CSA 250 Deep Learning

Project-2

System Specifications

Python version=3.7.3
Pytorch Version=1.3.1

Observations

Multilayer neural network

In multilayer neural network I have tried following things with given hyper parameters

Multilayer perceptron	learning rate	Activation functions	Average Loss	Accuracy	hidden layers	other info
1	0.002	ReLU	0.692	88.28	3	logsoftmax was used in the end
2	0.002	ReLU	4.727	83.19	3	softmax was used in the end
3	0.001	ReLU	0.582	88.56	3	logsoftmax was used in the end
4	0.001	Tanh	0.553	88.57	3	logsoftmax was used in the end
5	0.001	ReLU	0.535	88.89	6	logsoftmax was used in the end
6	0.001	ReLU	1.23	61.02	6	without log softmax
7	0.001	ReLU	0.665	87.83	6	logsoftmax
8	0.001	ReLU	0.494	88.39	6	logsoftmax

Table 1- Results of various iteration with different hyperparameters

In the above table it following things were inferred and the final model was selected based on them.

1.Activation function- Model is trained with Tanh and ReLU activation functions (I.e. 3,4) while keeping all the other parameters fixed. Both ReLU and Tanh are giving almost similar accuracies.

2.Learning Rate- learning rate of 0.002 and 0.001 are used with Adam optimizer. Learning rate of 0.001 is leading to better optimum than learning rate of 0.002.

3. Classifying function on the last layer- Logsoftmax and softmax functions are tried. Classifier is giving better accuracy with logsoftmax as the ouput is not getting saturated in that case.

4. Number of layers- Increasing number of hidden layers improves the accuracy of classifier but at the cost of more computations and more number of parameters.

Optimizer	Error	Classifying Accuracies	Learning rate
Adam	0.535	88.89	0.001
SGD	0.385	86.31	0.01

Table 2- Classification results with various optimizers

5.Optimizer-SGD and Adam Optimizers with enough epochs. Adam optimizer is reaching to better optimal value than SGD method.

Convolutional neural network

In convolutional neural network the below things were tried.

Optimizer	learning rate	Error	Accuracy
Adam	0.001	0.32157	93.13
SGD	0.01	0.523	80.56

Table 3- Classification results with various optimizers for same number of epochs

1) Optimizer- For the same number of epochs and with optimal parameters, Adam optimizer gives better accuracy than SGD.

Layers	Convolutional	Full connected	Error (NLL)	Accuracy
4	2	2	0.246	92.57
5	3	2	0.328	93.42
6	4	2	0.322	93.13

Table 3- Classification results with different number of convolutional layers

2) Number of layers- CNN network with 2,3 and 4 number of convolutional layers is tested with 2 linear layers and it is found that accuracy of classifier is best for 3 number of convolutional layers.

3)Activation function- As we have seen in the case of multi-layer network, ReLU and Tanh both are providing almost equivalent results if there is no saturation happening in the network. Here I used ReLU as activation function

4)learning rate- Leaning rate of 0.001 is used with Adam optimizer.

Final Model Selection

Multilayer neural network

According to above observations the final model is selected with 6 hidden layer, ReLU activation function, Adam optimizer with 0.001 learning rate.

Layer Number	Input	output
1	784	256
2	256	128
3	128	64
4	64	32
5	32	16
6	16	10

Table 4-Number of input and output to the layers of multi-layer neural network

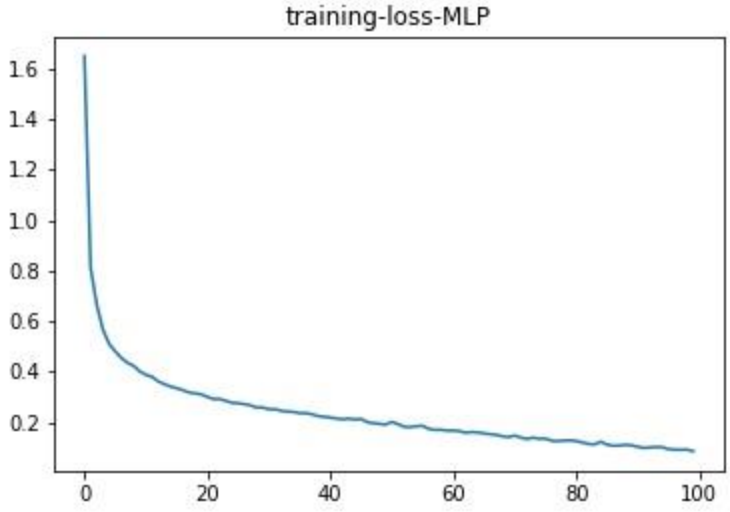

Fig 1- Decrease in error with number of epochs for multi-layer neural network for 100 epochs

Convolutional neural network

According to above observations the final model is selected with 6 layer (4 convolutional and 2 linear), ReLU activation function, Adam optimizer with 0.001 learning rate.

Kernel size of 5 with zero padding =2 is used. Maxpooling of size=2 & stride=2 is used after every convolutional layer.

Layer Number	In channel	Out channel	Kernal size	Zero padding	Maxpool kernal size	Maxpool stride
1	1	32	5	2	2	1
2	32	64	5	2	2	1
3	64	128	5	2	2	2
4	128	256	4	2	2	2

Table 4-Number of input and output to the convolutional layers of convolutional neural network

Layer Number	Input	output
5	7*7*256	1000
6	1000	10

Table 5-Number of input and output to the linear layers of convolutional neural network

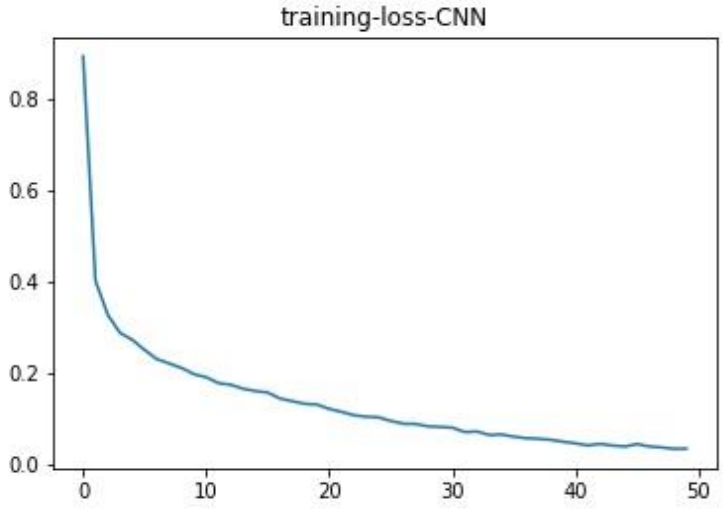

Fig 2- Decrease in error with number of epochs for multi-layer neural network for 50 epochs