Report NN and CNN

Notes	
Links	
Name	
Priorities	
Status	☐

About data

- I use datasets from keras build-in library because outsource datasets from the internet is too large for submission file's size limit
- Train/Validation/Test ratio: (77/9/14)% for (54000/6000/10000) samples

About model architecture

Convolution Neural Network

I had try to build my own CNN and also implement the original LeNet-5 with architecture from [1], then I had change some hyper-parameter for better performance, some hyper-parameter the i had change include:

- Activation function: after each convolution layer and dense layer from sigmoid to ReLU because I think ReLU will work better for deep layer
- Filter: original LeNet have 6 filter for the first convolution layer and 16 filter for the second layer, I make it to 32 and 48 filter, I try to help these convolution layers get more 'observation' from the data

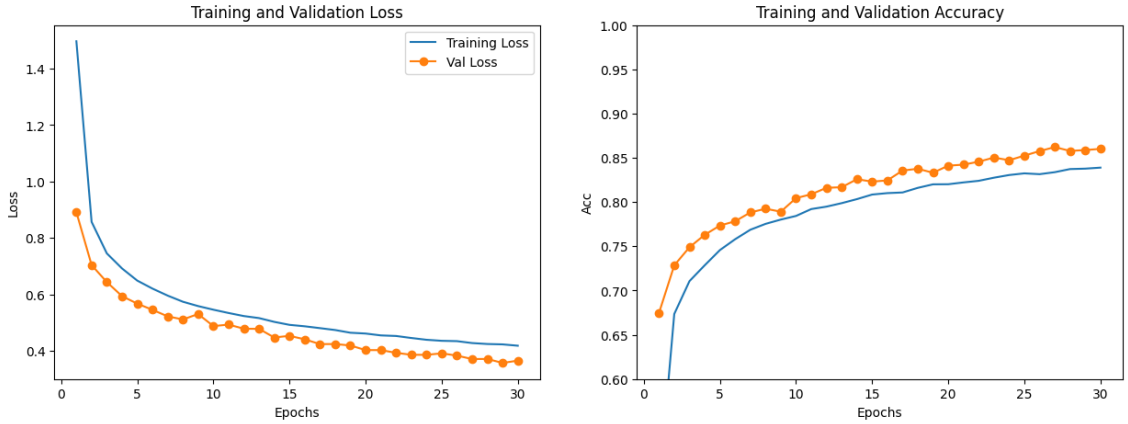
My own CNN

convolution (32, (3,3), relu) → convolution (64, (3,3), relu) → max pooling ((2,2)) → flatten → dense (64, relu) → dense (10, softmax)

- Accuracy
  - Fashion MNIST
    - Test data: 90.02%
    - Validation data: 91.27%
    - Training data: 99.07%
  - MNIST
    - Test data: 90.02%
    - Validation data: 91.27%
    - Training data: 99.07%

Original LeNet

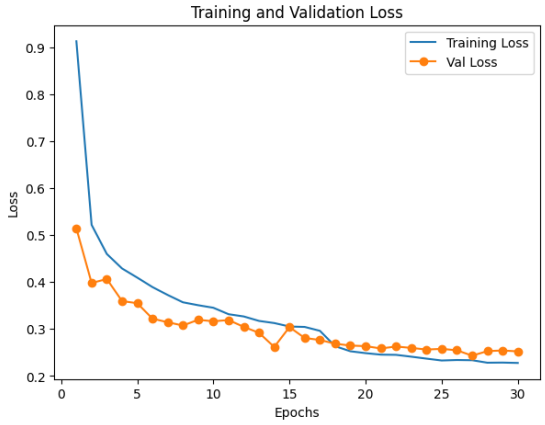
- Accuracy
  - Fashion MNIST: 30 epochs, the accuracy score is seem to increase after 30 epochs (base on the plot) so I try on 50 epochs
    - Test data: 90.02%
    - Validation data: 91.27%
    - Training data: 99.07%

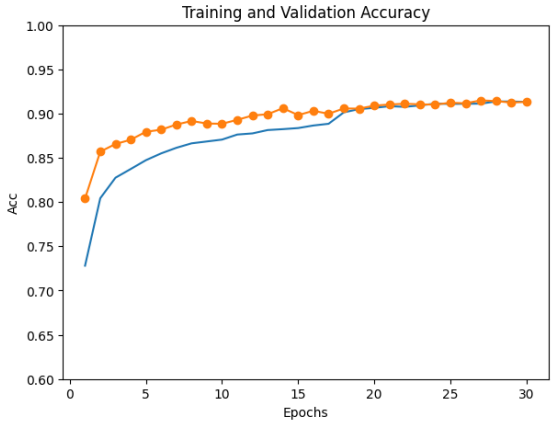


- MNIST
  - Test data: 98.62% ————
  - Validation data: 98.53%
  - Training data: 99.93%

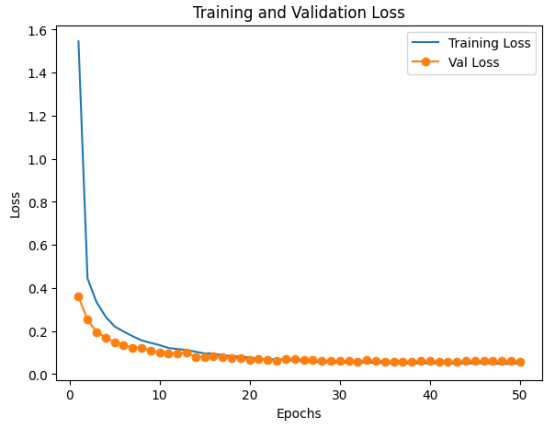
Fine-tune LeNet

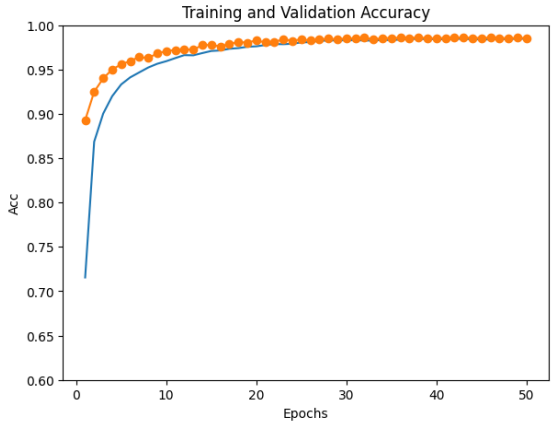
- Accuracy
  - Fashion MNIST
    - Test data: 91.19%
    - Validation data: 91.33%
    - Training data: 91.33%





  - MNIST
    - Test data: 98.87%
    - Validation data: 98.55%
    - Training data: 98.45%





AlexNet

- Accuracy for Fashion-MNIST datasets:
  - Test data: 90.82%
  - Validation data: 90.33%
  - Training data: 99.59% (best epoch: 99.84%)
- Both LeNet and my CNN are not perform well on Fashion-MNIST datasets so I try a deeper and more complex CNN to see the result → this AlexNet is much more bigger than LeNet with around 78 millions parameter. AlexNet seem to be over-fitting with this datasets when accuracy on training data is awesome but on validation and test data its even worse than fine-tuned LeNet (just a little), but we can see that LeNet can't have that fit on training data like AlexNet
- Original AlexNet architecture have 1000 output from the last dense layer but I change it to 10 output for this task

Conclusion

- LeNet not only have good accuracy but also in all datasets (training, val, test) those accuracy are very similar to each other that mean the model is not over-fitting → LeNet (CNN) architecture is very fit to 10 class image classification.
- The original LeNet and fine-tuned LeNet didn't give us a huge affect but fine-tuned have much more parameter to train and store (almost 5 time bigger)
- Accuracy on MNIST dataset always better than Fashion-MNIST (around 8% higher). I think because Fashion-MNIST image is more complex to classify and

architecture of LeNet cant handle it that well → I try on AlexNet, it is more “deeper” and bigger CNN architecture

- All my model didn’t work well on Fashion-MNIST datasets, maybe because preprocessing data
- Compare to the ANN homework last week:
  - All CNN-base architecture work better than ANN
  - A simple LeNet have around 61 thousand parameter but 1 hidden layer NN with 100 node already take 79 thousand parameter (784*100 + 100 + 100*10 + 10) but LeNet give us a better result
  - Feature extraction phase in CNN (convolution → pooling →convolution → pooling…) is very help-full for a NN after that it can reduce the size of model and increase performance in image classification task

Reference

[1] LeNet architecrue from Wikipedia

https://en.wikipedia.org/wiki/LeNet#:~:text=LeNet is a convolutional neural,a simple convolutional neural network.