

Deep Learning

Student I: kespa139 (Keshav Padiyar Manuru)

Student II: toran584 (Tore Andersson)

Question 1:

The shape of X_{train} and X_{test} has 4 values. What do each of these represent?

Answer:

`tensor.shape` returns the total number of axes of the tensor. Here there are 4 values returned for both train and test, which means datasets are consisting of 4 axes. Let us consider: **X_{train}** : (50000, 32, 32, 3)

Elements along Axis - 0 of the tensor: 50000 \ Elements along Axis - 1 of the tensor: 32 \ Elements along Axis - 2 of the tensor: 32 \ Elements along Axis - 3 of the tensor: 3

Considering `cifar10` data, it is containing the arrays of image pixels. There are 50000 image objects/items, each pictures with 32X32 pixels and these images are color images hence, the last axis is the *R, G and B* layer.

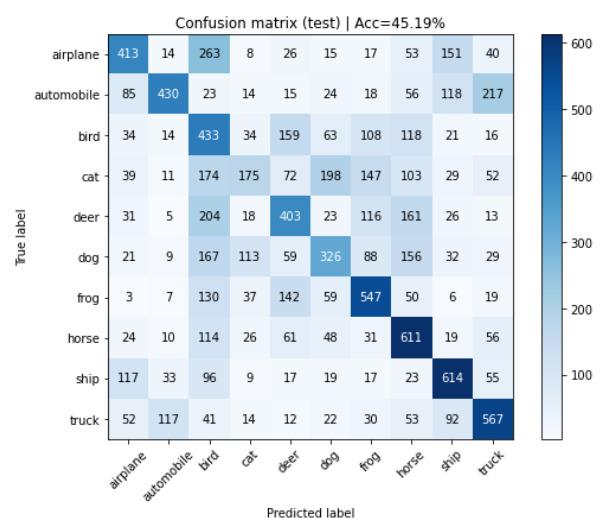
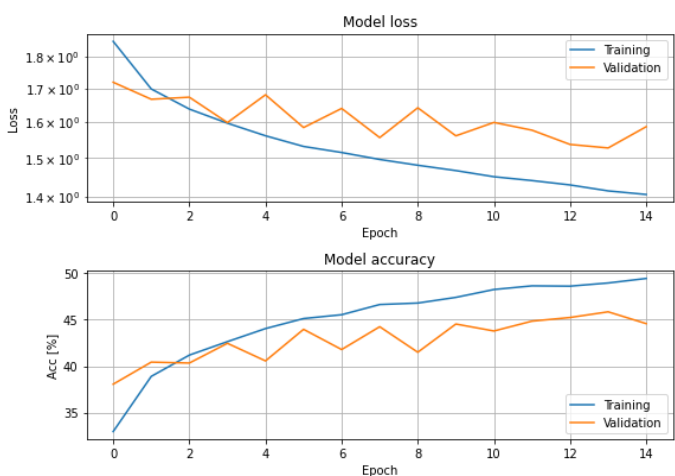
And in **X_{test}** , there are 10000 image objects(items), with other dimensions same as in train.

Question 2:

Train a model that achieves above 45% accuracy on the test data. In the report, provide a (short) description of your model and show the evaluation image.

Answer:

We have come up with a 2 layer neural network. Where, 1st layer consists of 128 neurons, 2nd layer contains 64 neurons. "Rectified Linear Unit (relu)" is used as a linear activation function to avoid the vanishing gradient problem. There data is comprised of 10 different classes, hence there are 10 units in the output layer. We are using "Softmax" as activation function to calculate the probability to classify the object in the output layer.



Question 3:

Compare this model to the one you used for the MNIST dataset in the first assignment, in terms of size and test accuracy. Why do you think this dataset is much harder to classify than the MNIST handwritten digits?

Answer:

MNIST data: this dataset contains pixels of images of handwritten digits (0-9) that are in black and white, hence there is only one color channel. In addition, the MNIST have less local features/ primitive details (edges/corners) which are not complex.

On the other hand, cifar10 data contains color images of objects such as aeroplane, truck, dog ect, which are having more local details/features than in the MNIST data.

The difference between the datasets in terms of complexity is quite large, an image from the MNSIT dataset represents a two dimensional object, whilst the CIFAR dataset visualizes 3 dimensional objects such as a car which could be viewed from multiple angles and has different features in it self. Hence the CIFAR10 data is more harder to classify.

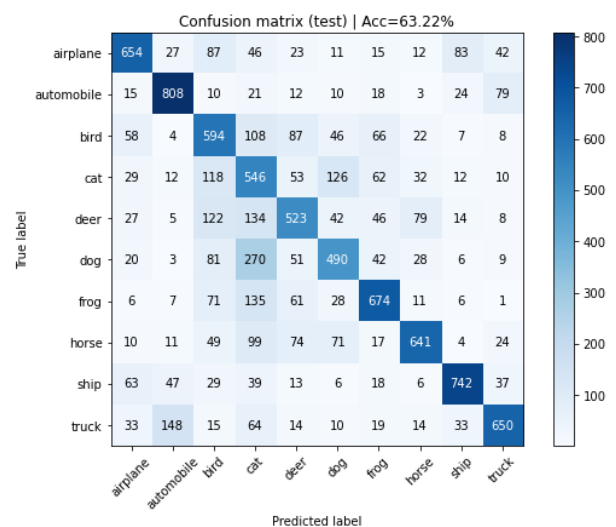
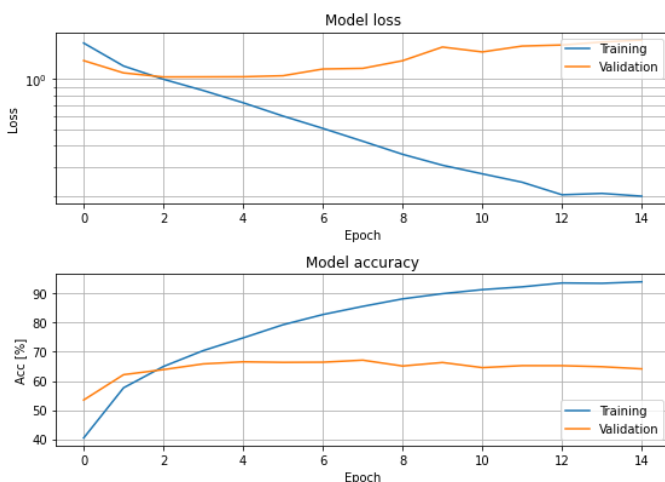
Question 4:

Train a model that achieves at least 62% test accuracy. In the report, provide a (short) description of your model and show the evaluation image.

Answer:

The neural network contains three convolutional layers with the activation function relu, the kernel size used for convolution is a (3x3) matrix. Each convolution layer is followed by a pooling function where the pooling is set to a "window" of (2x2) matrix. Where it will take the highest pixel-value in each "window". The strides or stepsize of which the window moves is (1x1). The filters of each of the convolutions goes from simple to more complex with the values: 32,64, 128.

Then the output is flattened into a single vector and used as input into the fully connected classifier with two layer network with relu as activation and 64 and 32 neurons each. The output layer uses softmax activation and has 10 neurons which corresponds to the number of classes.



Question 5:

Compare this model with the previous fully connected model. You should find that this one is much more efficient, i.e. achieves higher accuracy with fewer parameters. Explain in your own words how this is possible.

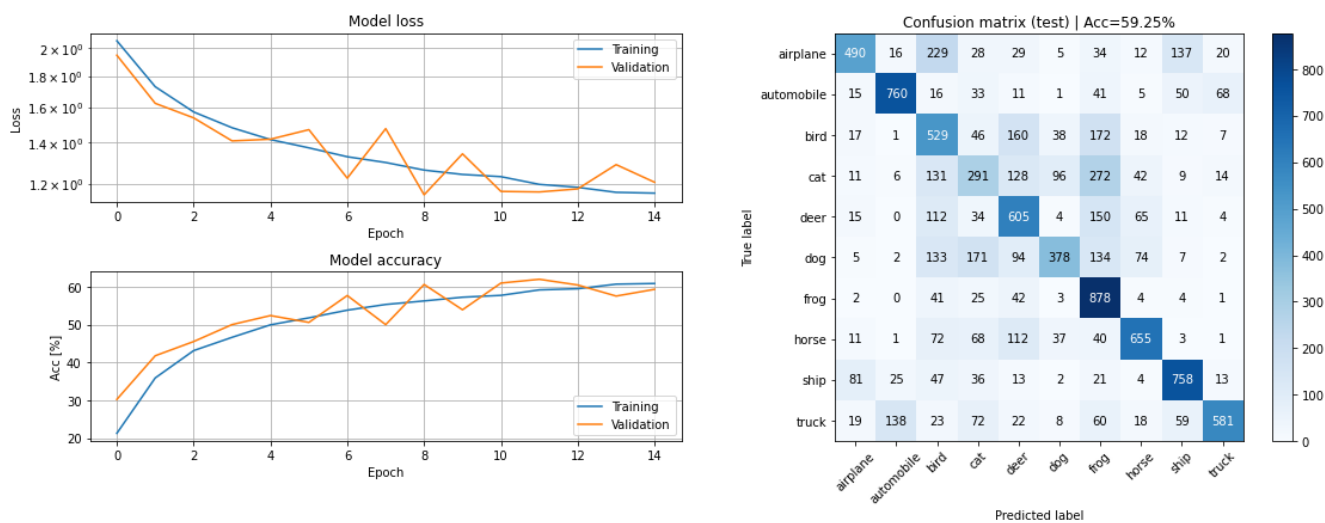
Answer:

The combination of convolution - activation - pooling is can be seen as the feature learning phase of the neural network. Where network tries to learn the primitive image features such as edges and lines of the dataset which will provide as kind of preprocessing for the neural network to improve the input into the fully connected network. The convoulution layer detects particular features depending on the kernel used. These multiple features are pooled together in the maxpooling phase which combines the maximum pixel values in the defined pooling window (in our case, window is a 2X2 matrix). Because of this preprocessing arrangement we get more abstract features as input to the fully connected classifier, hence we see a better result.

Question 6:

Train the modified CNN-model. Save the evaluation image for the report.

Answer:



Question 7:

Compare this model and the previous in terms of the training accuracy, validation accuracy, and test accuracy. Explain the similarities and differences (remember that the only difference between the models should be the addition of Dropout layers).

Hint: what does the dropout layer do at test time?

Answer:

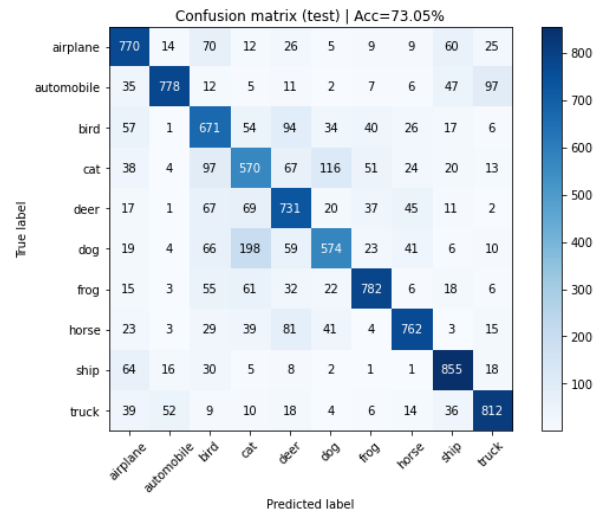
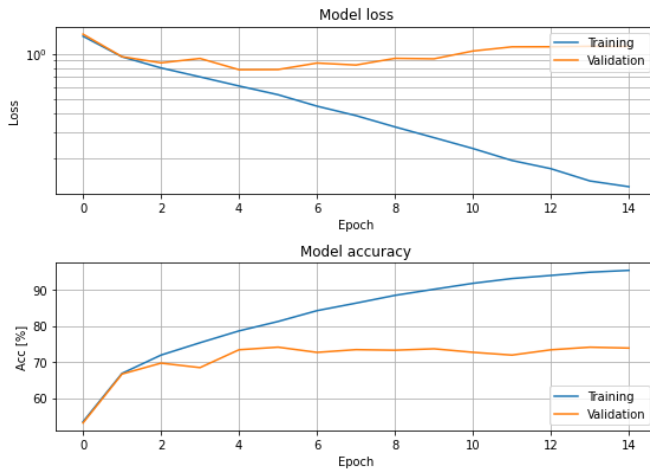
	Training	Validation	Test
CNN	0.9388	0.6414	0.632
CNN-drop	0.6087	0.5933	0.592

The training accuracy is much higher in the CNN without drop compared to the one with. Using drop, training; validation and test have a similar accuracy. The difference in training to validation and test accuracy for the CNN without drop is because, the model overfits the data for the training due to lack of regularization. By introducing the drop functionality the model generalizes better as and becomes less sensitive to the data, hence does not overfit.

Question 8:

Train the model and save the evaluation image for the report.

Answer:



Question 9:

When using BatchNorm one must take care to select a good minibatch size. Describe what problems might arise if the wrong minibatch size is used.

You can reason about this given the description of BatchNorm above, or you can search for the information in other sources. Do not forget to provide links to the sources if you do!

Answer:

As it is mentioned in the description of Batch Normalization, the batch size we select should contain the data points that represent the whole dataset, there by to calculate optimal statistics. In addition, from equations provided, if the batch size = 1, then the normalized output will be zero. Hence the batch size should be always >1 . Now if batch sizes are smaller, chances that the samples representing whole dataset is very less hence algorithm might update the noisy parameters, which is good in a way as we can have better regularization. There is another possibility of frequent skipping of unfavourable local minima if the batch sizes are smaller because of noisy updates. On the other hand, for the large batch size, the parameter updates are closer to the actual gradient and less regularization. (Ref: <https://kth.diva-portal.org/smash/get/diva2:955562/FULLTEXT01.pdf> (<https://kth.diva-portal.org/smash/get/diva2:955562/FULLTEXT01.pdf>))

Question 10:

Design and train a model that achieves at least 75% test accuracy in at most 25 epochs. Save the evaluation image for the report. Also, in the report you should explain your model and motivate the design choices you have made.

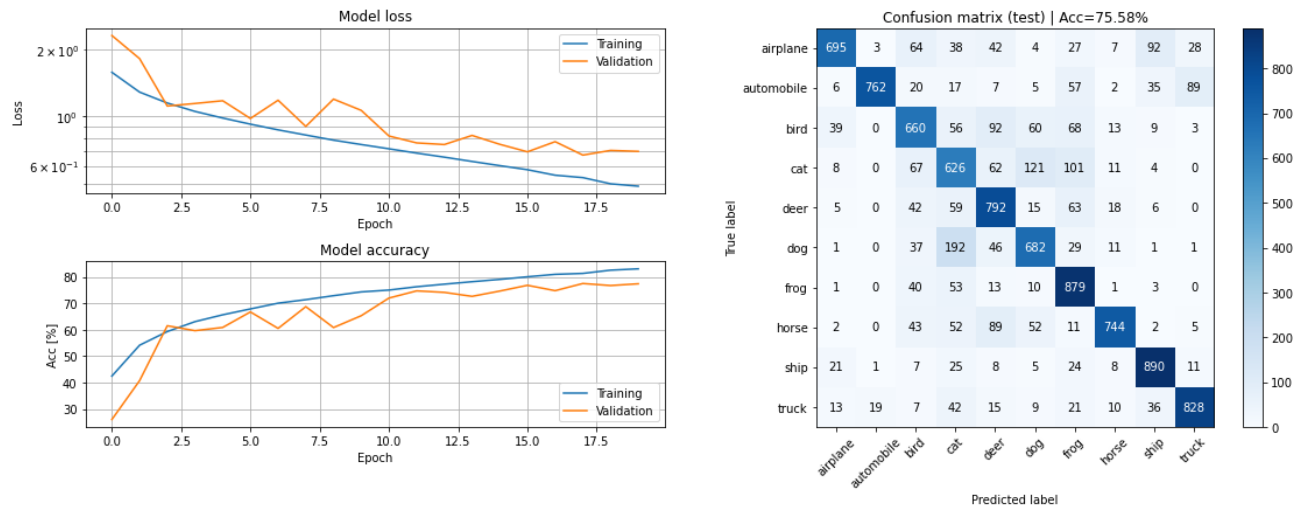
Answer:

The design of the model is an extension of the previous model from CNN now with 4 convoloutional layers with 32,64,128 and 256 filters each. The chosen paramaters are as for previous models. After each convolutional layer batchnormalization is performed with 32 as the min-batchsize. Then activation with relu is used followed by pooling as the same as before. After each pass through the steps of convolution - batchnormalization - activation -pooling there is a drop function called with the probability of 30% that the results calculated will not be used and previous results will be sent forward to the next part of the network.

For the fully connected part the neural network layers contained 128, 64 neurons. After each layer batchnormalization is applied and the drop function is called. As previous the output layer uses softmax as activation with 10 neurons.

This model achived in 20 epochs:

Test loss = 0.741\ Test accuracy = 0.756



The addition of batch normalization is to increase the speed of convergence of the network and the drop function is included for regularization such that the model will avoid overfitting on the data. From trial and error we found adding another learning layer and increasing the number of nerons in the fully connected neural network increased both the training and test accuracy. Adding another feature learning layer increased the abstraction of the image features and provide better input for the fully connected part. Implementing batchNormalization in the network after each convolution increases the speed of convergence of the network and the drop function is included for regularization such that the model will avoid overfitting on the data. Increasing the number of neurons in the fully connected part to the model increases the