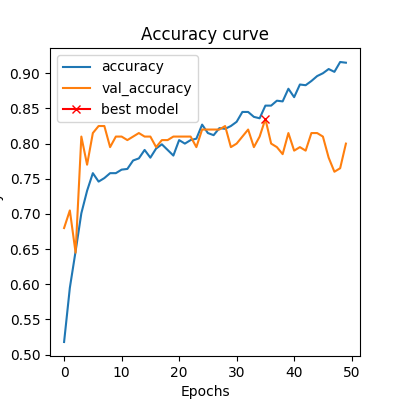
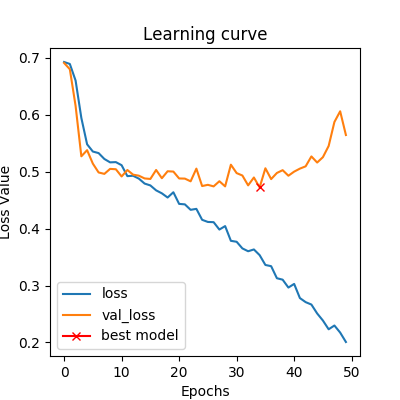
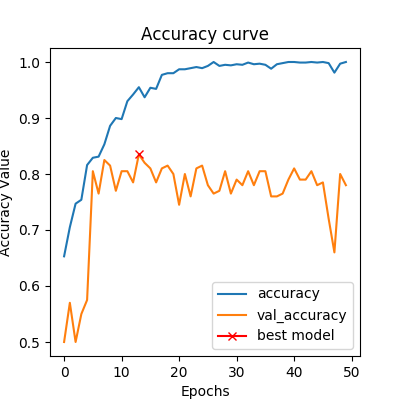
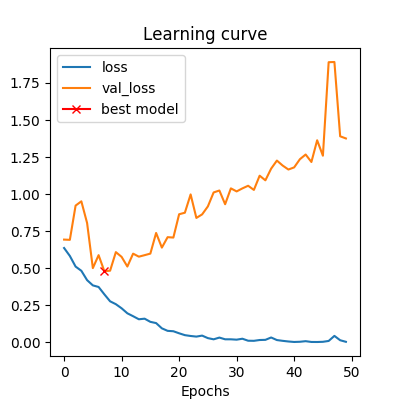
Lab 2

Task 1a

The final training accuracy for the AlexNet is – 0.90

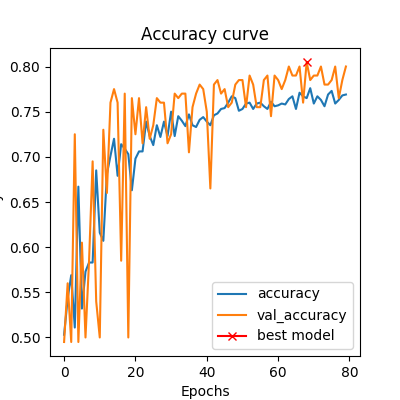
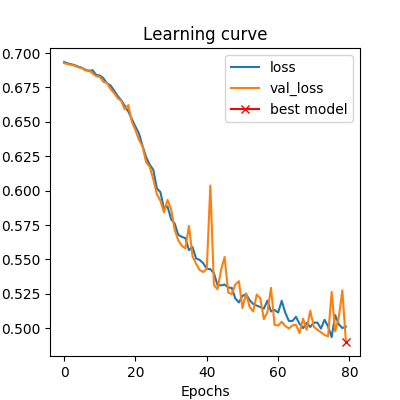
 

Task 1b

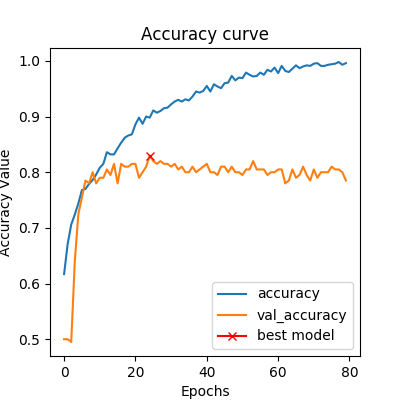
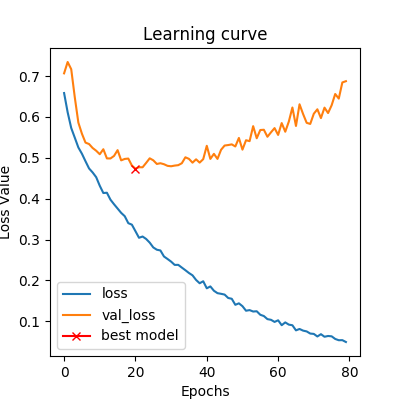
We observed the same training accuracy as task 1a after 10 epochs. The final training accuracy is 1.0. After inserting the batch-normalization, we can observe a very heavy overfitting in our model.   
 

Task 1c

The model with Batch Normalization had a higher validation accuracy. The model without the Batch Normalization has a higher generalization power because it is not overfitting for 80 epochs (given in the question). To increase the stability of a Neural Network, the batch normalization normalizes the output of the previous activation layer before it is fed into the next layer. Batch normalization also reduces overfitting because it has slight regularization effects. However, as we can see in this case that the model with batch normalization is still overfitting, so we cannot just primarily rely on Batch Normalization for solving the problem of overfitting. The main function of batch normalization is to normalize the outputs of the various layers of a Neural Network and speed up the training process.

The accuracy and loss graphs for the model without Batch normalization

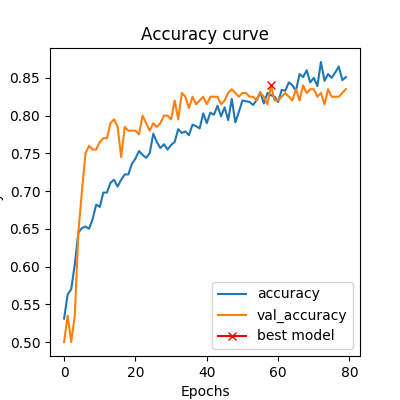
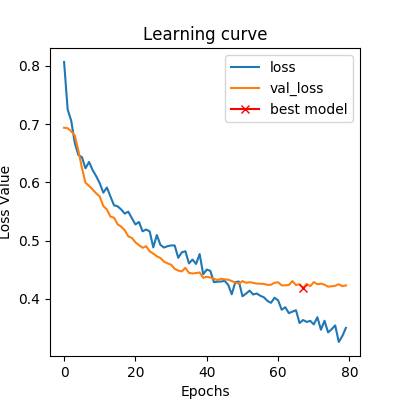
 

The accuracy and loss graphs for the model with Batch normalization

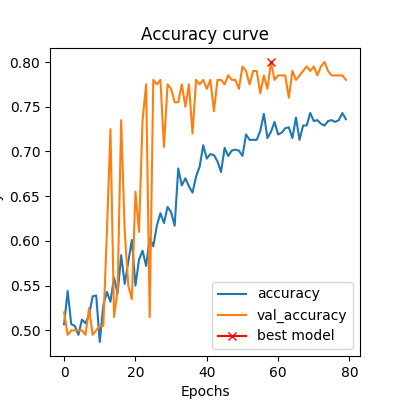
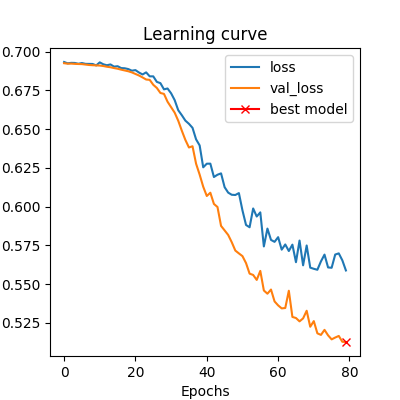
Task 2

The model with batch normalization layers has a higher validation accuracy and lower validation loss compared to the accuracy and validation loss of the model without batch normalization, over 80 epochs. Also, in the model without batch normalization, the validation loss is always lower than the training loss.

If we compare the models in the Task2 with the previous models, we can clearly observe that adding a dropout layer helped us in reducing the overfitting.

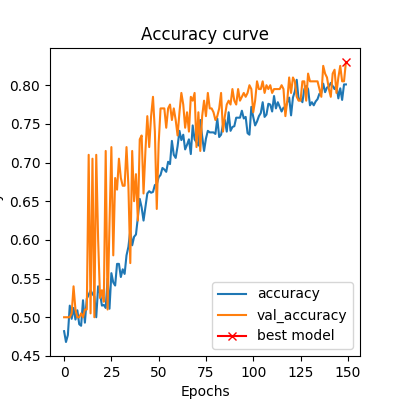
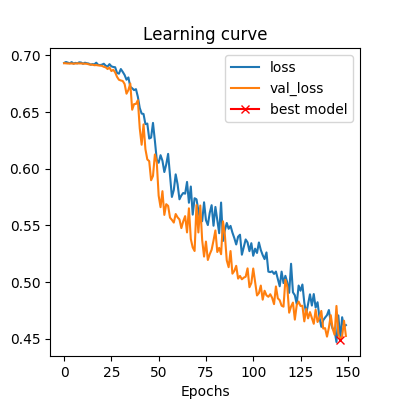
The accuracy and loss graphs for the model with Batch normalization

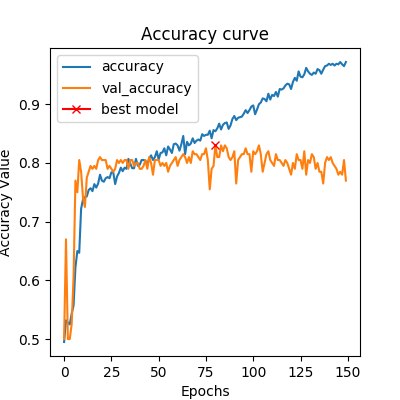
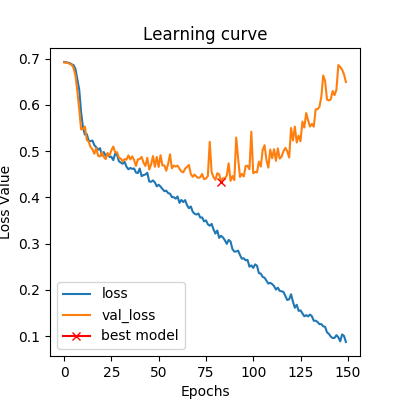
The accuracy and loss graphs for the model without Batch normalization

Task 3

The model with the drop out layers is performing better. The validation accuracy for the best model with dropout layers is higher than the validation accuracy for the best model without dropout layers. Also, the validation loss for the best model with dropout layers is lower than the validation loss for the best model without dropout layers. The model with the dropout layers has a better generalization capacity as it can be clearly seen that it is not overfitting over 150 epochs, whereas the model without dropout layers starts overfitting after 75 epochs. Dropout randomly shuts down neurons in the layers of a neural network during the training phase. The main purpose of dropout is to avoid overfitting.

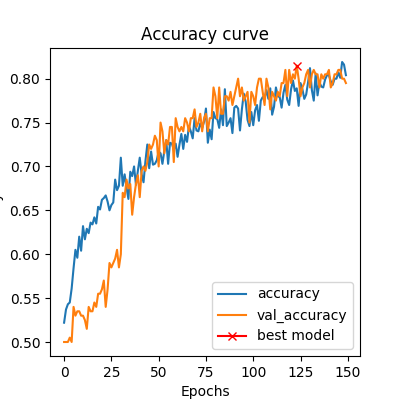
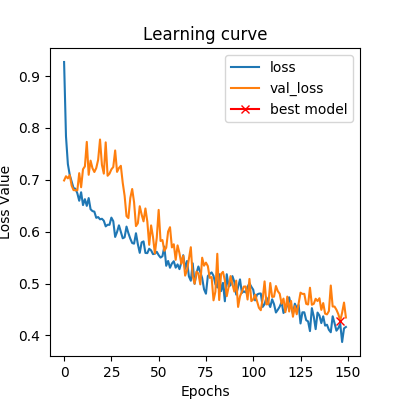
 

The accuracy and loss graphs for the model with Dropout layers

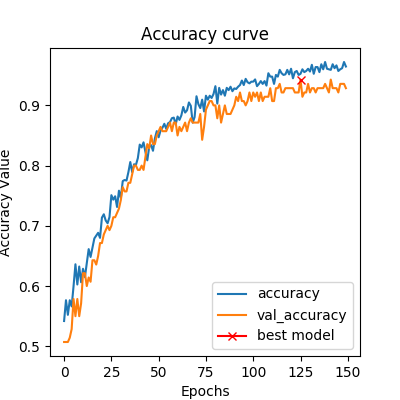
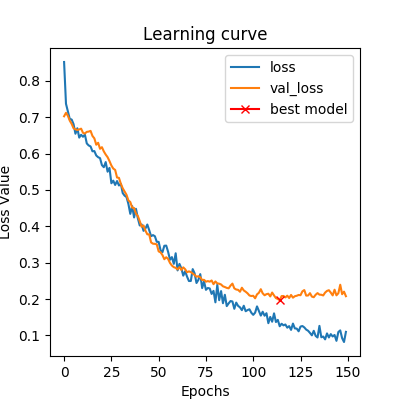
 

The accuracy and loss graphs for the model without Dropout layers

Task4

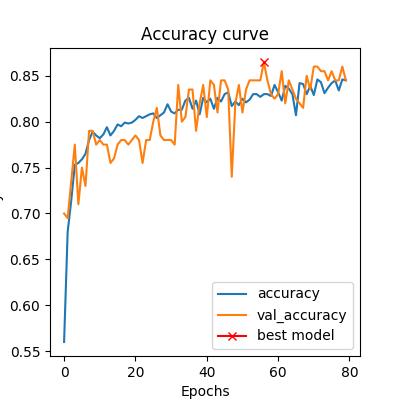
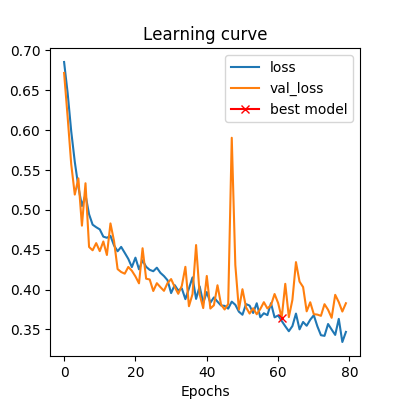
The accuracy and loss graphs for VGG model trained on Skin Data

The accuracy and loss graphs for VGG model trained on Bone Data

Task 6

The validation accuracy for the best model is 0.85 and the validation loss for the best model is 0.36. The model is preforming well, is not overfitting and seems to have a good generalization power. Data Augmentation is a method used to artificially expand the training data set by creating different variations (rotations, shifts, flips, zooms, etc.) of the training images and has definitely helped in improving the performance of the model. Training the model with a bigger dataset increases its generalization ability and leads to improved performance on unseen data.

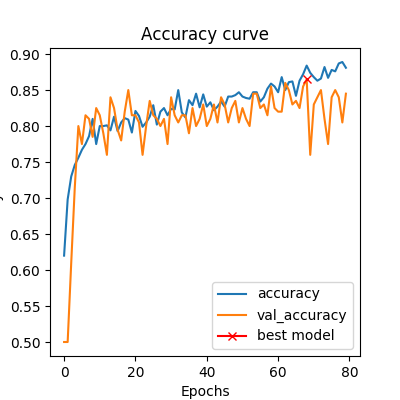
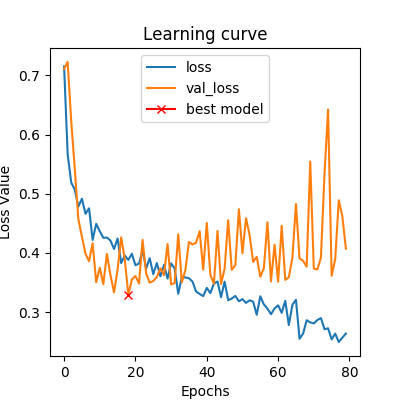
The accuracy and loss graphs for AlexNet model with batch normalization

and dropout layers, trained on the Augmented Dataset

Task 7

The validation accuracy for the best model is 0.85. The best training accuracy is 0.87.

The validation loss for the best model is 0.34. The best training loss is 0.22.

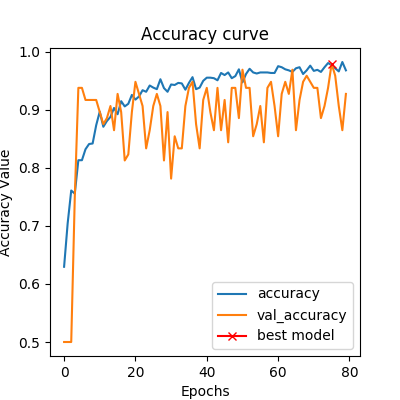
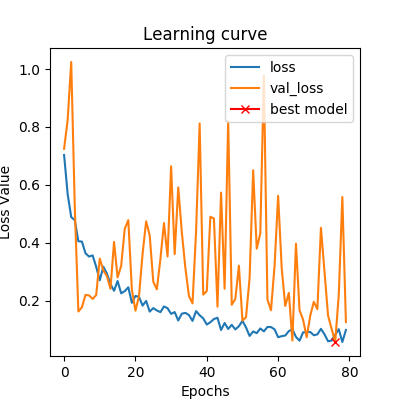
 

The accuracy and loss graphs for VGG model with batch normalization

and dropout layers, trained on the Augmented Dataset (SKIN DATASET)

The validation accuracy for the best model is 0.98. The best training accuracy is 0.98.

The validation loss for the best model is 0.12. The best training loss is 0.12.

The accuracy and loss graphs for VGG model with batch normalization

and dropout layers, trained on the Augmented Dataset (BONE DATASET)

Task 9

This model is preforming better than the previous VGG model on both skin and bone dataset.

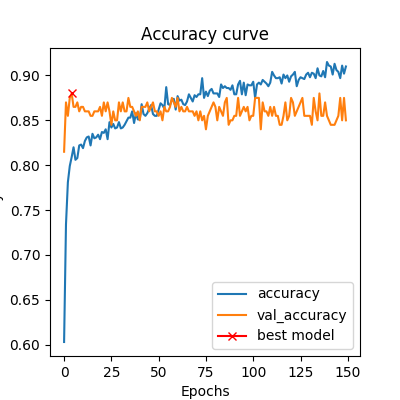
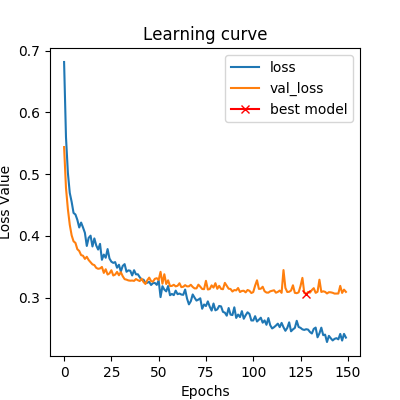
For the Skin dataset - Validation Accuracy for the best model is 0.87 and the validation loss for the best model is 0.32.

For the Bone dataset - Validation Accuracy for the best model is 1.00 and the validation loss for the best model is 0.02.

Transfer learning speeds up the training process and generally increases the generalization power of the model.

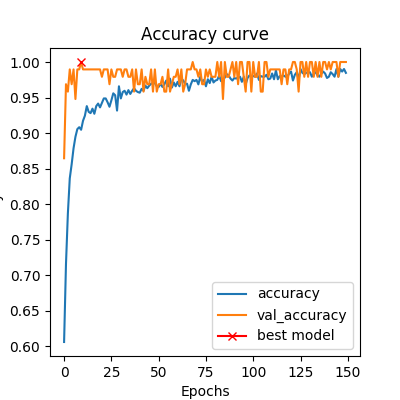
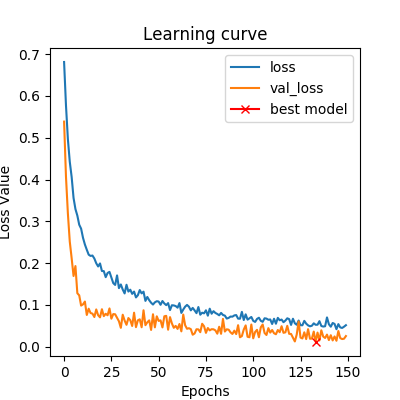
The best way to check if the training and validation results are reliable is to test the model’s performance on the test set. We can see that transfer learning is enabling the model to learn faster as the best validation accuracy for both the models was achieved in less than 10 epochs and by looking at the corresponding validation loss curves, we can see that the validation loss hasn’t gone down much after the best accuracy was achieved. The validation curves for both Bone and skin datasets show very less fluctuation after the epoch in which the best accuracy was achieved.

Comparing this to the VGG model that was trained in the task 7, the best accuracy for the skin dataset model was achieved in 70 epochs and the best accuracy for the bone dataset model was achieved in 80 epochs.

The accuracy and loss graphs for VGG model – Using transfer learning

for the SKIN Images

The accuracy and loss graphs for VGG model – Using transfer learning

for the BONE Images