

Week2 homework

1. Code

```
import tensorflow as tf
import numpy as np

y = tf.Variable([[0.5, 0.3, 0.2],
                 [0.3, 0.2, 0.5]], dtype=tf.float32)
t = tf.constant([[1.0, 0.0, 0.0],
                 [0.0, 0.0, 1.0]], dtype=tf.float32)

def compGradient(y, t, LossFun):
    with tf.GradientTape() as tape:
        if LossFun == 'MSE':
            loss = tf.reduce_mean((t - y)**2)

        elif LossFun == 'BCE':
            loss = -tf.reduce_mean(t * tf.math.log(y) + (1.0 - t) *
tf.math.log(1.0 - y))

        elif LossFun == 'CCE':
            loss = -tf.reduce_mean(tf.reduce_sum(t * tf.math.log(y),
axis=1))

        grads = tape.gradient(loss, y)
        return grads, loss

grads, loss = compGradient(y, t, 'MSE')
print("MSE grads:", grads)
print("MSE loss:", loss)

grads, loss = compGradient(y, t, 'BCE')
print("BCE grads:", grads)
print("BCE loss:", loss)

grads, loss = compGradient(y, t, 'CCE')
print("CCE grads:", grads)
print("CCE loss:", loss)
```

2. 執行畫面&結果

```
MSE grads: tf.Tensor(
  [[-0.16666667  0.10000001  0.06666667]
   [ 0.10000001  0.06666667 -0.16666667]], shape=(2, 3), dtype=float32)
MSE loss: tf.Tensor(0.12666667, shape=(), dtype=float32)
BCE grads: tf.Tensor(
  [[-0.33333334  0.23809525  0.20833334]
   [ 0.23809525  0.20833334 -0.33333334]], shape=(2, 3), dtype=float32)
BCE loss: tf.Tensor(0.4243219, shape=(), dtype=float32)
CCE grads: tf.Tensor(
  [[-1. -0. -0.]
   [-0. -0. -1.]], shape=(2, 3), dtype=float32)
CCE loss: tf.Tensor(0.6931472, shape=(), dtype=float32)
```

3. 心得

這次的作業實作了 Mean Squared Error, Binary Cross-Entropy 跟 Categorical Cross-Entropy，來計算梯度跟 loss 很簡單的程式碼就可以實作，都沒有遇到任何問題。

4. 組員名單:

11160801 陳禹豪 5 分，原因：報告跟程式碼撰寫

11360821 陳鈺安 4 分