# Computer Vision
Assignment 14

**Alireza Moradi**

January 3, 2021

# 1

## 1.1 Underfitting

Underfitting is the case where the model has "not learned enough" from the training data (e.g. small number of training epochs), resulting in low generalization and unreliable predictions.

## 1.2 Overfitting

Overfitting is the case where the overall cost is really small, but the generalization of the model is unreliable. This is due to the model learning “too much” from the training data set (memorizing the dataset). Regularization, Dropout, more training data, early stopping etc. are some of the ways to overcome ovefitting.
Source

# 2 Pretrained Networks

Transfer learning is the method of using the representations/information learned by one trained model for another model that needs to be trained on different data and for a similar/different task. Transfer learning uses pre-trained models. Training a neural network can take anywhere from minutes to months, depending on the data and the target task. But using transfer learning and pre-trained models we can now build deep learning applications that solve vision-related tasks much quicker. Using transfer learning and pre-trained models can boost accuracy of a model without taking much time to converge, as compared to a model trained from scratch.
Source

# 3 Parameters Count

Using normal convolution layers we have:

$$3 \times 5 \times 5 \times 128 + 128 = 9728$$

parameters.
With depthwise-separable convolution and M(depth multiplier) = 2, we have:

$$2 \times (5 \times 5 + 1) + 128 \times (3 \times 2 + 1) = 948$$

parameters.
Because in depthwise-separable convolution, convolution is applied to a single channel, followed by a $1 \times 1$ filter to cover the depth, we have much less parameters, and therefor narrower chance of overfitting.

## 4 Implementations

In the first two parts of this question, Training the model without Dropout, Normalization etc. extremely overfits the data, and that probably is because the model is very huge and too complex for this data set. And we can see the model trained with data augmentation has about 20% higher accuracy and less overfitting (both parts A and B have been trained for 30 epochs). In the third part, By freezing the Resnet model and training just the dense layers I got the accuracy of 72.5% and by fine-tuning the whole model, I got the accuracy of 70.1%, but in both cases the model is still a little overfitting because we still have a high loss (and low accuracy) value comparing to training loss. We can see that using a pre-trained model has a huge impact on the performance of the model.

| Experiment | Loss | Accuracy |
|---|---|---|
| Random Init | 5.71 | 23.3% |
| Random Init (with data aug) | 2.618 | 42.6% |
| Pre-Trained | 1.235 | 72.5% |
| Pre-Trained (Fine-tune) | 1.242 | 70.1% |

Table 1: Comparison in Loss and Acc for each experiment