



## D89：訓練神經網路的細節與技巧 - 撰寫自己的 Loss function



PDF 下載

全螢幕

### Sample Code & 作業內容

請參考範例程式碼Day089\_CustomizedLoss.ipynb，請自行定義一個 loss function，為 0.3 \* focal loss + 0.7 cross-entropy，訓練並比較結果 (請讓比例為可以隨意調控，而非寫死)

作業請提交Day089\_HW.ipynb

檢視範例

### 參考資料

#### CSDN - Keras 自定義 Loss 函數

有時候我們想根據任務調整損失函數，Keras 可以在 compile model 時使用自定義函數。

##### 最常用的方式

- 自定義函數必須至少要有兩個參數：y\_true, y\_pred。其他參數則可視狀況自行加入。

##### 較不常用的方式

- 定義一個 loss\_layer
- 在 call function 中用 self.add\_loss 加其加入

```
#custom loss
def mycrossentropy(y_true, y_pred, e=0.1):
    return (1-e)*K.categorical_crossentropy(y_pred,y_true) + \
           e*K.categorical_crossentropy(y_pred, K.ones_like(y_pred)/num_classes)

model.compile(loss=mycrossentropy,
              optimizer=keras.optimizers.Adadelta(),
              metrics=['accuracy'])
```

```
class CustomVariationalLayer(Layer):
    def __init__(self, **kwargs):
        self.is_placeholder = True
        super(CustomVariationalLayer, self).__init__(**kwargs)

    def vae_loss(self, x, x_decoded_mean):
        xent_loss = original_dim * metrics.binary_crossentropy(x, x_decoded_mean) #Square Loss
        kl_loss = - 0.5 * K.sum(1 + z_log_var - K.square(z_mean) - K.exp(z_log_var), axis=-1)
        return K.mean(xent_loss + kl_loss)

    def call(self, inputs):
        x = inputs[0]
        x_decoded_mean = inputs[1]
        loss = self.vae_loss(x, x_decoded_mean)
        self.add_loss(loss, inputs=inputs)
        # We won't actually use the output.
        return x
```

參考來源：[https://blog.csdn.net/A\\_a\\_ron/article/details/79050204](https://blog.csdn.net/A_a_ron/article/details/79050204)

### 提交作業

請將你的作業上傳至 Github，並貼上該網址，完成作業提交

<https://github.com/>

確定提交

[如何提交](#) ▼

### 到 Cupoy 問答社區提問，讓教練群回答你的疑難雜症

[向專家提問](#)

[如何提問](#) ▼