Q1.1: A black screen with blue text

Description automatically generated

A black background with many small colored letters

Description automatically generated with medium confidence

A computer screen shot of text

Description automatically generated

A screen shot of a computer program

Description automatically generated

Code just demonstrates the sampling from the dataset (2500 images and randomly sampling 500 of them)

Q1.2:

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Code picks 5 random images from the dataset. It shows the label below the image

Q1.3:

Normalization is incredibly important as it transforms the pixel value to be stuck (for at least this context) [-1,1]. By forcing the pixel values to be in this range, it becomes much easier for the model to learn because gradient updates are more stable during training. This in turn leads to faster training as the model features are on similar scales, as the values are normalized

A black background with white text

Description automatically generated

A screenshot of a computer screen

Description automatically generated

Q2.1

A screen shot of a computer program

Description automatically generated

A black rectangle with white text

Description automatically generated

A screen shot of a computer code

Description automatically generated

Created a basic CNN (LeNet5), where I adjusted the layers in the model to fit the requirements (Convolutional layers with ReLU activations and average-pooling layers, Fully connected layers with appropriate dropout rates, Output layer with 10 classes (one for each class in CIFAR-10).

Q3.1

A screen shot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generated

Used Adams optimizer and cross entropy loss function to train and evaluate the model. The model runs for 10 epochs

Q3.2

A black screen with white text

Description automatically generated

These are the results from the training of the model.

Q3.3 A graph of training and training loss

Description automatically generated

This is a graph demonstrating how the model was learning from accuracy as well as loss. As you can see, the training and validation decreased as epochs increased. The opposite happened for training accuracy and validation accuracy.

Q4.1

A computer screen shot of a program code

Description automatically generated

A computer screen with text

Description automatically generated

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer

Description automatically generated

Overall, the results are little less in comparison to the original model, though they are very comparable. Though, as the model is still improving similar to the original model, it shows that there regularization is still effective (which lowers overfitting), although it does this worse than original model.

Q4.2

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer program

Description automatically generated

A computer screen shot of a program code

Description automatically generated A screen shot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

The model training accuracy and validation accuracy was increasing for all of them. It has improved in comparison to the original model and ReLU activation function performed the best.

Q4.3

A screen shot of a computer program

Description automatically generated

A black rectangle with white text

Description automatically generatedA black screen with white text

Description automatically generated

The model I coded earlier already had a drop out layer in this. The dropout layer made overfitting not happen as badly, as you can see the validation accuracy is increasing. The dropout layer basically deactivates neurons, which can help prevent overfitting.

Q4.4:

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

For my code, my minimum pooling and max pooling code I believe was overfitting. The validation training is consistently low and random, increasing or decreasing. The training accuracy increases steadily so it does appear with min and max pooling that it leads to overfitting.

Q5.1:

A computer screen shot of a program code

Description automatically generated

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

Code shows model trained two different ways, augmented and without augmentation. As can be seen from the results, the augmented has a significantly better accuracy on identifying the images. The dataset for these images contained the entire one, so it took significantly longer in comparison to running only 2500 images.