NYCU Introduction to Machine Learning, Homework 4

111550108，吳佳諭

Part. 1, Kaggle (70% [50% comes from the competition]):

(10%) Implementation Details

Type your answer here. (e.g., other hyper-parameters, training strategy, pre/post-processing, anything about how you train the model) Also, you have to fill in the two information below.

I use ResNet18 as the model. At first, I used ResNet 50-V2, but it couldn't perform very well. I guess it is because our dataset is not very large, and ResNet 50 is too large for the dataset, so it will overfit and therefore have a bad performance. On the other hand, if I build a 3-layer CNN by myself, it will be too simple to understand the data. Therefore, the model will underfit and also give a poor prediction.

First, split the training data into train (80%) and val (20%) for training and validation. The image will randomly flip or rotate to make the images have more different view. This can improve the model's ability to generalize across different senior.

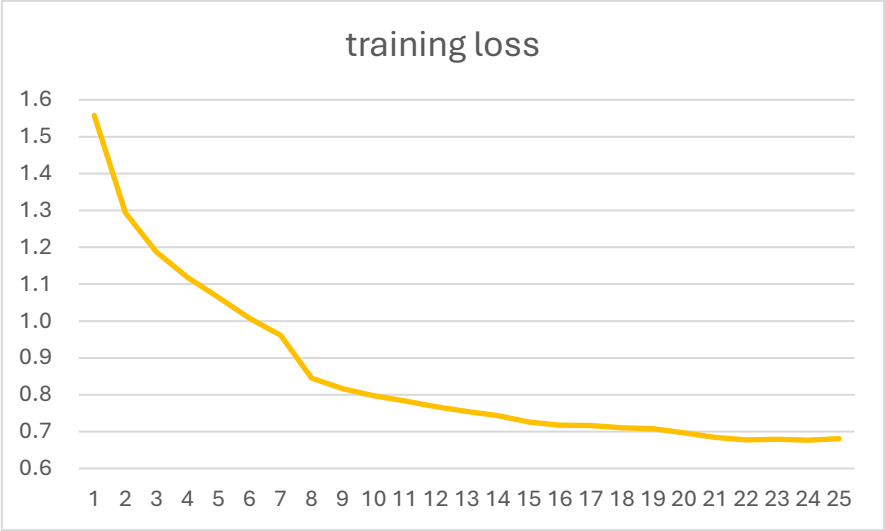
Then in training section, I use cross entropy as the loss function and SDG as the optimizer with learning rate 0.001. In addition, I use a learning rate scheduler to let the learning rate decrease overtime. That is, I use StepLR function with step size 7 and gamma 0.1, so the learning rate will drop by 10% every 7 epoch. In this way, the model can have a larger step at the beginning of the training, then use a smaller step to find the optimal solution. In the end of every epoch, there is a validation to test how well the model perform and avoid overfitting.

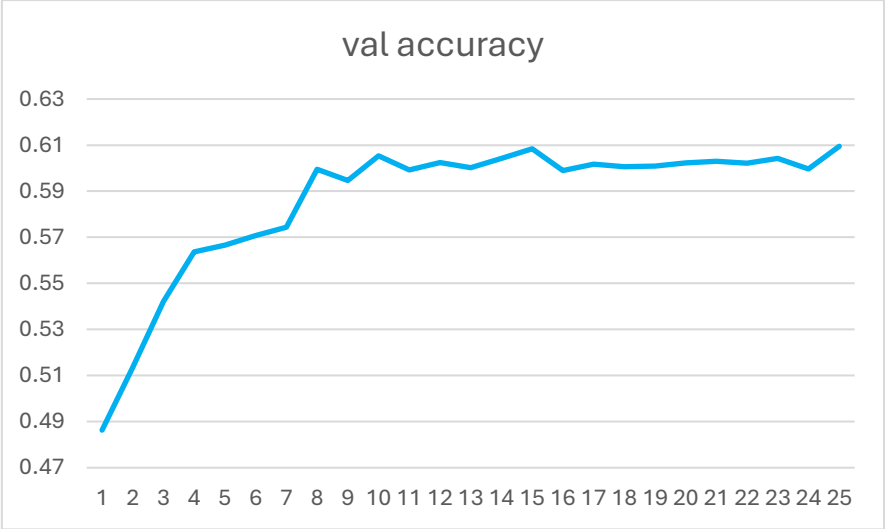
After the model finish training, I save the model with the highest accuracy. Predict data using this best model and save the result to CSV file.

Model backbone (e.g., VGG16, VGG19, Custom, etc)	ResNet18
Number of model parameters	11M

Other hyperparameters …	Batch size = 32 Learning rate = 0.001 Momentum = 0.9 Epoch = 25 Learning rate scheduler: step size = 7, gamma = 0.1 Loss = cross entropy Transform: flip, rotate 10

(10%) Experimental Results





At the end of training, the training loss can drop to 0.6811, and validation accuracy can reach 0.6095. Using this model to predict new data can get 0.54313 accuracy on testing images.

I also tried different models or hyperparameter, such as using ResNet50 or modifying the learning rate, but all of these cannot not have a better result than this model.

For ResNet50, the best result I could get is accuracy 0.52383, with learning rate 1e-4 and epoch 50. If we increase epoch to 100, the model will overfit.

ResNet50	lr	epoch	scheduler	Accuracy
1	1e-3	20	none	0.435
2	1e-4	20	none	0.499
6	1e-5	50	none	0.478
7	1e-4	50	none	0.515
8	1e-4	100	none	0.502
13	1e-3	14	ReduceLROnPlateau (factor=0.2, patience=2)	0.524

For a 3-layer CNN build by myself, with learning rate 1e-4 and epoch 100, it is too simple and only get accuracy 0.48921.

For ResNet18. If we use a fixed learning rate 0.001, it can only get 0.51793, which is even worse than ResNet50. Nevertheless, if we use a learning rate scheduler to modify the learning rate while training, the performance can improve a lot.

ResNet18	lr	epoch	optimizer	scheduler	acc
1	0.001	20	Adam	None	0.518
2	0.001	25	SDG (momentum=0.9)	StepLR(step_size = 7, gamma=0.1)	0.543
3	0.001	50	SDG (momentum=0.9)	ReduceLROnPlateau (factor= 0.2, patience=3)	0.539
--	0.001	30	SDG (momentum=0.9)	StepLR(step_size = 4, gamma=0.1)	0.505
4	0.001	30	SDG (momentum=0.9)	StepLR(step_size = 10, gamma=0.2)	0.531
5	0.001	50	SDG (momentum=0.9)	StepLR(step_size = 7, gamma=0.1)	0.534

Part. 2, Questions (30%):

1. (10%) Explain the support vector in SVM and the slack variable in Soft-margin SVM. Please provide a precise and concise answer. (each in two sentences)

   Support vector: data points that are lying on the decision boundary. Help to specify the boundary with the maximum margin.

   Slack variable: allow misclassification in non-linear separable classes. Create a soft margin to balance the maximum margin size and minimum error.

2. (10%) In training an SVM, how do the parameter C and the hyperparameters of the kernel function (e.g., γ for the RBF kernel) affect the model's performance? Please explain their roles and describe how to choose these parameters to achieve good performance.

   C: decide to maximize the margin and minimize the error by how much. If C is large, the model will focus more on minimize the error, increasing the potential of overfit. If C is small, the model will focus more on maximize the margin, which may lead to underfit.

   γ: control the importance of the training data on the decision boundary. If γ is large, the model will tend to create a complex boundary. On the other hand, if γ is small, the model will give a simple boundary, which may lead to underfit.

   These parameters affect the model's boundary, and therefore affect the performance. To achieve a good performance, we can use **cross validation** to evaluate the model's performance under different parameters.

3. (10%) SVM is often more accurate than Logistic Regression. Please compare SVM and Logistic Regression in handling outliers.

   In SVM, we can introduce slack variable to allow some misclassification, such that the model will not fit the outliers but let them be misclassified. However, in logistic regression, the model wants to minimize the loss, so the boundary will get close to the outliers to lower the loss. As a result, SVM can give a better prediction than logistic regression.