KOÇ ÜNİVERSİTESİ
COLLEGE OF ENGINEERING

COMP 448/548
Medical Image Analysis

Spring 24
Homework #3

Kaan Dai 71651 – Alp Akkanlar 76022

KOÇ ÜNİVERSİTESİ
COLLEGE OF ENGINEERING

Implementation Details

1. We normalized the input dataset using transforms.Normalize() function. The mean and standard deviation parameters were selected based on the values provided in AlexNet Pytorch documentation. (mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])

2. For AlexNet using the pre-trained weights, inputs would need to be normalized using the mean [0.485, 0.456, 0.406] and standard deviation [0.229, 0.224, 0.225] for each color channel. We normalized the input dataset using transforms.Normalize() function.

3. Modifications were made only to the classifier part of AlexNet. The last layer of the classifier was originally outputting 1000 classes and it was replaced with a new linear layer with 3 outputs to suit the 3-class problem. This was done by setting model_conv.classifier[6] = nn.Linear(num_ftrs, 3) where num_ftrs is the number of input features to the last layer.

4. We used the loss function Cross Entropy Loss (nn.CrossEntropyLoss()), which is used for multi-class classification problems. This loss function combines nn.LogSoftmax() and nn.NLLLoss() in one single class.

5. For backpropagation, we used the Stochastic Gradient Descent (SGD) optimizer with parameters: learning rate = 0.001 and momentum = 0.9. These parameters were chosen to balance convergence speed and stability. We implemented the StepLR learning rate scheduler with a step size of 1 and a gamma of 0.1 to reduce the learning rate periodically and enhance convergence during training.

6. The class imbalance problem was addressed by calculating weights for each class based on their frequency in the dataset and then applying these weights in a WeightedRandomSampler. This ensures that during training, each batch has a balanced representation of each class, thus preventing the model from being biased toward the more frequent classes

	Training portion of the training set				Validation portion of the training set				Test set			
	C1	C2	C3	All	C1	C2	C3	All	C1	C2	C3	All
With input normalization and with addressing the class imbalance problem	94%	88%	97%	97%	90%	83%	87%	93%	91%	83%	85%	92%
With input normalization and without addressing the class imbalance problem	89%	92%	78%	93%	84%	86%	74%	90%	86%	88%	74%	90%
Without input normalization and with addressing the class imbalance problem	81%	85%	77%	90%	80%	80%	70%	87%	79%	82%	70%	87%