

Assignment 11

```
In [1]: import tensorflow as tf
import numpy as np
```

```
In [2]: # Generate some random data
np.random.seed(0)
x = np.random.rand(1000, 2)
y = np.random.randint(2, size=1000)
```

```
In [3]: # Split the data into training and testing sets
x_train, x_test = x[:800], x[800:]
y_train, y_test = y[:800], y[800:]
```

```
In [4]: # Define the logistic regression model
model = tf.keras.Sequential([
    tf.keras.layers.Dense(1, activation='sigmoid', input_dim=2)
])
```

```
In [5]: # Compile and train the model
model.compile(optimizer='adam',
              loss=tf.keras.losses.BinaryCrossentropy(from_logits=False),
              metrics=['accuracy'])

model.fit(x_train, y_train, epochs=10)
```

```
Epoch 1/10
25/25 [=====] - 0s 867us/step - loss: 0.8967 - accuracy: 0.5125
Epoch 2/10
25/25 [=====] - 0s 727us/step - loss: 0.8842 - accuracy: 0.5138
Epoch 3/10
25/25 [=====] - 0s 681us/step - loss: 0.8721 - accuracy: 0.5138
Epoch 4/10
25/25 [=====] - 0s 666us/step - loss: 0.8603 - accuracy: 0.5150
Epoch 5/10
25/25 [=====] - 0s 714us/step - loss: 0.8491 - accuracy: 0.5150
Epoch 6/10
25/25 [=====] - 0s 652us/step - loss: 0.8386 - accuracy: 0.5138
Epoch 7/10
25/25 [=====] - 0s 706us/step - loss: 0.8283 - accuracy: 0.5125
Epoch 8/10
25/25 [=====] - 0s 665us/step - loss: 0.8186 - accuracy: 0.5125
Epoch 9/10
25/25 [=====] - 0s 667us/step - loss: 0.8094 - accuracy: 0.5175
Epoch 10/10
25/25 [=====] - 0s 627us/step - loss: 0.8010 - accuracy: 0.5175
```

```
Out[5]: <keras.callbacks.History at 0x1f0ba0689d0>
```

```
In [6]: # Evaluate the model on the testing data
loss, accuracy = model.evaluate(x_test, y_test)
print('Test loss:', loss)
print('Test accuracy:', accuracy)
```

```
7/7 [=====] - 0s 992us/step - loss: 0.7887 - accuracy: 0.4900
Test loss: 0.7886603474617004
Test accuracy: 0.49000000953674316
```

