824,
        0.62573099, 0.74853802, 0.63742691, 0.71929824, 0.73099416,
        0.62573099, 0.70760232, 0.65497077, 0.65497077, 0.71345031,
        0.62573099, 0.7368421 , 0.68421054, 0.68421054, 0.7368421 ,
        0.62573099, 0.70175439, 0.70175439, 0.68421054, 0.7368421 ,
        0.62573099, 0.70760232, 0.70760232, 0.70175439, 0.76023394,
        0.62573099, 0.7368421 , 0.75438595, 0.74269009, 0.74269009,
        0.62573099, 0.69005847, 0.73099416, 0.70760232, 0.72514617,
        0.62573099, 0.71929824, 0.68421054, 0.73099416, 0.69590646,
        0.62573099, 0.71345031, 0.68421054, 0.71345031, 0.71345031,
        0.62573099, 0.71929824, 0.71929824, 0.65497077, 0.68421054,
```

```
        0.62573099, 0.72514617, 0.69590646, 0.70760232, 0.73099416]),
 'split2_test_score': array([0.76023394, 0.71929824, 0.75438595, 0.68421054,
0.71345031,
        0.76023394, 0.69590646, 0.7368421 , 0.74269009, 0.70175439,
        0.76608187, 0.61988306, 0.76023394, 0.68421054, 0.71929824,
        0.70760232, 0.70760232, 0.72514617, 0.64327484, 0.71929824,
        0.72514617, 0.70760232, 0.71929824, 0.61403507, 0.74853802,
        0.74853802, 0.67836255, 0.7368421 , 0.7368421 , 0.68421054,
        0.71929824, 0.72514617, 0.76023394, 0.69590646, 0.69005847,
        0.71929824, 0.64912283, 0.73099416, 0.66666669, 0.69005847,
        0.70760232, 0.68421054, 0.7368421 , 0.67836255, 0.69005847,
        0.71929824, 0.6608187 , 0.71929824, 0.70175439, 0.70760232,
        0.71929824, 0.66666669, 0.73099416, 0.70760232, 0.68421054,
        0.67836255, 0.71345031, 0.72514617, 0.68421054, 0.70760232,
        0.69005847, 0.73099416, 0.7368421 , 0.64912283, 0.71929824,
        0.69005847, 0.72514617, 0.74853802, 0.69590646, 0.69590646,
        0.69005847, 0.70760232, 0.7368421 , 0.70760232, 0.71929824,
        0.73099416, 0.71929824, 0.75438595, 0.71345031, 0.73099416,
        0.70760232, 0.69590646, 0.7368421 , 0.67836255, 0.71929824,
        0.71345031, 0.57894737, 0.73099416, 0.72514617, 0.71929824,
        0.73099416, 0.69590646, 0.74269009, 0.70175439, 0.67836255,
        0.71929824, 0.66666669, 0.74269009, 0.69590646, 0.71345031,
        0.72514617, 0.69005847, 0.75438595, 0.69590646, 0.69590646,
        0.70175439, 0.70175439, 0.72514617, 0.67251462, 0.67836255,
        0.72514617, 0.57894737, 0.71345031, 0.70760232, 0.68421054,
        0.70760232, 0.71929824, 0.72514617, 0.67251462, 0.68421054,
        0.69005847, 0.63157892, 0.74269009, 0.71929824, 0.74269009,
        0.69005847, 0.71929824, 0.75438595, 0.70175439, 0.71345031,
        0.69005847, 0.73099416, 0.76023394, 0.74269009, 0.73099416,
        0.69005847, 0.71929824, 0.73099416, 0.71345031, 0.7368421 ,
        0.69005847, 0.69005847, 0.74853802, 0.71345031, 0.71345031,
        0.69005847, 0.71929824, 0.76023394, 0.65497077, 0.70760232,
        0.74269009, 0.71345031, 0.7368421 , 0.73099416, 0.69005847,
        0.76023394, 0.71345031, 0.71929824, 0.71929824, 0.71345031,
        0.69005847, 0.67836255, 0.76023394, 0.70760232, 0.71929824,
        0.69005847, 0.71929824, 0.7368421 , 0.74269009, 0.67251462,
        0.75438595, 0.69005847, 0.75438595, 0.71345031, 0.69590646,
        0.72514617, 0.70760232, 0.72514617, 0.69005847, 0.70175439,
        0.69005847, 0.72514617, 0.75438595, 0.71929824, 0.74853802,
        0.69005847, 0.73099416, 0.75438595, 0.7368421 , 0.73099416,
        0.69005847, 0.74269009, 0.74269009, 0.72514617, 0.71929824,
        0.69005847, 0.56140351, 0.76023394, 0.71929824, 0.71929824,
        0.69005847, 0.71345031, 0.73099416, 0.71929824, 0.72514617,
        0.69005847, 0.65497077, 0.7368421 , 0.71929824, 0.70760232,
        0.69005847, 0.70760232, 0.73099416, 0.70760232, 0.70175439,
        0.69005847, 0.72514617, 0.70760232, 0.6608187 , 0.68421054,
        0.69005847, 0.71345031, 0.75438595, 0.69005847, 0.70175439,
```

```
        0.69005847, 0.62573099, 0.74853802, 0.69590646, 0.70760232,
        0.69005847, 0.71929824, 0.72514617, 0.71929824, 0.69005847,
        0.69005847, 0.69005847, 0.73099416, 0.67251462, 0.71929824]),
 'mean_test_score': array([0.69980508, 0.6920078 , 0.71734891, 0.68615985,
0.72709552,
        0.71539962, 0.70370372, 0.71929824, 0.71150098, 0.71539962,
        0.72319688, 0.67251462, 0.72904485, 0.6920078 , 0.7270955 ,
        0.71345029, 0.70565303, 0.71929824, 0.69785575, 0.70955165,
        0.71929824, 0.66276803, 0.71734893, 0.68421052, 0.71929826,
        0.74074074, 0.69200778, 0.72124757, 0.7153996 , 0.70955165,
        0.71734893, 0.71345029, 0.74074076, 0.70175441, 0.69590644,
        0.73099415, 0.64912281, 0.73294348, 0.67251464, 0.69980508,
        0.69980506, 0.6920078 , 0.72319688, 0.69590642, 0.70955165,
        0.71734891, 0.69590642, 0.72904483, 0.6920078 , 0.72514619,
        0.71150098, 0.69005849, 0.73489279, 0.70175439, 0.69590644,
        0.70955165, 0.69785575, 0.73489279, 0.68226121, 0.69395711,
        0.66276803, 0.69785575, 0.71929824, 0.68421054, 0.72319686,
        0.66666667, 0.72124755, 0.72319688, 0.70565303, 0.71345029,
        0.66081872, 0.70760232, 0.69980508, 0.70955165, 0.72709552,
        0.68810916, 0.71539962, 0.73489279, 0.69980508, 0.70370372,
        0.67641324, 0.67836259, 0.73489279, 0.68810916, 0.71929824,
        0.68421054, 0.65497077, 0.72709552, 0.72319686, 0.70175439,
        0.72904483, 0.70565303, 0.73099416, 0.71150098, 0.69980506,
        0.71345029, 0.68421054, 0.73099416, 0.70175441, 0.69200782,
        0.72319688, 0.68226121, 0.73879143, 0.70175439, 0.69395713,
        0.72319688, 0.71929824, 0.73879143, 0.66666667, 0.70760234,
        0.73294346, 0.66081872, 0.73489279, 0.70955165, 0.70760236,
        0.71150098, 0.7037037 , 0.7368421 , 0.68031188, 0.7153996 ,
        0.66081872, 0.67836257, 0.7153996 , 0.72319688, 0.73294348,
        0.66081872, 0.69785575, 0.71734891, 0.71929824, 0.72709552,
        0.66081872, 0.72514619, 0.74853802, 0.71734895, 0.73489281,
        0.66276803, 0.69980508, 0.73684212, 0.68615985, 0.73684212,
        0.67056531, 0.7037037 , 0.73489279, 0.69980508, 0.73099416,
        0.66081872, 0.70955165, 0.73099416, 0.7037037 , 0.71734891,
        0.69005849, 0.71345029, 0.74074074, 0.69395713, 0.71734893,
        0.68421054, 0.65692008, 0.73879143, 0.72514619, 0.71734893,
        0.66276803, 0.70175437, 0.73879145, 0.69590642, 0.71345029,
        0.69395711, 0.71734893, 0.74074074, 0.71345029, 0.69005849,
        0.70565301, 0.69200778, 0.74269005, 0.69980508, 0.71345029,
        0.72124755, 0.71929824, 0.72514619, 0.69785575, 0.70175439,
        0.66081872, 0.71734891, 0.72904483, 0.71150098, 0.74658869,
        0.66081872, 0.71734893, 0.7153996 , 0.7270955 , 0.73879143,
        0.66081872, 0.71150098, 0.70370372, 0.70175437, 0.72514621,
        0.66081872, 0.66276803, 0.72514621, 0.70565303, 0.72124755,
        0.66081872, 0.70370372, 0.72124757, 0.70955165, 0.72124755,
        0.66081872, 0.68421052, 0.7270955 , 0.70760234, 0.73489279,
        0.66081872, 0.72904483, 0.74463938, 0.73294348, 0.72514621,
```

```
        0.66081872, 0.71345027, 0.72514619, 0.68031188, 0.70955165,
        0.66081872, 0.70760234, 0.72709552, 0.68421052, 0.71734893,
        0.66081872, 0.6920078 , 0.72319688, 0.69785577, 0.7037037 ,
        0.66081872, 0.66666667, 0.7270955 , 0.70175439, 0.6920078 ,
        0.66276803, 0.71345027, 0.71150098, 0.69980506, 0.72514619]),
 'std_test_score': array([0.05575186, 0.03069787, 0.03106702, 0.00275674,
0.0099396 ,
        0.05150039, 0.02917473, 0.02081302, 0.03250162, 0.01929727,
        0.03179241, 0.04079619, 0.02205401, 0.00729368, 0.00729368,
        0.0172159 , 0.00729367, 0.01263302, 0.03888877, 0.022565  ,
        0.01263302, 0.04782785, 0.0120164 , 0.04962154, 0.02188103,
        0.01534894, 0.0120164 , 0.02629773, 0.03032426, 0.01807722,
        0.00275674, 0.012633  , 0.01676868, 0.03131068, 0.00477484,
        0.0165405 , 0.02387415, 0.0072937 , 0.00827025, 0.00993962,
        0.00729368, 0.00729368, 0.01929724, 0.02481078, 0.01458736,
        0.00729368, 0.02658515, 0.00729368, 0.00993963, 0.01721592,
        0.01102699, 0.02904418, 0.01458735, 0.00477482, 0.01263303,
        0.02917473, 0.01458738, 0.00993962, 0.02153091, 0.01929724,
        0.02403281, 0.02878134, 0.01721589, 0.02481075, 0.00275674,
        0.02904418, 0.01458735, 0.01807723, 0.0099396 , 0.012633  ,
        0.02658515, 0.0143245 , 0.02878132, 0.00275677, 0.00551351,
        0.04171724, 0.02403281, 0.02756749, 0.0120164 , 0.01987922,
        0.03250163, 0.01263303, 0.00275674, 0.0072937 , 0.00477482,
        0.02981884, 0.05380965, 0.01929726, 0.00275674, 0.01263302,
        0.02153091, 0.03179241, 0.01654053, 0.01378376, 0.022565  ,
        0.012633  , 0.0143245 , 0.01654053, 0.00827025, 0.01929726,
        0.01676867, 0.01534896, 0.01807722, 0.00477482, 0.00275677,
        0.02629775, 0.01263302, 0.00993963, 0.00827025, 0.02188105,
        0.01534894, 0.05808836, 0.01676865, 0.01676868, 0.02526605,
        0.01458736, 0.0120164 , 0.01263302, 0.00729368, 0.02256497,
        0.02658515, 0.04376208, 0.02715087, 0.0099396 , 0.01807724,
        0.02658515, 0.01929726, 0.02715085, 0.01263302, 0.01929724,
        0.02658515, 0.00477484, 0.00954968, 0.01929726, 0.01929726,
        0.02715085, 0.02355369, 0.00827025, 0.03454199, 0.00477484,
        0.03179242, 0.00993962, 0.02355369, 0.01929727, 0.0143245 ,
        0.02658515, 0.00729368, 0.02081302, 0.03519584, 0.00729368,
        0.04079621, 0.00954966, 0.00551351, 0.03387552, 0.01929727,
        0.05629447, 0.07584826, 0.01458736, 0.02188103, 0.01458735,
        0.02715085, 0.01909934, 0.03032426, 0.0165405 , 0.00477484,
        0.01458738, 0.01676865, 0.02403279, 0.02188105, 0.01721592,
        0.04969803, 0.0120164 , 0.01263303, 0.00993962, 0.012633  ,
        0.00551348, 0.00954968, 0.00954966, 0.00729367, 0.00477482,
        0.02658515, 0.00729368, 0.01807722, 0.00729368, 0.02153091,
        0.02658515, 0.03250163, 0.05513502, 0.00729368, 0.01102699,
        0.02658515, 0.02403281, 0.03646841, 0.03308101, 0.01263303,
        0.02658515, 0.07417659, 0.03131068, 0.01534894, 0.0120164 ,
        0.02658515, 0.00729368, 0.01378376, 0.01807722, 0.01458735,
```

```
        0.02658515, 0.02188103, 0.01378376, 0.00827025, 0.02153093,
        0.02658515, 0.01534896, 0.0099396 , 0.01807724, 0.01721592,
        0.02658515, 0.0165405 , 0.01263303, 0.01987922, 0.01807722,
        0.02658515, 0.01263303, 0.03069788, 0.04079621, 0.02629772,
        0.02658515, 0.04782785, 0.02797797, 0.0120164 , 0.00993962,
        0.02658515, 0.07443229, 0.00729368, 0.03342384, 0.00729368,
        0.02715085, 0.0165405 , 0.01458736, 0.01987922, 0.00477484]),
 'rank_test_score': array([154, 180,  89, 193,  39,  94, 137,  73, 111,  94,
57, 210,  34,
        180,  44, 101, 132,  73, 165, 118,  73, 215,  82, 199,  72,   6,
        186,  66,  97, 118,  82, 101,   5, 145, 170,  33, 240,  25, 209,
        154, 161, 180,  57, 172, 118,  89, 172,  35, 180,  51, 111, 188,
         18, 147, 170, 118, 165,  18, 202, 177, 215, 165,  73, 195,  64,
        212,  68,  57, 132, 101, 221, 131, 154, 118,  39, 191,  94,  18,
        154, 137, 208, 206,  18, 191,  73, 195, 239,  39,  64, 147,  35,
        132,  29, 111, 161, 101, 195,  29, 145, 179,  57, 202,  10, 147,
        175,  57,  73,  10, 212, 128,  28, 221,  18, 118, 127, 111, 141,
         16, 204,  97, 221, 207,  97,  57,  25, 221, 165,  89,  73,  39,
        221,  51,   1,  81,  17, 215, 154,  14, 193,  14, 211, 141,  18,
        154,  29, 221, 118,  29, 141,  89, 188, 101,   6, 175,  82, 195,
        238,  10,  51,  82, 215, 152,   9, 172, 101, 177,  82,   6, 101,
        188, 136, 186,   4, 154, 101,  68,  73,  51, 165, 147, 221,  89,
         35, 111,   2, 221,  82,  97,  44,  10, 221, 111, 137, 152,  48,
        221, 215,  48, 132,  68, 221, 137,  66, 118,  68, 221, 199,  44,
        128,  18, 221,  35,   3,  25,  48, 221, 109,  51, 204, 118, 221,
        128,  39, 199,  82, 221, 180,  57, 164, 141, 221, 212,  44, 147,
        180, 215, 109, 111, 161,  51], dtype=int32)}
```

2 Testing the Model

[18]:
```
# model = KerasClassifier(build_fn=create_model, epochs=grid_result.
 ↪best_params_['epochs'], batch_size=grid_result.
 ↪best_params_['batch_size'],optimizer=grid_result.
 ↪best_params_['optimizer'],init=grid_result.best_params_['init'])
model = KerasClassifier(build_fn=create_model, epochs=50, batch_size=64,␣
 ↪optimizer='Adagrad',init='uniform')
kfold = KFold(n_splits=3, shuffle=True)
results = cross_val_score(model, X_train, y_train, cv=kfold)
print("Baseline Accuracy: %.2f%% (%.2f%%)" % (results.mean()*100, results.
 ↪std()*100))
```

```
Train on 342 samples
Epoch 1/50
342/342 [==============================] - 2s 4ms/sample - loss: 0.6859 -
accuracy: 0.5936
Epoch 2/50
```

```
342/342 [==============================] - 0s 176us/sample - loss: 0.6620 -
accuracy: 0.5789
Epoch 3/50
342/342 [==============================] - 0s 161us/sample - loss: 0.6557 -
accuracy: 0.6111
Epoch 4/50
342/342 [==============================] - 0s 162us/sample - loss: 0.6507 -
accuracy: 0.6433
Epoch 5/50
342/342 [==============================] - 0s 157us/sample - loss: 0.6571 -
accuracy: 0.6111
Epoch 6/50
342/342 [==============================] - 0s 159us/sample - loss: 0.6500 -
accuracy: 0.6199
Epoch 7/50
342/342 [==============================] - 0s 156us/sample - loss: 0.6424 -
accuracy: 0.6345
Epoch 8/50
342/342 [==============================] - 0s 155us/sample - loss: 0.6340 -
accuracy: 0.6374
Epoch 9/50
342/342 [==============================] - 0s 154us/sample - loss: 0.6204 -
accuracy: 0.6491
Epoch 10/50
342/342 [==============================] - 0s 156us/sample - loss: 0.6281 -
accuracy: 0.6462
Epoch 11/50
342/342 [==============================] - 0s 157us/sample - loss: 0.6190 -
accuracy: 0.6462
Epoch 12/50
342/342 [==============================] - 0s 153us/sample - loss: 0.6305 -
accuracy: 0.6579
Epoch 13/50
342/342 [==============================] - 0s 153us/sample - loss: 0.6168 -
accuracy: 0.6404
Epoch 14/50
342/342 [==============================] - 0s 155us/sample - loss: 0.6371 -
accuracy: 0.6404
Epoch 15/50
342/342 [==============================] - 0s 156us/sample - loss: 0.6229 -
accuracy: 0.6404
Epoch 16/50
342/342 [==============================] - 0s 158us/sample - loss: 0.6142 -
accuracy: 0.6667
Epoch 17/50
342/342 [==============================] - 0s 157us/sample - loss: 0.6192 -
accuracy: 0.6520
Epoch 18/50
```

```
342/342 [==============================] - 0s 155us/sample - loss: 0.6083 -
accuracy: 0.6667
Epoch 19/50
342/342 [==============================] - 0s 155us/sample - loss: 0.6126 -
accuracy: 0.6725
Epoch 20/50
342/342 [==============================] - 0s 153us/sample - loss: 0.6133 -
accuracy: 0.6608
Epoch 21/50
342/342 [==============================] - 0s 155us/sample - loss: 0.6014 -
accuracy: 0.6842
Epoch 22/50
342/342 [==============================] - 0s 154us/sample - loss: 0.6035 -
accuracy: 0.6784
Epoch 23/50
342/342 [==============================] - 0s 154us/sample - loss: 0.6210 -
accuracy: 0.6637
Epoch 24/50
342/342 [==============================] - 0s 158us/sample - loss: 0.6175 -
accuracy: 0.6725
Epoch 25/50
342/342 [==============================] - 0s 158us/sample - loss: 0.6200 -
accuracy: 0.6667
Epoch 26/50
342/342 [==============================] - 0s 157us/sample - loss: 0.5892 -
accuracy: 0.6754
Epoch 27/50
342/342 [==============================] - 0s 154us/sample - loss: 0.5892 -
accuracy: 0.6871
Epoch 28/50
342/342 [==============================] - 0s 155us/sample - loss: 0.5875 -
accuracy: 0.6901
Epoch 29/50
342/342 [==============================] - 0s 158us/sample - loss: 0.5976 -
accuracy: 0.6813
Epoch 30/50
342/342 [==============================] - 0s 157us/sample - loss: 0.5749 -
accuracy: 0.7047
Epoch 31/50
342/342 [==============================] - 0s 155us/sample - loss: 0.5864 -
accuracy: 0.6725
Epoch 32/50
342/342 [==============================] - 0s 158us/sample - loss: 0.5604 -
accuracy: 0.7047
Epoch 33/50
342/342 [==============================] - 0s 156us/sample - loss: 0.5886 -
accuracy: 0.6959
Epoch 34/50
```

```
342/342 [==============================] - 0s 157us/sample - loss: 0.5877 -
accuracy: 0.6988
Epoch 35/50
342/342 [==============================] - 0s 159us/sample - loss: 0.5805 -
accuracy: 0.7076
Epoch 36/50
342/342 [==============================] - 0s 158us/sample - loss: 0.5601 -
accuracy: 0.7193
Epoch 37/50
342/342 [==============================] - 0s 160us/sample - loss: 0.5803 -
accuracy: 0.7135
Epoch 38/50
342/342 [==============================] - 0s 158us/sample - loss: 0.5670 -
accuracy: 0.7339
Epoch 39/50
342/342 [==============================] - 0s 157us/sample - loss: 0.5825 -
accuracy: 0.6959
Epoch 40/50
342/342 [==============================] - 0s 158us/sample - loss: 0.5621 -
accuracy: 0.7076
Epoch 41/50
342/342 [==============================] - 0s 158us/sample - loss: 0.5596 -
accuracy: 0.7251
Epoch 42/50
342/342 [==============================] - 0s 158us/sample - loss: 0.5794 -
accuracy: 0.7018
Epoch 43/50
342/342 [==============================] - 0s 159us/sample - loss: 0.5850 -
accuracy: 0.7135
Epoch 44/50
342/342 [==============================] - 0s 156us/sample - loss: 0.5486 -
accuracy: 0.7398
Epoch 45/50
342/342 [==============================] - 0s 154us/sample - loss: 0.5539 -
accuracy: 0.7076
Epoch 46/50
342/342 [==============================] - 0s 158us/sample - loss: 0.5697 -
accuracy: 0.7047
Epoch 47/50
342/342 [==============================] - 0s 162us/sample - loss: 0.5621 -
accuracy: 0.7339
Epoch 48/50
342/342 [==============================] - 0s 189us/sample - loss: 0.5584 -
accuracy: 0.7310
Epoch 49/50
342/342 [==============================] - 0s 162us/sample - loss: 0.5517 -
accuracy: 0.7076
Epoch 50/50
```

```
342/342 [==============================] - 0s 159us/sample - loss: 0.5464 -
accuracy: 0.7076
171/1 [=========================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
```

```
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
=================] - 0s 1ms/sample - loss: 0.5211 - accuracy: 0.7953
Train on 342 samples
Epoch 1/50
342/342 [==============================] - 1s 4ms/sample - loss: 0.6455 -
accuracy: 0.6228
Epoch 2/50
342/342 [==============================] - 0s 159us/sample - loss: 0.6232 -
accuracy: 0.6930
Epoch 3/50
342/342 [==============================] - 0s 158us/sample - loss: 0.6106 -
accuracy: 0.6930
Epoch 4/50
342/342 [==============================] - 0s 159us/sample - loss: 0.6111 -
accuracy: 0.7047
Epoch 5/50
342/342 [==============================] - 0s 162us/sample - loss: 0.6277 -
accuracy: 0.6959
Epoch 6/50
342/342 [==============================] - 0s 163us/sample - loss: 0.6053 -
accuracy: 0.6930
Epoch 7/50
342/342 [==============================] - 0s 166us/sample - loss: 0.6015 -
accuracy: 0.6959
Epoch 8/50
342/342 [==============================] - 0s 166us/sample - loss: 0.6058 -
accuracy: 0.6988
Epoch 9/50
342/342 [==============================] - 0s 163us/sample - loss: 0.5948 -
accuracy: 0.7047
Epoch 10/50
```

```
342/342 [==============================] - 0s 160us/sample - loss: 0.6029 -
accuracy: 0.7105
Epoch 11/50
342/342 [==============================] - 0s 159us/sample - loss: 0.5949 -
accuracy: 0.6959
Epoch 12/50
342/342 [==============================] - 0s 164us/sample - loss: 0.6024 -
accuracy: 0.6988
Epoch 13/50
342/342 [==============================] - 0s 163us/sample - loss: 0.5850 -
accuracy: 0.6842
Epoch 14/50
342/342 [==============================] - 0s 168us/sample - loss: 0.5963 -
accuracy: 0.7076
Epoch 15/50
342/342 [==============================] - 0s 164us/sample - loss: 0.5854 -
accuracy: 0.6988
Epoch 16/50
342/342 [==============================] - 0s 167us/sample - loss: 0.5906 -
accuracy: 0.6959
Epoch 17/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5997 -
accuracy: 0.6901
Epoch 18/50
342/342 [==============================] - 0s 166us/sample - loss: 0.5985 -
accuracy: 0.6988
Epoch 19/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5741 -
accuracy: 0.7105
Epoch 20/50
342/342 [==============================] - 0s 166us/sample - loss: 0.6000 -
accuracy: 0.6988
Epoch 21/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5788 -
accuracy: 0.7047
Epoch 22/50
342/342 [==============================] - 0s 166us/sample - loss: 0.5689 -
accuracy: 0.7135
Epoch 23/50
342/342 [==============================] - 0s 167us/sample - loss: 0.5861 -
accuracy: 0.7018
Epoch 24/50
342/342 [==============================] - 0s 168us/sample - loss: 0.5761 -
accuracy: 0.6930
Epoch 25/50
342/342 [==============================] - 0s 164us/sample - loss: 0.5891 -
accuracy: 0.6930
Epoch 26/50
```

```
342/342 [==============================] - 0s 162us/sample - loss: 0.5733 -
accuracy: 0.7135
Epoch 27/50
342/342 [==============================] - 0s 164us/sample - loss: 0.5660 -
accuracy: 0.7135
Epoch 28/50
342/342 [==============================] - 0s 164us/sample - loss: 0.5715 -
accuracy: 0.7135
Epoch 29/50
342/342 [==============================] - 0s 161us/sample - loss: 0.5698 -
accuracy: 0.7135
Epoch 30/50
342/342 [==============================] - 0s 166us/sample - loss: 0.5602 -
accuracy: 0.7135
Epoch 31/50
342/342 [==============================] - 0s 164us/sample - loss: 0.5674 -
accuracy: 0.7105
Epoch 32/50
342/342 [==============================] - 0s 163us/sample - loss: 0.5549 -
accuracy: 0.7105
Epoch 33/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5605 -
accuracy: 0.7047
Epoch 34/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5755 -
accuracy: 0.7105
Epoch 35/50
342/342 [==============================] - 0s 163us/sample - loss: 0.5654 -
accuracy: 0.7193
Epoch 36/50
342/342 [==============================] - 0s 166us/sample - loss: 0.5648 -
accuracy: 0.7135
Epoch 37/50
342/342 [==============================] - 0s 167us/sample - loss: 0.5602 -
accuracy: 0.7164
Epoch 38/50
342/342 [==============================] - 0s 166us/sample - loss: 0.5654 -
accuracy: 0.7018
Epoch 39/50
342/342 [==============================] - 0s 168us/sample - loss: 0.5468 -
accuracy: 0.7135
Epoch 40/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5527 -
accuracy: 0.7135
Epoch 41/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5767 -
accuracy: 0.7105
Epoch 42/50
```

```
342/342 [==============================] - 0s 165us/sample - loss: 0.5608 -
accuracy: 0.7251
Epoch 43/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5516 -
accuracy: 0.7193
Epoch 44/50
342/342 [==============================] - 0s 168us/sample - loss: 0.5445 -
accuracy: 0.7310
Epoch 45/50
342/342 [==============================] - 0s 167us/sample - loss: 0.5533 -
accuracy: 0.7105
Epoch 46/50
342/342 [==============================] - 0s 169us/sample - loss: 0.5378 -
accuracy: 0.7222
Epoch 47/50
342/342 [==============================] - 0s 168us/sample - loss: 0.5566 -
accuracy: 0.7047
Epoch 48/50
342/342 [==============================] - 0s 172us/sample - loss: 0.5522 -
accuracy: 0.7222
Epoch 49/50
342/342 [==============================] - 0s 165us/sample - loss: 0.5476 -
accuracy: 0.7047
Epoch 50/50
342/342 [==============================] - 0s 170us/sample - loss: 0.5473 -
accuracy: 0.7339
171/1 [=========================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
```

```
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
=================] - 0s 2ms/sample - loss: 0.6557 - accuracy: 0.6140
Train on 342 samples
Epoch 1/50
342/342 [==============================] - 1s 4ms/sample - loss: 0.7115 -
accuracy: 0.4912
Epoch 2/50
```

```
342/342 [==============================] - 0s 160us/sample - loss: 0.6808 -
accuracy: 0.5789
Epoch 3/50
342/342 [==============================] - 0s 176us/sample - loss: 0.6673 -
accuracy: 0.6111
Epoch 4/50
342/342 [==============================] - 0s 165us/sample - loss: 0.6636 -
accuracy: 0.6023
Epoch 5/50
342/342 [==============================] - 0s 168us/sample - loss: 0.6619 -
accuracy: 0.6287
Epoch 6/50
342/342 [==============================] - 0s 169us/sample - loss: 0.6496 -
accuracy: 0.6579
Epoch 7/50
342/342 [==============================] - 0s 171us/sample - loss: 0.6438 -
accuracy: 0.6374
Epoch 8/50
342/342 [==============================] - 0s 172us/sample - loss: 0.6437 -
accuracy: 0.6199
Epoch 9/50
342/342 [==============================] - 0s 168us/sample - loss: 0.6558 -
accuracy: 0.6140
Epoch 10/50
342/342 [==============================] - 0s 166us/sample - loss: 0.6456 -
accuracy: 0.6374
Epoch 11/50
342/342 [==============================] - 0s 168us/sample - loss: 0.6566 -
accuracy: 0.6316
Epoch 12/50
342/342 [==============================] - 0s 170us/sample - loss: 0.6500 -
accuracy: 0.6462
Epoch 13/50
342/342 [==============================] - 0s 173us/sample - loss: 0.6476 -
accuracy: 0.6491
Epoch 14/50
342/342 [==============================] - 0s 172us/sample - loss: 0.6372 -
accuracy: 0.6462
Epoch 15/50
342/342 [==============================] - 0s 168us/sample - loss: 0.6411 -
accuracy: 0.6433
Epoch 16/50
342/342 [==============================] - 0s 171us/sample - loss: 0.6439 -
accuracy: 0.6433
Epoch 17/50
342/342 [==============================] - 0s 171us/sample - loss: 0.6529 -
accuracy: 0.6374
Epoch 18/50
```

```
342/342 [==============================] - 0s 173us/sample - loss: 0.6402 -
accuracy: 0.6374
Epoch 19/50
342/342 [==============================] - 0s 175us/sample - loss: 0.6515 -
accuracy: 0.6199
Epoch 20/50
342/342 [==============================] - 0s 172us/sample - loss: 0.6271 -
accuracy: 0.6784
Epoch 21/50
342/342 [==============================] - 0s 174us/sample - loss: 0.6423 -
accuracy: 0.6462
Epoch 22/50
342/342 [==============================] - 0s 171us/sample - loss: 0.6308 -
accuracy: 0.6608
Epoch 23/50
342/342 [==============================] - 0s 174us/sample - loss: 0.6304 -
accuracy: 0.6433
Epoch 24/50
342/342 [==============================] - 0s 175us/sample - loss: 0.6291 -
accuracy: 0.6608
Epoch 25/50
342/342 [==============================] - 0s 174us/sample - loss: 0.6307 -
accuracy: 0.6257
Epoch 26/50
342/342 [==============================] - 0s 174us/sample - loss: 0.6338 -
accuracy: 0.6257
Epoch 27/50
342/342 [==============================] - 0s 177us/sample - loss: 0.6273 -
accuracy: 0.6550
Epoch 28/50
342/342 [==============================] - 0s 178us/sample - loss: 0.6231 -
accuracy: 0.6901
Epoch 29/50
342/342 [==============================] - 0s 173us/sample - loss: 0.6197 -
accuracy: 0.6608
Epoch 30/50
342/342 [==============================] - 0s 169us/sample - loss: 0.6269 -
accuracy: 0.6520
Epoch 31/50
342/342 [==============================] - 0s 170us/sample - loss: 0.6204 -
accuracy: 0.6579
Epoch 32/50
342/342 [==============================] - 0s 171us/sample - loss: 0.6135 -
accuracy: 0.6725
Epoch 33/50
342/342 [==============================] - 0s 176us/sample - loss: 0.6097 -
accuracy: 0.6374
Epoch 34/50
```

```
342/342 [==============================] - 0s 174us/sample - loss: 0.6091 -
accuracy: 0.6696
Epoch 35/50
342/342 [==============================] - 0s 174us/sample - loss: 0.6212 -
accuracy: 0.6842
Epoch 36/50
342/342 [==============================] - 0s 174us/sample - loss: 0.6210 -
accuracy: 0.6462
Epoch 37/50
342/342 [==============================] - 0s 173us/sample - loss: 0.6192 -
accuracy: 0.6637
Epoch 38/50
342/342 [==============================] - 0s 176us/sample - loss: 0.6154 -
accuracy: 0.6608
Epoch 39/50
342/342 [==============================] - 0s 168us/sample - loss: 0.6025 -
accuracy: 0.6842
Epoch 40/50
342/342 [==============================] - 0s 176us/sample - loss: 0.6103 -
accuracy: 0.6579
Epoch 41/50
342/342 [==============================] - 0s 176us/sample - loss: 0.6029 -
accuracy: 0.6725
Epoch 42/50
342/342 [==============================] - 0s 173us/sample - loss: 0.6031 -
accuracy: 0.6871
Epoch 43/50
342/342 [==============================] - 0s 170us/sample - loss: 0.6247 -
accuracy: 0.6608
Epoch 44/50
342/342 [==============================] - 0s 173us/sample - loss: 0.6085 -
accuracy: 0.6754
Epoch 45/50
342/342 [==============================] - 0s 173us/sample - loss: 0.5932 -
accuracy: 0.6784
Epoch 46/50
342/342 [==============================] - 0s 173us/sample - loss: 0.6009 -
accuracy: 0.6842
Epoch 47/50
342/342 [==============================] - 0s 174us/sample - loss: 0.5953 -
accuracy: 0.6959
Epoch 48/50
342/342 [==============================] - 0s 172us/sample - loss: 0.5887 -
accuracy: 0.6754
Epoch 49/50
342/342 [==============================] - 0s 172us/sample - loss: 0.6095 -
accuracy: 0.6579
Epoch 50/50
```

```
342/342 [==============================] - 0s 175us/sample - loss: 0.6138 -
accuracy: 0.6901
171/1 [=========================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
```

```
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
================================================================================
=================] - 0s 2ms/sample - loss: 0.5577 - accuracy: 0.7135
Baseline Accuracy: 70.76% (7.41%)
```

[22]:

```
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

```
Train on 513 samples
Epoch 1/50
513/513 [==============================] - 1s 3ms/sample - loss: 0.6500 -
accuracy: 0.6569
Epoch 2/50
513/513 [==============================] - 0s 155us/sample - loss: 0.6509 -
accuracy: 0.6550
Epoch 3/50
513/513 [==============================] - 0s 155us/sample - loss: 0.6274 -
accuracy: 0.6608
Epoch 4/50
513/513 [==============================] - 0s 152us/sample - loss: 0.6355 -
accuracy: 0.6686
Epoch 5/50
513/513 [==============================] - 0s 158us/sample - loss: 0.6343 -
accuracy: 0.6550
Epoch 6/50
513/513 [==============================] - 0s 158us/sample - loss: 0.6189 -
accuracy: 0.6608
Epoch 7/50
513/513 [==============================] - 0s 157us/sample - loss: 0.6163 -
accuracy: 0.6706
Epoch 8/50
513/513 [==============================] - 0s 158us/sample - loss: 0.6046 -
```

```
accuracy: 0.6608
Epoch 9/50
513/513 [==============================] - 0s 159us/sample - loss: 0.6062 -
accuracy: 0.6764
Epoch 10/50
513/513 [==============================] - 0s 156us/sample - loss: 0.6211 -
accuracy: 0.6608
Epoch 11/50
513/513 [==============================] - 0s 158us/sample - loss: 0.6253 -
accuracy: 0.6667
Epoch 12/50
513/513 [==============================] - 0s 158us/sample - loss: 0.6205 -
accuracy: 0.6764
Epoch 13/50
513/513 [==============================] - 0s 159us/sample - loss: 0.6112 -
accuracy: 0.6628
Epoch 14/50
513/513 [==============================] - 0s 157us/sample - loss: 0.6081 -
accuracy: 0.6472
Epoch 15/50
513/513 [==============================] - 0s 158us/sample - loss: 0.6206 -
accuracy: 0.6472
Epoch 16/50
513/513 [==============================] - 0s 158us/sample - loss: 0.6043 -
accuracy: 0.6647
Epoch 17/50
513/513 [==============================] - 0s 163us/sample - loss: 0.6078 -
accuracy: 0.6511
Epoch 18/50
513/513 [==============================] - 0s 157us/sample - loss: 0.6098 -
accuracy: 0.6686
Epoch 19/50
513/513 [==============================] - 0s 154us/sample - loss: 0.6111 -
accuracy: 0.6569
Epoch 20/50
513/513 [==============================] - 0s 157us/sample - loss: 0.6265 -
accuracy: 0.6608
Epoch 21/50
513/513 [==============================] - 0s 159us/sample - loss: 0.6070 -
accuracy: 0.6628
Epoch 22/50
513/513 [==============================] - 0s 159us/sample - loss: 0.6006 -
accuracy: 0.6608
Epoch 23/50
513/513 [==============================] - 0s 157us/sample - loss: 0.6152 -
accuracy: 0.6803
Epoch 24/50
513/513 [==============================] - 0s 158us/sample - loss: 0.6085 -
```

```
accuracy: 0.6667
Epoch 25/50
513/513 [==============================] - 0s 157us/sample - loss: 0.6121 -
accuracy: 0.6725
Epoch 26/50
513/513 [==============================] - 0s 156us/sample - loss: 0.6016 -
accuracy: 0.6764
Epoch 27/50
513/513 [==============================] - 0s 161us/sample - loss: 0.5964 -
accuracy: 0.6647
Epoch 28/50
513/513 [==============================] - 0s 157us/sample - loss: 0.6053 -
accuracy: 0.6686
Epoch 29/50
513/513 [==============================] - 0s 156us/sample - loss: 0.6097 -
accuracy: 0.6686
Epoch 30/50
513/513 [==============================] - 0s 158us/sample - loss: 0.5960 -
accuracy: 0.6745
Epoch 31/50
513/513 [==============================] - 0s 164us/sample - loss: 0.5819 -
accuracy: 0.7096
Epoch 32/50
513/513 [==============================] - 0s 163us/sample - loss: 0.6078 -
accuracy: 0.6550
Epoch 33/50
513/513 [==============================] - 0s 158us/sample - loss: 0.5974 -
accuracy: 0.6706
Epoch 34/50
513/513 [==============================] - 0s 164us/sample - loss: 0.6039 -
accuracy: 0.6706
Epoch 35/50
513/513 [==============================] - 0s 163us/sample - loss: 0.5869 -
accuracy: 0.7057
Epoch 36/50
513/513 [==============================] - 0s 162us/sample - loss: 0.6008 -
accuracy: 0.7037
Epoch 37/50
513/513 [==============================] - 0s 162us/sample - loss: 0.6008 -
accuracy: 0.6706
Epoch 38/50
513/513 [==============================] - 0s 164us/sample - loss: 0.5836 -
accuracy: 0.6940
Epoch 39/50
513/513 [==============================] - 0s 165us/sample - loss: 0.5951 -
accuracy: 0.6901
Epoch 40/50
513/513 [==============================] - 0s 164us/sample - loss: 0.5777 -
```

```
accuracy: 0.6920
Epoch 41/50
513/513 [==============================] - 0s 158us/sample - loss: 0.5922 -
accuracy: 0.6998
Epoch 42/50
513/513 [==============================] - 0s 159us/sample - loss: 0.5819 -
accuracy: 0.6920
Epoch 43/50
513/513 [==============================] - 0s 161us/sample - loss: 0.5706 -
accuracy: 0.6920
Epoch 44/50
513/513 [==============================] - 0s 162us/sample - loss: 0.5829 -
accuracy: 0.6803
Epoch 45/50
513/513 [==============================] - 0s 161us/sample - loss: 0.5805 -
accuracy: 0.7018
Epoch 46/50
513/513 [==============================] - 0s 165us/sample - loss: 0.6111 -
accuracy: 0.6550
Epoch 47/50
513/513 [==============================] - 0s 165us/sample - loss: 0.5792 -
accuracy: 0.6823
Epoch 48/50
513/513 [==============================] - 0s 163us/sample - loss: 0.5857 -
accuracy: 0.7135
Epoch 49/50
513/513 [==============================] - 0s 165us/sample - loss: 0.5988 -
accuracy: 0.6823
Epoch 50/50
513/513 [==============================] - 0s 166us/sample - loss: 0.5842 -
accuracy: 0.6881
```

[23]:
```
print(classification_report(y_test, test_predictions))
print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           1       0.65      0.95      0.77        75
           2       0.79      0.28      0.41        54

    accuracy                           0.67       129
   macro avg       0.72      0.61      0.59       129
weighted avg       0.71      0.67      0.62       129

              precision    recall  f1-score   support

           1       0.63      0.97      0.77        75
           2       0.86      0.22      0.35        54
```

```
    accuracy                           0.66       129
   macro avg       0.75      0.60      0.56       129
weighted avg       0.73      0.66      0.59       129
```

[26]:
```
print(confusion_matrix(y_test,test_predictions))
```

```
[[71  4]
 [39 15]]
```

[21]:
```
model.model.save('models/model_1.h5')
```

[40]:
```

```