TBMI26 – Computer Assignment Reports Deep Learning

Authors

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 toatl 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.

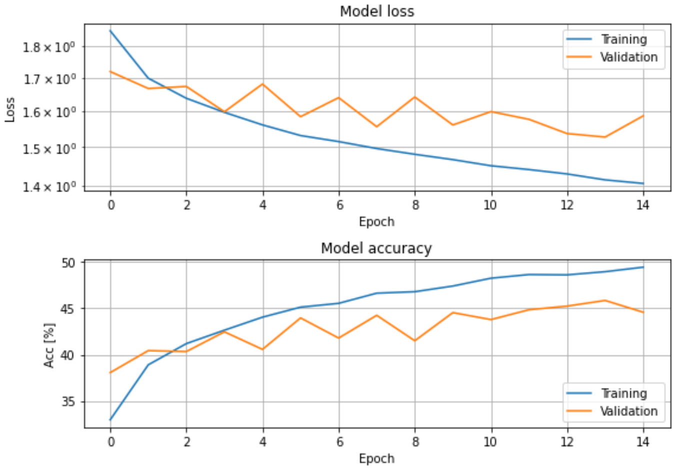
And in **X_test**, there are 10000 image objects(items), with other dimenstions same as in train.

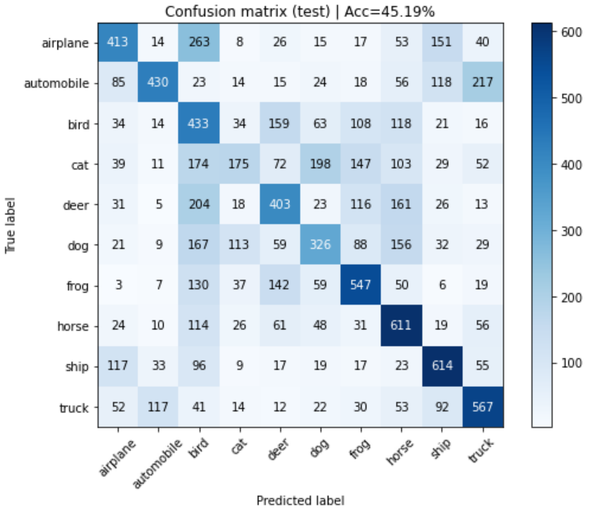
Question 2:

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 comeup 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 balck 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 repesents 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 featrures in it self. Hence the CIFAR10 data is more harder to classify.

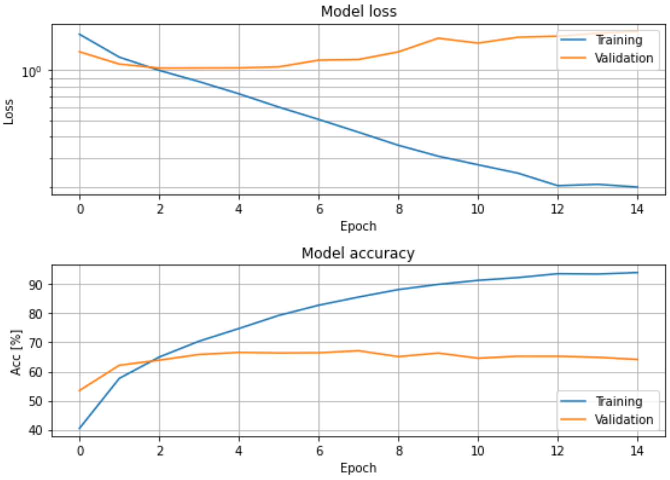
Question 4:

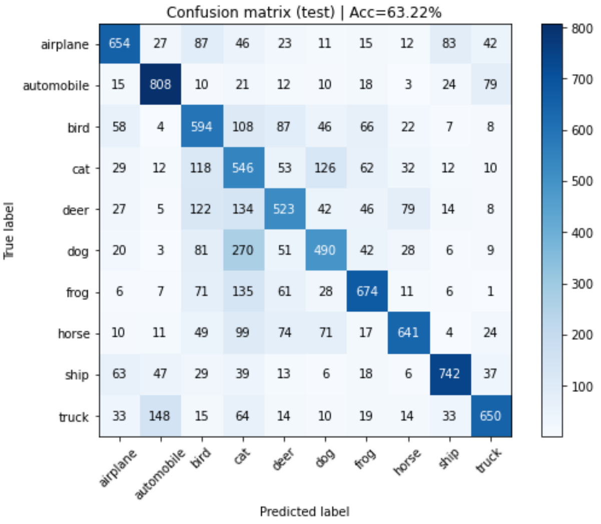
Train a model that achieves at least 62% test accuracy. In the report, provide a (short) description of your model and show the evaluation image.

Answer:

The neural network contains three convolutional layers with the activation function relu, the kernel size used for convolution is a (3x3) matrix. Each convoultion 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 convultions 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.





	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	654	27	87	46	23	11	15	12	83	42
automobile	15	808	10	21	12	10	18	3	24	79
bird	58	4	594	108	87	46	66	22	7	8
cat	29	12	118	546	53	126	62	32	12	10
deer	27	5	122	134	523	42	46	79	14	8
dog	20	3	81	270	51	490	42	28	6	9
frog	6	7	71	135	61	28	674	11	6	1
horse	10	11	49	99	74	71	17	641	4	24
ship	63	47	29	39	13	6	18	6	742	37
truck	33	148	15	64	14	10	19	14	33	650

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.

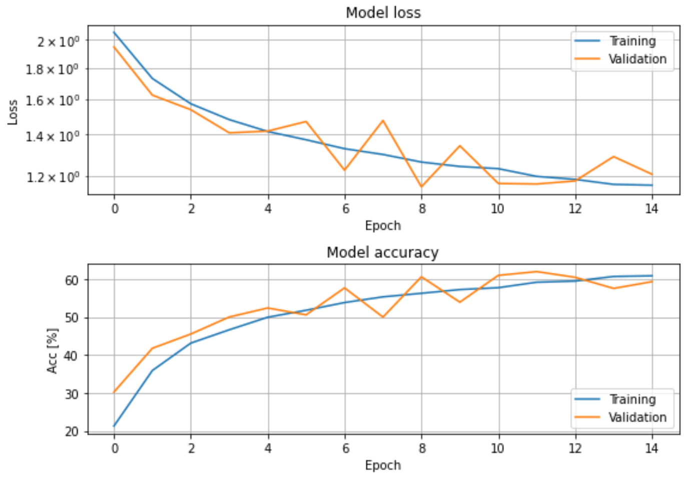
Answer:

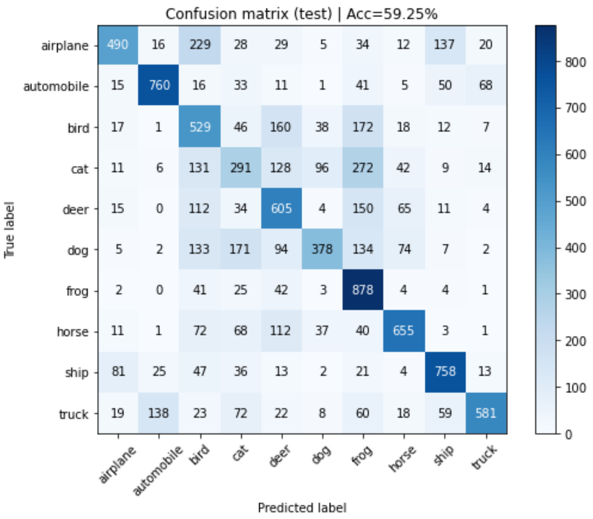
The combination of convolution - activation - pooling is can be seen as the feature learning phase of the neural network. Where network tries to learn the primitive image features such as edges and lines of the dataset which will provide as kind of preprocessing for the neural network to improve the 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:





	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	490	16	229	28	29	5	34	12	137	20
automobile	15	760	16	33	11	1	41	5	50	68
bird	17	1	529	46	160	38	172	18	12	7
cat	11	6	131	291	128	96	272	42	9	14
deer	15	0	112	34	605	4	150	65	11	4
dog	5	2	133	171	94	378	134	74	7	2
frog	2	0	41	25	42	3	878	4	4	1
horse	11	1	72	68	112	37	40	655	3	1
ship	81	25	47	36	13	2	21	4	758	13
truck	19	138	23	72	22	8	60	18	59	581

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:

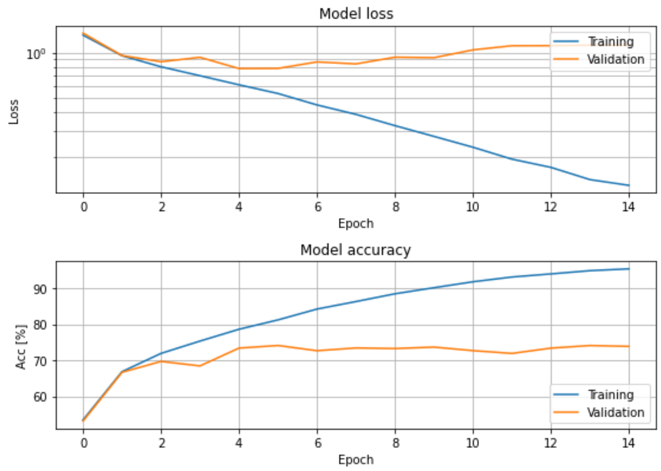
	Training	Validation	Test
CNN	0.9388	0.6414	0.632
CNN-drop	0.6087	0.5933	0.592

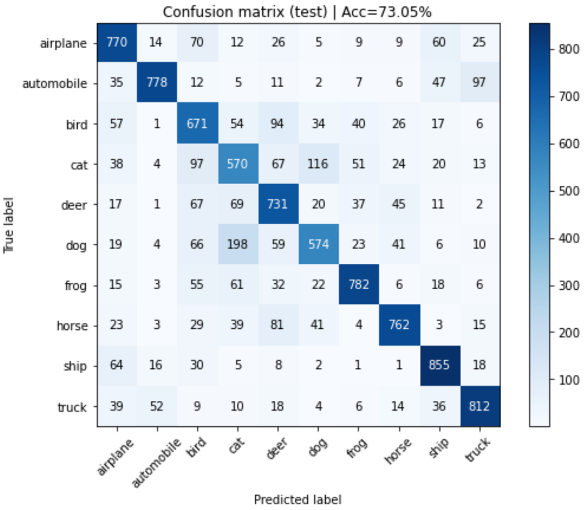
The training accuracy is much higher in the CNN without drop compared to the one with. Using drop, training; validation and test have a similar accuracy. The difference in training to validation and test accuracy for the CNN without drop is because, the model overfits the data for the training due to 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:





	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	770	14	70	12	26	5	9	9	60	25
automobile	35	778	12	5	11	2	7	6	47	97
bird	57	1	671	54	94	34	40	26	17	6
cat	38	4	97	570	67	116	51	24	20	13
deer	17	1	67	69	731	20	37	45	11	2
dog	19	4	66	198	59	574	23	41	6	10
frog	15	3	55	61	32	22	782	6	18	6
horse	23	3	29	39	81	41	4	762	3	15
ship	64	16	30	5	8	2	1	1	855	18
truck	39	52	9	10	18	4	6	14	36	812

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 in equation $y_i = \gamma \hat{x}_i + \beta$, where $\hat{x}_i = 0$ and only β (intercept) term will remain. Only intercept term going as intput to the next layer will not be the desired results. Therefore 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 as the gradient descent algorithm may get stuck in local minima. On the other hand, for larger batch sizes the batch normalisation statistics (mean, variance) will be too closer to the true training data, this also makes the parameters of gradient descent closer to the estimates over entire training data. Along with the normalisation statistics, the learning parameters α and β will have less variation through each iteration and be closer to learned on the entire training data which might not give good result when there is a change in data which will lead to poor regularization.

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.

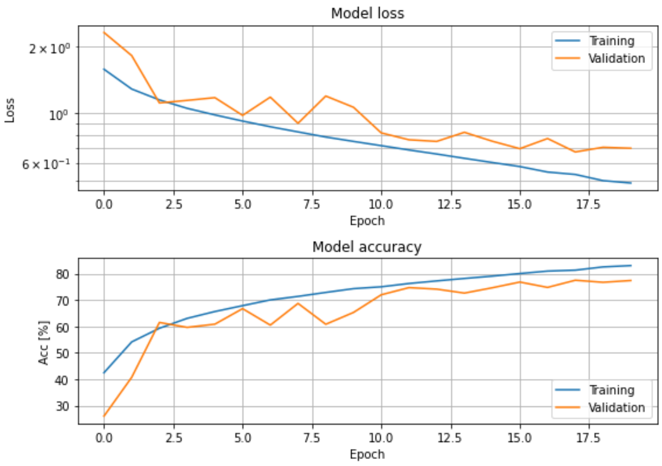
Answer:

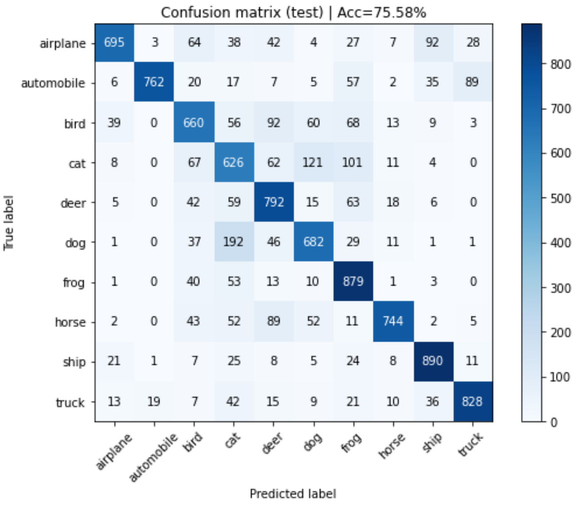
The design of the model is an extension of the previous model from CNN now with 4 convuloutional layers with 32,64,128 and 256 filters each. The chosen paramaters are as for previous models. After each convolutional layer batchnormalization is performed with 32 as the min-batchsize. Then activation with 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 and 32 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





	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	695	3	64	38	42	4	27	7	92	28
automobile	6	762	20	17	7	5	57	2	35	89
bird	39	0	660	56	92	60	68	13	9	3
cat	8	0	67	626	62	121	101	11	4	0
deer	5	0	42	59	792	15	63	18	6	0
dog	1	0	37	192	46	682	29	11	1	1
frog	1	0	40	53	13	10	879	1	3	0
horse	2	0	43	52	89	52	11	744	2	5
ship	21	1	7	25	8	5	24	8	890	11
truck	13	19	7	42	15	9	21	10	36	828

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. Addding 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 number of non-linear operations which improves the classification.