

- parameter setting 및 모델 설계

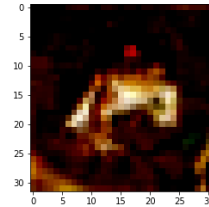
: data – train 50000 & test 10000 from CIFAR10

- input 3*32*32 images, class 10개

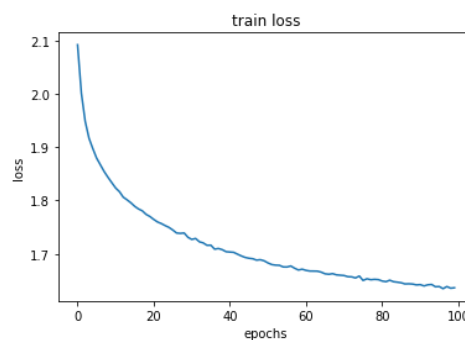
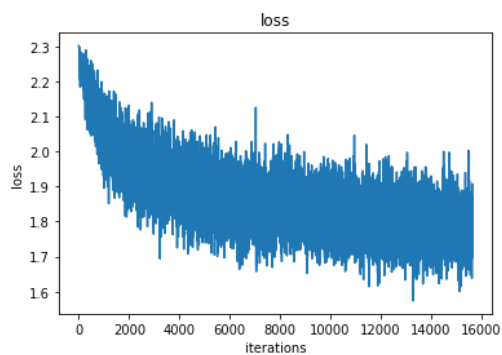
: xavier init for conv2d & linear

: 정규 init for batchnorm2d

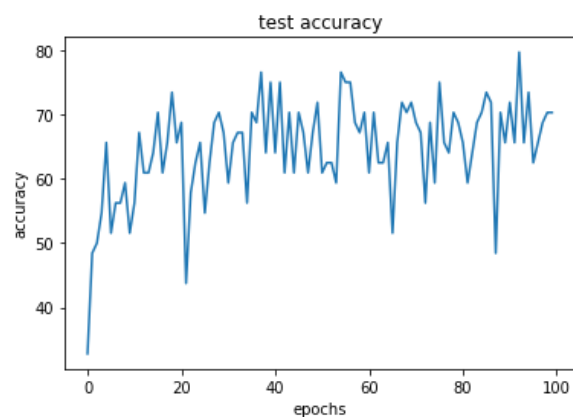
: epoch == [20,40,100] lr == default for adam, batch size == 64



- Model error curve (for training loss)



- accuracy curve



- 모델 summary

```
Test(  
  (layer1): Sequential(  
    (0): Conv2d(3, 6, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))  
    (1): BatchNorm2d(6, eps=1e-05, momentum=0.1, affine=True,  
track_running_stats=True)  
    (2): ReLU()  
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1,  
ceil_mode=False)  
  )  
  (layer2): Sequential(  
    (0): Conv2d(6, 16, kernel_size=(3, 3), stride=(1, 1), padding=(1,  
1))  
    (1): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True,  
track_running_stats=True)  
    (2): ReLU()  
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1,  
ceil_mode=False)  
  )  
  (layer3): Sequential(  
    (0): Conv2d(16, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,  
1))  
    (1): BatchNorm2d(32, eps=1e-05, momentum=0.1, affine=True,  
track_running_stats=True)  
    (2): ReLU()  
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1,  
ceil_mode=False)  
  )  
  (fc1): Linear(in_features=512, out_features=120, bias=True)  
  (fc2): Linear(in_features=120, out_features=84, bias=True)  
  (fc3): Linear(in_features=84, out_features=10, bias=True)  
)
```

- 모델 중간 사이즈

```
torch.Size([64, 3, 32, 32])  
torch.Size([64, 6, 16, 16])  
torch.Size([64, 16, 8, 8])  
torch.Size([64, 32, 4, 4])  
torch.Size([64, 512])  
torch.Size([64, 120])  
torch.Size([64, 84])  
torch.Size([64, 10])
```

test 할 경우

```
torch.Size([10000, 3, 32, 32])  
torch.Size([10000, 6, 16, 16])  
torch.Size([10000, 16, 8, 8])  
torch.Size([10000, 32, 4, 4])  
torch.Size([10000, 512])  
torch.Size([10000, 120])  
torch.Size([10000, 84])
```

```
torch.Size([10000, 10])
```

- 학습 결과의 일부

```
Epoch [9/10], Step [6401/50048], Loss: 1.7524, Accuracy: 65.6%  
Epoch [9/10], Step [12801/50048], Loss: 1.8655, Accuracy: 71.9%  
Epoch [9/10], Step [19201/50048], Loss: 1.8129, Accuracy: 64.1%  
Epoch [9/10], Step [25601/50048], Loss: 1.7790, Accuracy: 60.9%  
Epoch [9/10], Step [32001/50048], Loss: 1.8075, Accuracy: 57.8%  
Epoch [9/10], Step [38401/50048], Loss: 1.7916, Accuracy: 62.5%  
Epoch [9/10], Step [44801/50048], Loss: 1.7833, Accuracy: 67.2%  
Epoch [10/10], Step [6401/50048], Loss: 1.6860, Accuracy: 65.6%  
Epoch [10/10], Step [12801/50048], Loss: 1.7446, Accuracy: 70.3%  
Epoch [10/10], Step [19201/50048], Loss: 1.8081, Accuracy: 51.6%  
Epoch [10/10], Step [25601/50048], Loss: 1.7854, Accuracy: 64.1%  
Epoch [10/10], Step [32001/50048], Loss: 1.7000, Accuracy: 64.1%  
Epoch [10/10], Step [38401/50048], Loss: 1.8745, Accuracy: 65.6%  
Epoch [10/10], Step [44801/50048], Loss: 1.7909, Accuracy: 62.5%
```

- 결과

: cifar10을 학습하는 것에 20 epoch정도면 괜찮다.

: 약 20 에포크가 넘어가면 validation 성능은 더 증가하지 않는다.

```
Accuracy of the network on the 50000 test images 83.068%
```

```
Accuracy of the network on the 10000 test images 68.29%
```

(30번째 정확도, 40번째 정확도가 여전히 test데이터에서 샘플링한 데이터의 60~70사이를 왔다갔다 한다.)

: conv2d의 채널을 더 늘려보는 것도 좋을 것 같고, linear 레이어를 2개만 해도 추가해도 될 것 같다.