

# Machine Learning HW3

學號: B06902060 系級: 資工三 姓名: 鄒宗霖

1. 請說明你實作的 CNN 模型 ( best model ) , 其模型架構、訓練參數量和準確率為何? ( 1% )

在我實作的 CNN 模型架構中有六層的 Conv2d 以及五層的 MaxPool2d , activation function 均為 PReLU ( parametric ReLU ) , 而在 fully connected NN 中實作了 Dropout (  $p = 0.5$  ) 。其模型的參數量為 15913664 , 準確率為 0.83801 ( Kaggle 上的成績 ) , 而下表為不同 iteration 下 validation 以及 training 上的準確率。

iter	0-25	25-50	50-75	75-100
val_acc	0.686	0.750	0.758	0.766
tra_acc	0.784	0.951	0.977	0.983

2. 請實作與第一題接近的參數量 , 但 CNN 深度 ( CNN 層數 ) 減半的模型 , 並說明其模型架構、訓練參數量和準確率為何? ( 1% )

CNN 模型架構中有三層的 Conv2d 以及三層的 MaxPool2d , activation function 均為 PReLU ( parametric ReLU ) , 而在 fully connected NN 中實作了 Dropout (  $p = 0.5$  ) 。其模型的參數量為 16198016 , 準確率為 0.79916 ( Kaggle 上的成績 ) , 而下表為不同 iteration 下 validation 以及 training 上的準確率。

iter	0-25	25-50	50-75	75-100
val_acc	0.675	0.717	0.724	0.725
tra_acc	0.753	0.931	0.954	0.966

3. 請實作與第一題接近的參數量 , 簡單的 DNN 模型 , 同時也說明其模型架構、訓練參數和準確率為何? ( 1% )

DNN 模型架構中有五層的 Linear , activation function 均為 PReLU ( parametric ReLU ) 。其模型的參數量為 15361792 , 準確率為 0.35803 ( Kaggle 上的成績 ) , 而下表為不同 iteration 下 validation 以及 training 上的準確率。

iter	0-25	25-50	50-75	75-100
val_acc	0.294	0.304	0.287	0.265
tra_acc	0.326	0.433	0.710	0.878

4. 請說明由 1 ~ 3 題的實驗中你觀察到了什麼? ( 1% )

首先 , 我們拿第一題的 CNN 模型與第二題的 CNN 模型做比較 , 第一題的 CNN 模型比較瘦且深 , 第二題的 CNN 模型比較胖且淺 , 雖然在 training set 上有著差不多的表現 , 但在 validation set 上有較大的差距 , 推測是因為較深的 CNN 模型中有比較完整的 modularization , 也就是說下層的 hidden layer 可以妥善的利用上層的 hidden layer 的結果 , 因此較能符合實際分類的情況。再來 , 我們拿 CNN 模型與 DNN 模型比較 , 雖然 DNN 模型在 training set 上有不錯的表現 , 但在 validation set 上的表現相當糟糕 , 推測是因為 convolution 以及 max pooling 清楚地掌握了照片的特質 , 找出照片裡的 patterns 進而拼湊出更精密的圖案 , 較能符合實際分類的情況。

5. 請嘗試 data normalization 及 data augmentation，說明實作方法並且說明實行前後對準確率有什麼樣的影響？( 1% )

```
train_transform = transforms.Compose([
    transforms.ToPILImage(),
    transforms.RandomHorizontalFlip(),
    transforms.RandomRotation(15), # 15 degrees
    transforms.ToTensor(), # normalize data to [0, 1] (data normalization)
])

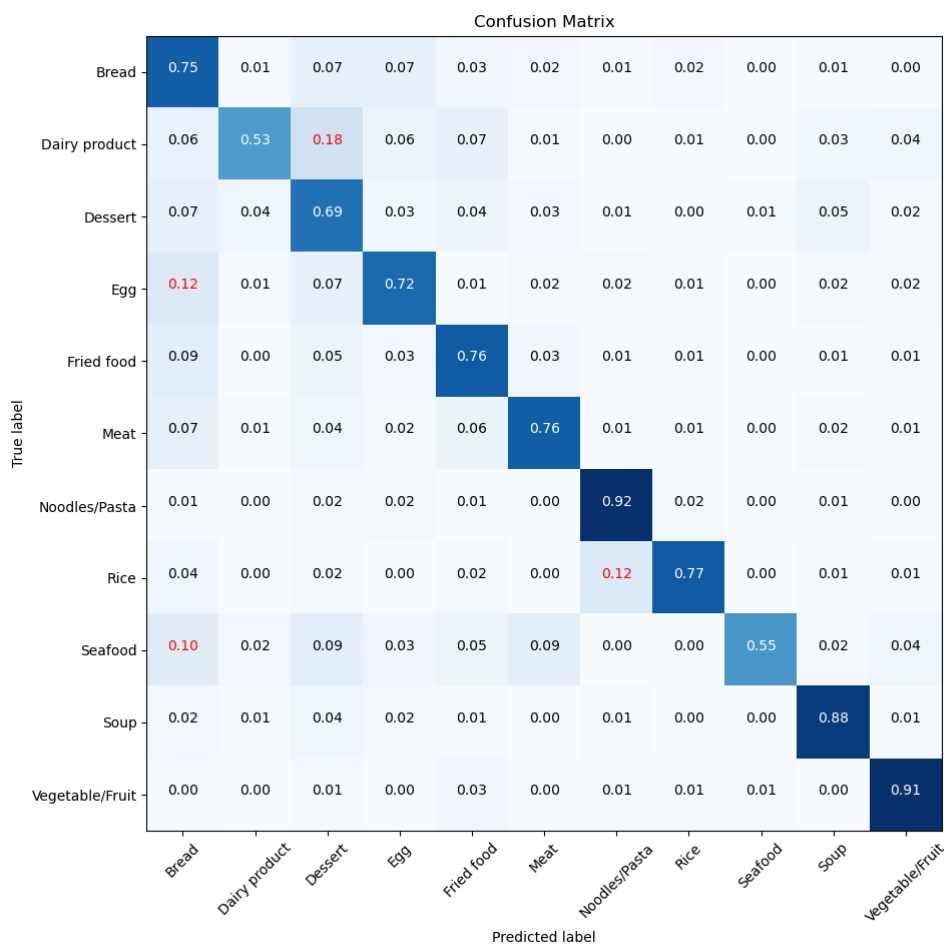
nn.BatchNorm2d(num_features)
```

利用 torchvision.transforms, nn.BatchNorm2d 實作 data normalization 及 data augmentation

iter	0-25	25-50	50-75	75-100
before	0.603	0.595	0.577	0.518
after	0.686	0.750	0.758	0.766

上表為在不同 iteration 下 validation set 的準確率，實作 data normalization 及 data augmentation 的模型準確率明顯較高，推測是因為 data augmentation 以後新增了水平翻轉以及旋轉的資料，擴大了 database；data normalization 以後每次更新參數的方向都是指向 loss function 的最低點，把原本複雜的 loss function 變得形狀規則些，因此較容易找到好的參數，準確率自然較高。

6. 觀察答錯的圖片中，哪些 class 彼此間容易用混？( 繪出 confusion matrix 分析 ) ( 1% )



上圖為各個類別間的 confusion matrix，可以看到紅色數字的部分為較容易用混的類別，像是 Dairy product 容易誤判成 dessert、egg 容易誤判成 bread、seafood 容易誤判成 bread、rice 容易誤判成 noodle/pasta，然而 noodle/pasta、vegetable/fruit、soup 這些類別正確率很高。