Hierarchical_classifier

March 31, 2020

[1]:
```
import tensorflow as tf
import tensorflow_quantum as tfq
import cirq
from cirq import Circuit
from cirq.devices import GridQubit
from cirq import Simulator
import numpy as np
import sympy as sp
from numpy import genfromtxt

c = genfromtxt('data.csv', delimiter=',')
# print(c.shape)
np.random.shuffle(c)
x_train=c[:,:4]
print(x_train.shape)
x_train_n = np.pi*((x_train - x_train.min(0)) / 2*x_train.ptp(0))
# x_test_n = np.pi*((x_test - x_test.min(0)) / 2*x_test.ptp(0))

x_train_n = x_train_n[:100,:]
x_test_n= x_train_n[80:100,:]

# print(x_train_n.max())
# print(x_train_n.min())
# exit()

y_train=c[:100,4]
y_test = c[80:100,4]
# y_train=y_train*2-1
# print(y_train)
# exit()
# print(y_train.shape)
# y_train_cat = tf.keras.utils.to_categorical(y_train)
# y_test = tf.keras.utils.to_categorical(y_test)
# print(y_train)
# exit()
# y_train=y_train*2-1
```

```
def convert_to_circuit(values):
        qubits = cirq.GridQubit.rect(1,4)
        circuit = cirq.Circuit()
        for i, value in enumerate(values):
                rot = cirq.ry(value*2/np.pi)
                circuit.append(rot(qubits[i]))
                # circuit.append(cirq.X(qubits[i]))

        # print(circuit)
        return circuit

x_train_cirq = [convert_to_circuit(x) for x in x_train_n]
x_test_cirq = [convert_to_circuit(x) for x in x_test_n]
# print(x_train_cirq)
x_train_tf_circ = tfq.convert_to_tensor(x_train_cirq)
x_test_tf_circ = tfq.convert_to_tensor(x_test_cirq)
# print(x_test_tf_circ)

def one_bit_unitary_new(circuit, symbols, bit):
        circuit.append(cirq.X(bit)**symbols[0])
        circuit.append(cirq.Y(bit)**symbols[1])
        circuit.append(cirq.Z(bit)**symbols[2])


def one_bit_unitary(bit, symbols):
        rot = cirq.ry(symbols)
        return rot(bit)

def create_model_new():
        data_qubit = cirq.GridQubit.rect(1,4)
        readout = cirq.GridQubit(-1,-1)
        circuit = cirq.Circuit()
        symbols = sp.symbols('x0:100')
        k=0
        for i, bit in enumerate(data_qubit):
                one_bit_unitary_new(circuit, [symbols[k], symbols[k+1],␣
 ↪symbols[k+2]],bit)
                k=k+3
        circuit.append(cirq.CNOT(data_qubit[0], data_qubit[1]))
        circuit.append(cirq.CNOT(data_qubit[3], data_qubit[2]))
        one_bit_unitary_new(circuit, [symbols[k], symbols[k+1],␣
 ↪symbols[k+2]],data_qubit[1])
        k=k+3
        one_bit_unitary_new(circuit, [symbols[k], symbols[k+1],␣
 ↪symbols[k+2]],data_qubit[2])
        k=k+3
        circuit.append(cirq.CNOT(data_qubit[1], data_qubit[2]))
```

```
        one_bit_unitary_new(circuit, [symbols[k], symbols[k+1],␣
↪symbols[k+2]],data_qubit[2])
        k=k+3
        print (circuit)
        return circuit, cirq.Z(data_qubit[2])
model_new, readout_new = create_model_new()
model = tf.keras.Sequential([
                tf.keras.layers.Input(shape=(), dtype=tf.string),
                tfq.layers.PQC(model_new, readout_new),
                tf.keras.layers.Activation('sigmoid')
                ])
model.summary()
sgd = tf.keras.optimizers.SGD(lr=0.01, decay=1e-6, momentum=0.75, nesterov=True)
def hinge_accuracy(y_true, y_pred):
    y_true = tf.squeeze(y_true) > 0.0
    y_pred = tf.squeeze(y_pred) > 0.0
    result = tf.cast(y_true == y_pred, tf.float32)

    return tf.reduce_mean(result)
model.compile(loss = 'binary_crossentropy',#tf.keras.losses.Hinge(),
                        optimizer = 'adam',
                        metrics=['accuracy'])
model.fit(x_train_tf_circ, y_train,shuffle=True, batch_size = 32,epochs=100,␣
↪validation_split=0.2)
scores = model.evaluate(x_train_tf_circ, y_train, verbose=0)
scores3 = model.evaluate(x_test_tf_circ, y_test, verbose=0)
print('Accuracy on training data: {} \n Error on training data: {}'.
↪format(scores[1], 1 - scores[1]))
print('Accuracy on test data: {} \n Error on test data: {}'.format(scores3[1],␣
↪1 - scores3[1]))


model2 = tf.keras.Sequential([
  tf.keras.layers.Dense(4, activation=tf.nn.relu, input_shape=x_train_n[0].
↪shape),
  tf.keras.layers.Dense(1, activation='sigmoid')
])
model2.summary()
model2.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])
model2.fit(x_train_n, y_train, batch_size=32, validation_split=0.2, epochs=100)
scores2 = model2.evaluate(x_train_n, y_train, verbose=0)
scores4 = model2.evaluate(x_test_n, y_test, verbose=0)
print('Classical ML: Accuracy on training data: {} \n Error on training data:␣
↪{}'.format(scores2[1], 1 - scores2[1]))
```

```
print('Quantum ML: Accuracy on training data: {} \n Error on training data: {}'.
 ↪format(scores[1], 1 - scores[1]))
print('Classical ML: Accuracy on test data: {} \n Error on test data: {}'.
 ↪format(scores4[1], 1 - scores4[1]))
print('Quantm ML: Accuracy on test data: {} \n Error on test data: {}'.
 ↪format(scores3[1], 1 - scores3[1]))
```

```
(1372, 4)
(0, 0):  X^x0  Y^x1  Z^x2  @


(0, 1):  X^x3  Y^x4  Z^x5  X  X^x12  Y^x13  Z^x14  @


(0, 2):  X^x6  Y^x7  Z^x8  X  X^x15  Y^x16  Z^x17  X  X^x18  Y^x19
 Z^x20

(0, 3):  X^x9  Y^x10  Z^x11  @

Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
pqc (PQC)                    (None, 1)                 21
_________________________________________________________________
activation (Activation)      (None, 1)                 0
=================================================================
Total params: 21
Trainable params: 21
Non-trainable params: 0
_________________________________________________________________
Train on 80 samples, validate on 20 samples
Epoch 1/100
80/80 [==============================] - 3s 39ms/sample - loss: 0.6898 -
accuracy: 0.6000 - val_loss: 0.7058 - val_accuracy: 0.5000
Epoch 2/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6888 -
accuracy: 0.6125 - val_loss: 0.7055 - val_accuracy: 0.5000
Epoch 3/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6880 -
accuracy: 0.6125 - val_loss: 0.7049 - val_accuracy: 0.5000
Epoch 4/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6874 -
accuracy: 0.6125 - val_loss: 0.7044 - val_accuracy: 0.5000
Epoch 5/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6867 -
accuracy: 0.6125 - val_loss: 0.7039 - val_accuracy: 0.5000
```

```
Epoch 6/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6862 -
accuracy: 0.6125 - val_loss: 0.7033 - val_accuracy: 0.5000
Epoch 7/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6855 -
accuracy: 0.6125 - val_loss: 0.7028 - val_accuracy: 0.5000
Epoch 8/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6848 -
accuracy: 0.6125 - val_loss: 0.7023 - val_accuracy: 0.5000
Epoch 9/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6843 -
accuracy: 0.6000 - val_loss: 0.7018 - val_accuracy: 0.5000
Epoch 10/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6837 -
accuracy: 0.6125 - val_loss: 0.7013 - val_accuracy: 0.5000
Epoch 11/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6830 -
accuracy: 0.6125 - val_loss: 0.7010 - val_accuracy: 0.5000
Epoch 12/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6827 -
accuracy: 0.6125 - val_loss: 0.7005 - val_accuracy: 0.4500
Epoch 13/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6820 -
accuracy: 0.6250 - val_loss: 0.7002 - val_accuracy: 0.4500
Epoch 14/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6815 -
accuracy: 0.6250 - val_loss: 0.6999 - val_accuracy: 0.4500
Epoch 15/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6810 -
accuracy: 0.6250 - val_loss: 0.6995 - val_accuracy: 0.5000
Epoch 16/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6805 -
accuracy: 0.6250 - val_loss: 0.6994 - val_accuracy: 0.5000
Epoch 17/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6801 -
accuracy: 0.6250 - val_loss: 0.6992 - val_accuracy: 0.5000
Epoch 18/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6796 -
accuracy: 0.6250 - val_loss: 0.6991 - val_accuracy: 0.5000
Epoch 19/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6792 -
accuracy: 0.6250 - val_loss: 0.6991 - val_accuracy: 0.5000
Epoch 20/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6788 -
accuracy: 0.6125 - val_loss: 0.6991 - val_accuracy: 0.5000
Epoch 21/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6783 -
accuracy: 0.6125 - val_loss: 0.6990 - val_accuracy: 0.5000
```

```
Epoch 22/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6779 -
accuracy: 0.6125 - val_loss: 0.6991 - val_accuracy: 0.5000
Epoch 23/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6774 -
accuracy: 0.6125 - val_loss: 0.6991 - val_accuracy: 0.5000
Epoch 24/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6771 -
accuracy: 0.6125 - val_loss: 0.6990 - val_accuracy: 0.5500
Epoch 25/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6767 -
accuracy: 0.6125 - val_loss: 0.6989 - val_accuracy: 0.5500
Epoch 26/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6764 -
accuracy: 0.6125 - val_loss: 0.6987 - val_accuracy: 0.5500
Epoch 27/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6760 -
accuracy: 0.6125 - val_loss: 0.6986 - val_accuracy: 0.5500
Epoch 28/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6756 -
accuracy: 0.6125 - val_loss: 0.6983 - val_accuracy: 0.5500
Epoch 29/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6753 -
accuracy: 0.6125 - val_loss: 0.6982 - val_accuracy: 0.5500
Epoch 30/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6749 -
accuracy: 0.6000 - val_loss: 0.6981 - val_accuracy: 0.5500
Epoch 31/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6746 -
accuracy: 0.6000 - val_loss: 0.6980 - val_accuracy: 0.5500
Epoch 32/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6742 -
accuracy: 0.6000 - val_loss: 0.6977 - val_accuracy: 0.5500
Epoch 33/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6739 -
accuracy: 0.6000 - val_loss: 0.6976 - val_accuracy: 0.5500
Epoch 34/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6736 -
accuracy: 0.6000 - val_loss: 0.6975 - val_accuracy: 0.5500
Epoch 35/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6733 -
accuracy: 0.6000 - val_loss: 0.6974 - val_accuracy: 0.5500
Epoch 36/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6730 -
accuracy: 0.6000 - val_loss: 0.6974 - val_accuracy: 0.5500
Epoch 37/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6727 -
accuracy: 0.5875 - val_loss: 0.6974 - val_accuracy: 0.5500
```

```
Epoch 38/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6724 -
accuracy: 0.5875 - val_loss: 0.6973 - val_accuracy: 0.5500
Epoch 39/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6721 -
accuracy: 0.5875 - val_loss: 0.6972 - val_accuracy: 0.5500
Epoch 40/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6719 -
accuracy: 0.5875 - val_loss: 0.6970 - val_accuracy: 0.5500
Epoch 41/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6716 -
accuracy: 0.5875 - val_loss: 0.6968 - val_accuracy: 0.5500
Epoch 42/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6714 -
accuracy: 0.5875 - val_loss: 0.6967 - val_accuracy: 0.5500
Epoch 43/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6711 -
accuracy: 0.5875 - val_loss: 0.6965 - val_accuracy: 0.5500
Epoch 44/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6709 -
accuracy: 0.5875 - val_loss: 0.6963 - val_accuracy: 0.5500
Epoch 45/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6706 -
accuracy: 0.5875 - val_loss: 0.6961 - val_accuracy: 0.5500
Epoch 46/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6705 -
accuracy: 0.5875 - val_loss: 0.6961 - val_accuracy: 0.5500
Epoch 47/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6702 -
accuracy: 0.5875 - val_loss: 0.6961 - val_accuracy: 0.5500
Epoch 48/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6700 -
accuracy: 0.5875 - val_loss: 0.6962 - val_accuracy: 0.5500
Epoch 49/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6698 -
accuracy: 0.6000 - val_loss: 0.6962 - val_accuracy: 0.5500
Epoch 50/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6696 -
accuracy: 0.6000 - val_loss: 0.6961 - val_accuracy: 0.5500
Epoch 51/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6693 -
accuracy: 0.6000 - val_loss: 0.6959 - val_accuracy: 0.6000
Epoch 52/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6692 -
accuracy: 0.6000 - val_loss: 0.6960 - val_accuracy: 0.6000
Epoch 53/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6689 -
accuracy: 0.6000 - val_loss: 0.6960 - val_accuracy: 0.6000
```

```
Epoch 54/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6687 -
accuracy: 0.6000 - val_loss: 0.6961 - val_accuracy: 0.6000
Epoch 55/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6685 -
accuracy: 0.6000 - val_loss: 0.6962 - val_accuracy: 0.6000
Epoch 56/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6684 -
accuracy: 0.6000 - val_loss: 0.6965 - val_accuracy: 0.6000
Epoch 57/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6681 -
accuracy: 0.6000 - val_loss: 0.6968 - val_accuracy: 0.6000
Epoch 58/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6681 -
accuracy: 0.6000 - val_loss: 0.6970 - val_accuracy: 0.6000
Epoch 59/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6678 -
accuracy: 0.6000 - val_loss: 0.6970 - val_accuracy: 0.6000
Epoch 60/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6676 -
accuracy: 0.6000 - val_loss: 0.6970 - val_accuracy: 0.6000
Epoch 61/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6675 -
accuracy: 0.6000 - val_loss: 0.6970 - val_accuracy: 0.6000
Epoch 62/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6673 -
accuracy: 0.6000 - val_loss: 0.6970 - val_accuracy: 0.6000
Epoch 63/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6672 -
accuracy: 0.6000 - val_loss: 0.6970 - val_accuracy: 0.6000
Epoch 64/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6671 -
accuracy: 0.6000 - val_loss: 0.6972 - val_accuracy: 0.6000
Epoch 65/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6669 -
accuracy: 0.6000 - val_loss: 0.6973 - val_accuracy: 0.6000
Epoch 66/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6668 -
accuracy: 0.6000 - val_loss: 0.6973 - val_accuracy: 0.6000
Epoch 67/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6666 -
accuracy: 0.6000 - val_loss: 0.6974 - val_accuracy: 0.6000
Epoch 68/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6665 -
accuracy: 0.6000 - val_loss: 0.6977 - val_accuracy: 0.6000
Epoch 69/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6663 -
accuracy: 0.6000 - val_loss: 0.6978 - val_accuracy: 0.6000
```

```
Epoch 70/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6662 -
accuracy: 0.6000 - val_loss: 0.6981 - val_accuracy: 0.6000
Epoch 71/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6660 -
accuracy: 0.6000 - val_loss: 0.6983 - val_accuracy: 0.6000
Epoch 72/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6659 -
accuracy: 0.6000 - val_loss: 0.6986 - val_accuracy: 0.6000
Epoch 73/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6657 -
accuracy: 0.6000 - val_loss: 0.6987 - val_accuracy: 0.6000
Epoch 74/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6656 -
accuracy: 0.6000 - val_loss: 0.6988 - val_accuracy: 0.6000
Epoch 75/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6654 -
accuracy: 0.6000 - val_loss: 0.6988 - val_accuracy: 0.6000
Epoch 76/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6653 -
accuracy: 0.6000 - val_loss: 0.6987 - val_accuracy: 0.6000
Epoch 77/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6652 -
accuracy: 0.6000 - val_loss: 0.6986 - val_accuracy: 0.6000
Epoch 78/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6650 -
accuracy: 0.6000 - val_loss: 0.6988 - val_accuracy: 0.6000
Epoch 79/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6649 -
accuracy: 0.6000 - val_loss: 0.6990 - val_accuracy: 0.6000
Epoch 80/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6648 -
accuracy: 0.6000 - val_loss: 0.6993 - val_accuracy: 0.6000
Epoch 81/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6646 -
accuracy: 0.6000 - val_loss: 0.6998 - val_accuracy: 0.6000
Epoch 82/100
80/80 [==============================] - 0s 4ms/sample - loss: 0.6645 -
accuracy: 0.5875 - val_loss: 0.6998 - val_accuracy: 0.6000
Epoch 83/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6643 -
accuracy: 0.5875 - val_loss: 0.7001 - val_accuracy: 0.6000
Epoch 84/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6642 -
accuracy: 0.5875 - val_loss: 0.7004 - val_accuracy: 0.6000
Epoch 85/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6640 -
accuracy: 0.5875 - val_loss: 0.7006 - val_accuracy: 0.6000
```

```
Epoch 86/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6639 -
accuracy: 0.5875 - val_loss: 0.7009 - val_accuracy: 0.6000
Epoch 87/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6638 -
accuracy: 0.5875 - val_loss: 0.7012 - val_accuracy: 0.5500
Epoch 88/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6636 -
accuracy: 0.5875 - val_loss: 0.7013 - val_accuracy: 0.5500
Epoch 89/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6635 -
accuracy: 0.5875 - val_loss: 0.7015 - val_accuracy: 0.5500
Epoch 90/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6634 -
accuracy: 0.5875 - val_loss: 0.7017 - val_accuracy: 0.5500
Epoch 91/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6633 -
accuracy: 0.5875 - val_loss: 0.7019 - val_accuracy: 0.5500
Epoch 92/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6632 -
accuracy: 0.5875 - val_loss: 0.7020 - val_accuracy: 0.5500
Epoch 93/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6630 -
accuracy: 0.5875 - val_loss: 0.7023 - val_accuracy: 0.5500
Epoch 94/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6629 -
accuracy: 0.5875 - val_loss: 0.7027 - val_accuracy: 0.5500
Epoch 95/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6628 -
accuracy: 0.5875 - val_loss: 0.7031 - val_accuracy: 0.5500
Epoch 96/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6627 -
accuracy: 0.5875 - val_loss: 0.7035 - val_accuracy: 0.5500
Epoch 97/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6626 -
accuracy: 0.5875 - val_loss: 0.7037 - val_accuracy: 0.5500
Epoch 98/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6624 -
accuracy: 0.5875 - val_loss: 0.7038 - val_accuracy: 0.5500
Epoch 99/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6622 -
accuracy: 0.5875 - val_loss: 0.7040 - val_accuracy: 0.5500
Epoch 100/100
80/80 [==============================] - 0s 3ms/sample - loss: 0.6621 -
accuracy: 0.5875 - val_loss: 0.7043 - val_accuracy: 0.5500
Accuracy on training data: 0.5799999833106995
 Error on training data: 0.42000001668930054
Accuracy on test data: 0.550000011920929
```

```
 Error on test data: 0.44999998807907104
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 4)                 20
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 5
=================================================================
Total params: 25
Trainable params: 25
Non-trainable params: 0
_________________________________________________________________
Train on 80 samples, validate on 20 samples
Epoch 1/100
80/80 [==============================] - 1s 7ms/sample - loss: 111.2401 -
accuracy: 0.2500 - val_loss: 104.9736 - val_accuracy: 0.3000
Epoch 2/100
80/80 [==============================] - 0s 172us/sample - loss: 106.6937 -
accuracy: 0.2500 - val_loss: 100.8001 - val_accuracy: 0.3000
Epoch 3/100
80/80 [==============================] - 0s 333us/sample - loss: 102.0109 -
accuracy: 0.2500 - val_loss: 96.6206 - val_accuracy: 0.3000
Epoch 4/100
80/80 [==============================] - 0s 160us/sample - loss: 97.3480 -
accuracy: 0.2500 - val_loss: 92.4358 - val_accuracy: 0.3000
Epoch 5/100
80/80 [==============================] - 0s 210us/sample - loss: 92.6756 -
accuracy: 0.2375 - val_loss: 88.2694 - val_accuracy: 0.3000
Epoch 6/100
80/80 [==============================] - 0s 266us/sample - loss: 87.8693 -
accuracy: 0.2375 - val_loss: 84.1563 - val_accuracy: 0.2500
Epoch 7/100
80/80 [==============================] - 0s 272us/sample - loss: 83.2241 -
accuracy: 0.2250 - val_loss: 80.1794 - val_accuracy: 0.2500
Epoch 8/100
80/80 [==============================] - 0s 235us/sample - loss: 78.9299 -
accuracy: 0.2125 - val_loss: 76.2142 - val_accuracy: 0.2500
Epoch 9/100
80/80 [==============================] - 0s 259us/sample - loss: 74.3027 -
accuracy: 0.2250 - val_loss: 72.3200 - val_accuracy: 0.2500
Epoch 10/100
80/80 [==============================] - 0s 317us/sample - loss: 69.9056 -
accuracy: 0.2250 - val_loss: 68.4708 - val_accuracy: 0.2500
Epoch 11/100
80/80 [==============================] - 0s 330us/sample - loss: 66.0189 -
accuracy: 0.2125 - val_loss: 64.5882 - val_accuracy: 0.2500
Epoch 12/100
```

```
80/80 [==============================] - 0s 351us/sample - loss: 62.5310 -
accuracy: 0.2250 - val_loss: 60.9226 - val_accuracy: 0.3000
Epoch 13/100
80/80 [==============================] - 0s 260us/sample - loss: 59.0175 -
accuracy: 0.2375 - val_loss: 57.5984 - val_accuracy: 0.3000
Epoch 14/100
80/80 [==============================] - 0s 317us/sample - loss: 55.7190 -
accuracy: 0.2375 - val_loss: 54.3249 - val_accuracy: 0.3000
Epoch 15/100
80/80 [==============================] - 0s 250us/sample - loss: 52.2057 -
accuracy: 0.2375 - val_loss: 51.2086 - val_accuracy: 0.2500
Epoch 16/100
80/80 [==============================] - 0s 224us/sample - loss: 48.9647 -
accuracy: 0.2500 - val_loss: 48.5659 - val_accuracy: 0.2000
Epoch 17/100
80/80 [==============================] - 0s 275us/sample - loss: 46.0978 -
accuracy: 0.2500 - val_loss: 46.1645 - val_accuracy: 0.1500
Epoch 18/100
80/80 [==============================] - 0s 270us/sample - loss: 42.4676 -
accuracy: 0.2500 - val_loss: 43.9320 - val_accuracy: 0.2000
Epoch 19/100
80/80 [==============================] - 0s 333us/sample - loss: 39.6435 -
accuracy: 0.2875 - val_loss: 42.0095 - val_accuracy: 0.2000
Epoch 20/100
80/80 [==============================] - 0s 292us/sample - loss: 36.9422 -
accuracy: 0.3000 - val_loss: 40.4298 - val_accuracy: 0.2500
Epoch 21/100
80/80 [==============================] - 0s 373us/sample - loss: 34.3082 -
accuracy: 0.2875 - val_loss: 39.0744 - val_accuracy: 0.2500
Epoch 22/100
80/80 [==============================] - 0s 244us/sample - loss: 32.2587 -
accuracy: 0.2875 - val_loss: 37.7597 - val_accuracy: 0.2500
Epoch 23/100
80/80 [==============================] - 0s 330us/sample - loss: 30.4026 -
accuracy: 0.3000 - val_loss: 36.5192 - val_accuracy: 0.2500
Epoch 24/100
80/80 [==============================] - 0s 212us/sample - loss: 28.7368 -
accuracy: 0.3125 - val_loss: 35.5001 - val_accuracy: 0.2000
Epoch 25/100
80/80 [==============================] - 0s 449us/sample - loss: 27.8195 -
accuracy: 0.3125 - val_loss: 34.7255 - val_accuracy: 0.2500
Epoch 26/100
80/80 [==============================] - 0s 254us/sample - loss: 26.7102 -
accuracy: 0.3125 - val_loss: 34.2211 - val_accuracy: 0.2500
Epoch 27/100
80/80 [==============================] - 0s 367us/sample - loss: 25.6259 -
accuracy: 0.3125 - val_loss: 33.7652 - val_accuracy: 0.2500
Epoch 28/100
```

```
80/80 [==============================] - 0s 264us/sample - loss: 24.9841 -
accuracy: 0.3250 - val_loss: 33.3367 - val_accuracy: 0.2500
Epoch 29/100
80/80 [==============================] - 0s 348us/sample - loss: 24.4909 -
accuracy: 0.3500 - val_loss: 32.9440 - val_accuracy: 0.2500
Epoch 30/100
80/80 [==============================] - 0s 303us/sample - loss: 24.1416 -
accuracy: 0.3500 - val_loss: 32.5917 - val_accuracy: 0.2500
Epoch 31/100
80/80 [==============================] - 0s 335us/sample - loss: 23.7846 -
accuracy: 0.3625 - val_loss: 32.3490 - val_accuracy: 0.3000
Epoch 32/100
80/80 [==============================] - 0s 310us/sample - loss: 23.5566 -
accuracy: 0.3375 - val_loss: 32.1744 - val_accuracy: 0.3000
Epoch 33/100
80/80 [==============================] - 0s 430us/sample - loss: 23.2625 -
accuracy: 0.3500 - val_loss: 32.0155 - val_accuracy: 0.3000
Epoch 34/100
80/80 [==============================] - 0s 341us/sample - loss: 23.1314 -
accuracy: 0.3500 - val_loss: 31.8585 - val_accuracy: 0.3000
Epoch 35/100
80/80 [==============================] - 0s 439us/sample - loss: 22.9821 -
accuracy: 0.3750 - val_loss: 31.7044 - val_accuracy: 0.3000
Epoch 36/100
80/80 [==============================] - 0s 406us/sample - loss: 22.8592 -
accuracy: 0.3750 - val_loss: 31.5588 - val_accuracy: 0.3000
Epoch 37/100
80/80 [==============================] - 0s 343us/sample - loss: 22.7257 -
accuracy: 0.3750 - val_loss: 31.4139 - val_accuracy: 0.3000
Epoch 38/100
80/80 [==============================] - 0s 299us/sample - loss: 22.5728 -
accuracy: 0.3750 - val_loss: 31.2743 - val_accuracy: 0.3000
Epoch 39/100
80/80 [==============================] - 0s 337us/sample - loss: 22.4565 -
accuracy: 0.3750 - val_loss: 31.1395 - val_accuracy: 0.3000
Epoch 40/100
80/80 [==============================] - 0s 335us/sample - loss: 22.3244 -
accuracy: 0.3750 - val_loss: 31.0014 - val_accuracy: 0.3000
Epoch 41/100
80/80 [==============================] - 0s 349us/sample - loss: 22.1948 -
accuracy: 0.3750 - val_loss: 30.8678 - val_accuracy: 0.3000
Epoch 42/100
80/80 [==============================] - 0s 385us/sample - loss: 22.0758 -
accuracy: 0.3750 - val_loss: 30.7345 - val_accuracy: 0.3000
Epoch 43/100
80/80 [==============================] - 0s 289us/sample - loss: 21.9429 -
accuracy: 0.3750 - val_loss: 30.6033 - val_accuracy: 0.3000
Epoch 44/100
```

```
80/80 [==============================] - 0s 418us/sample - loss: 21.8332 -
accuracy: 0.3750 - val_loss: 30.4715 - val_accuracy: 0.3000
Epoch 45/100
80/80 [==============================] - 0s 261us/sample - loss: 21.7120 -
accuracy: 0.3750 - val_loss: 30.3291 - val_accuracy: 0.3000
Epoch 46/100
80/80 [==============================] - 0s 219us/sample - loss: 21.5874 -
accuracy: 0.3750 - val_loss: 30.1918 - val_accuracy: 0.3000
Epoch 47/100
80/80 [==============================] - 0s 262us/sample - loss: 21.4631 -
accuracy: 0.3750 - val_loss: 30.0553 - val_accuracy: 0.3000
Epoch 48/100
80/80 [==============================] - 0s 303us/sample - loss: 21.3605 -
accuracy: 0.3750 - val_loss: 29.9203 - val_accuracy: 0.3000
Epoch 49/100
80/80 [==============================] - 0s 261us/sample - loss: 21.2438 -
accuracy: 0.3750 - val_loss: 29.7810 - val_accuracy: 0.3000
Epoch 50/100
80/80 [==============================] - 0s 271us/sample - loss: 21.1184 -
accuracy: 0.3750 - val_loss: 29.6481 - val_accuracy: 0.3000
Epoch 51/100
80/80 [==============================] - 0s 323us/sample - loss: 20.9984 -
accuracy: 0.3750 - val_loss: 29.5188 - val_accuracy: 0.3000
Epoch 52/100
80/80 [==============================] - 0s 370us/sample - loss: 20.8812 -
accuracy: 0.3750 - val_loss: 29.3916 - val_accuracy: 0.3000
Epoch 53/100
80/80 [==============================] - 0s 273us/sample - loss: 20.7740 -
accuracy: 0.3750 - val_loss: 29.2664 - val_accuracy: 0.3000
Epoch 54/100
80/80 [==============================] - 0s 307us/sample - loss: 20.6549 -
accuracy: 0.3750 - val_loss: 29.1426 - val_accuracy: 0.3000
Epoch 55/100
80/80 [==============================] - 0s 238us/sample - loss: 20.5377 -
accuracy: 0.3750 - val_loss: 29.0204 - val_accuracy: 0.3000
Epoch 56/100
80/80 [==============================] - 0s 123us/sample - loss: 20.4613 -
accuracy: 0.3750 - val_loss: 28.9046 - val_accuracy: 0.3000
Epoch 57/100
80/80 [==============================] - 0s 127us/sample - loss: 20.3230 -
accuracy: 0.3750 - val_loss: 28.7796 - val_accuracy: 0.3000
Epoch 58/100
80/80 [==============================] - 0s 201us/sample - loss: 20.2310 -
accuracy: 0.3750 - val_loss: 28.6553 - val_accuracy: 0.3000
Epoch 59/100
80/80 [==============================] - 0s 228us/sample - loss: 20.1218 -
accuracy: 0.3750 - val_loss: 28.5212 - val_accuracy: 0.3000
Epoch 60/100
```

```
80/80 [==============================] - 0s 141us/sample - loss: 19.9947 -
accuracy: 0.3750 - val_loss: 28.3981 - val_accuracy: 0.3000
Epoch 61/100
80/80 [==============================] - 0s 146us/sample - loss: 19.8901 -
accuracy: 0.3750 - val_loss: 28.2675 - val_accuracy: 0.3000
Epoch 62/100
80/80 [==============================] - 0s 161us/sample - loss: 19.7795 -
accuracy: 0.3750 - val_loss: 28.1372 - val_accuracy: 0.3000
Epoch 63/100
80/80 [==============================] - 0s 134us/sample - loss: 19.6702 -
accuracy: 0.3750 - val_loss: 28.0043 - val_accuracy: 0.3000
Epoch 64/100
80/80 [==============================] - 0s 151us/sample - loss: 19.5705 -
accuracy: 0.3750 - val_loss: 27.8735 - val_accuracy: 0.3000
Epoch 65/100
80/80 [==============================] - 0s 263us/sample - loss: 19.4711 -
accuracy: 0.3750 - val_loss: 27.7496 - val_accuracy: 0.3000
Epoch 66/100
80/80 [==============================] - 0s 468us/sample - loss: 19.3745 -
accuracy: 0.3750 - val_loss: 27.6210 - val_accuracy: 0.3000
Epoch 67/100
80/80 [==============================] - 0s 222us/sample - loss: 19.2754 -
accuracy: 0.3875 - val_loss: 27.4908 - val_accuracy: 0.3000
Epoch 68/100
80/80 [==============================] - 0s 235us/sample - loss: 19.1725 -
accuracy: 0.3875 - val_loss: 27.3611 - val_accuracy: 0.3000
Epoch 69/100
80/80 [==============================] - 0s 295us/sample - loss: 19.0780 -
accuracy: 0.4000 - val_loss: 27.2355 - val_accuracy: 0.3000
Epoch 70/100
80/80 [==============================] - 0s 258us/sample - loss: 18.9682 -
accuracy: 0.4000 - val_loss: 27.0962 - val_accuracy: 0.3000
Epoch 71/100
80/80 [==============================] - 0s 297us/sample - loss: 18.8616 -
accuracy: 0.4000 - val_loss: 26.9631 - val_accuracy: 0.3000
Epoch 72/100
80/80 [==============================] - 0s 267us/sample - loss: 18.7645 -
accuracy: 0.4125 - val_loss: 26.8362 - val_accuracy: 0.3000
Epoch 73/100
80/80 [==============================] - 0s 185us/sample - loss: 18.6590 -
accuracy: 0.4125 - val_loss: 26.7089 - val_accuracy: 0.3000
Epoch 74/100
80/80 [==============================] - 0s 228us/sample - loss: 18.5642 -
accuracy: 0.4250 - val_loss: 26.5805 - val_accuracy: 0.3000
Epoch 75/100
80/80 [==============================] - 0s 232us/sample - loss: 18.4649 -
accuracy: 0.4250 - val_loss: 26.4517 - val_accuracy: 0.3000
Epoch 76/100
```

```
80/80 [==============================] - 0s 250us/sample - loss: 18.3747 -
accuracy: 0.4250 - val_loss: 26.3245 - val_accuracy: 0.3000
Epoch 77/100
80/80 [==============================] - 0s 253us/sample - loss: 18.2659 -
accuracy: 0.4250 - val_loss: 26.2094 - val_accuracy: 0.3000
Epoch 78/100
80/80 [==============================] - 0s 237us/sample - loss: 18.1856 -
accuracy: 0.4250 - val_loss: 26.0973 - val_accuracy: 0.3500
Epoch 79/100
80/80 [==============================] - 0s 209us/sample - loss: 18.0907 -
accuracy: 0.4250 - val_loss: 25.9835 - val_accuracy: 0.3500
Epoch 80/100
80/80 [==============================] - 0s 176us/sample - loss: 17.9988 -
accuracy: 0.4250 - val_loss: 25.8676 - val_accuracy: 0.4000
Epoch 81/100
80/80 [==============================] - 0s 151us/sample - loss: 17.9046 -
accuracy: 0.4250 - val_loss: 25.7573 - val_accuracy: 0.4000
Epoch 82/100
80/80 [==============================] - 0s 215us/sample - loss: 17.8089 -
accuracy: 0.4250 - val_loss: 25.6461 - val_accuracy: 0.4000
Epoch 83/100
80/80 [==============================] - 0s 210us/sample - loss: 17.7307 -
accuracy: 0.4375 - val_loss: 25.5388 - val_accuracy: 0.4000
Epoch 84/100
80/80 [==============================] - 0s 219us/sample - loss: 17.6428 -
accuracy: 0.4375 - val_loss: 25.4270 - val_accuracy: 0.4000
Epoch 85/100
80/80 [==============================] - 0s 269us/sample - loss: 17.5425 -
accuracy: 0.4375 - val_loss: 25.3238 - val_accuracy: 0.4000
Epoch 86/100
80/80 [==============================] - 0s 252us/sample - loss: 17.4551 -
accuracy: 0.4375 - val_loss: 25.2237 - val_accuracy: 0.4000
Epoch 87/100
80/80 [==============================] - 0s 249us/sample - loss: 17.3696 -
accuracy: 0.4375 - val_loss: 25.1236 - val_accuracy: 0.4000
Epoch 88/100
80/80 [==============================] - 0s 228us/sample - loss: 17.2856 -
accuracy: 0.4500 - val_loss: 25.0234 - val_accuracy: 0.4000
Epoch 89/100
80/80 [==============================] - 0s 296us/sample - loss: 17.2375 -
accuracy: 0.4625 - val_loss: 24.9246 - val_accuracy: 0.3500
Epoch 90/100
80/80 [==============================] - 0s 228us/sample - loss: 17.1417 -
accuracy: 0.4625 - val_loss: 24.8282 - val_accuracy: 0.3500
Epoch 91/100
80/80 [==============================] - 0s 210us/sample - loss: 17.0647 -
accuracy: 0.4750 - val_loss: 24.7305 - val_accuracy: 0.3500
Epoch 92/100
```

```
80/80 [==============================] - 0s 228us/sample - loss: 16.9936 -
accuracy: 0.4750 - val_loss: 24.6366 - val_accuracy: 0.3500
Epoch 93/100
80/80 [==============================] - 0s 239us/sample - loss: 16.9101 -
accuracy: 0.4750 - val_loss: 24.5440 - val_accuracy: 0.3500
Epoch 94/100
80/80 [==============================] - 0s 260us/sample - loss: 16.8415 -
accuracy: 0.4750 - val_loss: 24.4510 - val_accuracy: 0.3500
Epoch 95/100
80/80 [==============================] - 0s 213us/sample - loss: 16.7753 -
accuracy: 0.4750 - val_loss: 24.3595 - val_accuracy: 0.3500
Epoch 96/100
80/80 [==============================] - 0s 201us/sample - loss: 16.6946 -
accuracy: 0.4750 - val_loss: 24.2690 - val_accuracy: 0.3500
Epoch 97/100
80/80 [==============================] - 0s 235us/sample - loss: 16.6163 -
accuracy: 0.4625 - val_loss: 24.1780 - val_accuracy: 0.3500
Epoch 98/100
80/80 [==============================] - 0s 253us/sample - loss: 16.5517 -
accuracy: 0.4625 - val_loss: 24.0935 - val_accuracy: 0.3500
Epoch 99/100
80/80 [==============================] - 0s 269us/sample - loss: 16.5457 -
accuracy: 0.4500 - val_loss: 24.0106 - val_accuracy: 0.3500
Epoch 100/100
80/80 [==============================] - 0s 213us/sample - loss: 16.4226 -
accuracy: 0.4500 - val_loss: 23.9208 - val_accuracy: 0.3500
Classical ML: Accuracy on training data: 0.4300000071525574
 Error on training data: 0.5699999928474426
Quantum ML: Accuracy on training data: 0.5799999833106995
 Error on training data: 0.42000001668930054
Classical ML: Accuracy on test data: 0.3499999940395355
 Error on test data: 0.6500000059604645
Quantm ML: Accuracy on test data: 0.550000011920929
 Error on test data: 0.44999998807907104
```

[ ]: