Yuguang Yao
CSE

Homework 4

CSE 803
November 2, 2021

1 Optimization and Fitting

1. The implementation of the forward and the backward pass is completed.

2. The implementation of the gradient descent is completed. I chose the learning rate as 1e-4. The figure is shown in Figure 1:

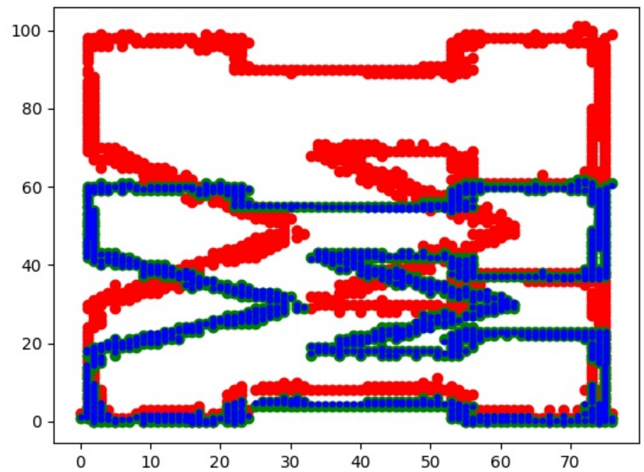

Figure 1: case.jpg

2 Softmax Classifier Plus 1-layer NN

1. The implementation of ReLU layer and softmax layer is finished. The implementation of the softmax classifier is finished.

2. Hyperparameters and train accuracy vs. validation accuracy.

   I use the default hyperparamters for this single layer, i.e., learning_rate = 5e-3, lr_decay=0.9, num_epochs=60, batch_size=256, reg = 0.0.

   Below Figure 7 is the training vs val curve.

   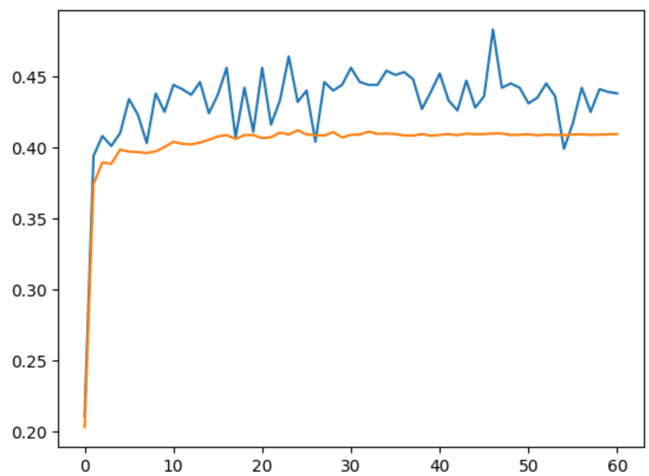

   Figure 2: train vs val

   x axis is epoch. y axis is accuracy. Orange is validation. Blue is training.

3. Test set accuracy evaluation.

   Test accuracy: 41.07%

4. Ablation studies for hyperparamters. We use the default hyperparameters as the baseline and do the ablation study over different hyperparamters. First again, the default hyperparamters are as follwoing:

   the default hyperparamters for this single layer, i.e., learning_rate = 5e-3, lr_decay=0.9, num_epochs=30, batch_size=256, reg = 0.0.

   Then, we ablate each hyperparamter one by one to do the ablation study and experiment over different hyperparamters training the CIFAR10.

(a) Discussion over the regularizer;

Test Acc with Various Reg Coeff

Regularization Coeff	Test Acc
0.0	0.4098
0.001	0.4103
0.01	0.4101
0.1	0.4108

Figure 3: reg

we can see that the training performance is not sensitive to the choice of regularizer.

(b) Discussion over the batch-size;

Test Acc with Various Batch Size

Batch Size	Test Acc
128	0.4125
256	0.4111
512	0.4059
1024	0.3997

Figure 4: batch size

we can see that with a larger batch_size in this model, the test acc is worse.

(c) Discussion over the learning-rate;

Test Acc with diff. LR

Learning Rate	Test Acc
1e-3	0.409
5e-3	0.4088
1e-2	0.4078
5e-2	0.3968

Figure 5: LR

we can see that when the learning rate is too large, the test acc is low because the large LR may lead to underfitting.

(d) Discussion over the LR decay;

Test Acc with diff. LR decay

LR Decay	Test Acc
1	0.3988
0.99	0.4051
0.9	0.4092
0.8	0.4108

Figure 6: decay

we can see the magic of the learning rate decay, which means after epoch the learning rate should be tuned smaller for better fine-grained optimization. Otherwise without learning rate decay, the test acc is very low.

3 Softmax Classifier with Hidden Layers

1. The Implementation of the NN is finished.

2. Hyperparameters and train accuracy vs. validation accuracy.

   I chose 0 regularizer coeff, 128 batch size, 0.1 learning rate, 0.9 lr decay, 100 hidden dim. 30 epoches for training

   The training val curve is below:

   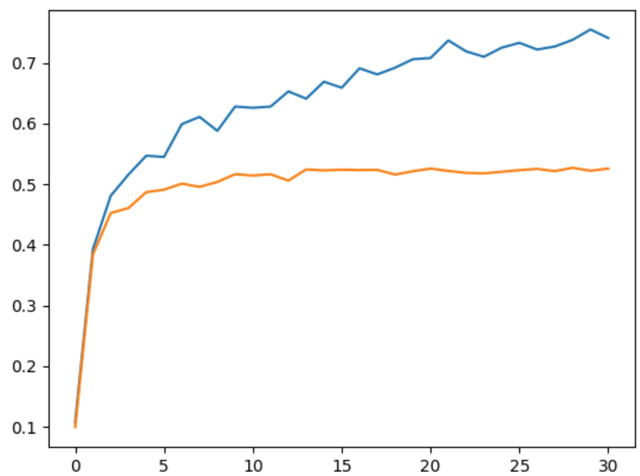

   Figure 7: train vs val

   The axis meaning and the color is as last problem.

3. Test set accuracy evaluation.

   Test accuracy is 51.4%

4. Ablation studies for hyperparamters.

   Then, we ablate each hyperparamter one by one to do the ablation study and experiment over different hyperparamters training the CIFAR10.

   (a) Discussion over the hidden dim:

Test Acc with Various Hidden Dim

Hidden Dim	Test Acc
50	0.503
100	0.514
150	0.518
200	0.5272

Figure 8: hidden dim

we can see that the training performance really depends on the dim of the hidden layer. when the hidden layer size is larger, we can possibly get even better test accuracy.

(b) Discussion over the batch-size;

Test Acc with Various Batch Size

Batch Size	Test Acc
128	0.514
256	0.5052
512	0.4946
1024	0.4696

Figure 9: batch size

we can see that with a larger batch_size in this model, the test acc is worse.

(c) Discussion over the learning-rate;

Test Acc with diff. LR

Learning Rate	Test Acc
0.2	0.4944
0.1	0.514
0.05	0.4996
0.01	0.4621

Figure 10: LR

we can see that when the learning rate is too large or too low, the test acc is low because the large LR may lead to underfitting. This teaches me that we need to finetune the learning rate for SGD carefully.

(d) Discussion over the LR decay;

Test Acc with diff. LR decay

LR Decay	Test Acc
1	0.4844
0.99	0.4941
0.9	0.514
0.8	0.5024

Figure 11: decay

we can see the magic of the learning rate decay again, which means after epoch the learning rate should be tuned smaller for better fine-grained optimization. Otherwise without learning rate decay, the test acc is very low. Also, you should tune the decay carefully, either too large or too small may lead to underfitting.

4 Fooling Images: Adversarial Examples for the NN in the last section

1. The gradients w.r.t. the input are implemented in the code.

2. Target PGD attack implementation. Ascent gradient is implemented in the code.

3. Discussions over the robustness of the model. It is obvious that my model is not adversarially robust at all considering such a small imperceptible perturbation can lead to our model's misclassification over a previously correctly classified image. In detail, the correct truck image is original correctly classified. After one-step target attack, the resulted image is fooling the model to classifiy the image as automobile.

   The difference norm is 91 in the scale of 3X32X32 [0,255] pixel-value image, which is really small, and the figure is black due to almost 0 pixel value.

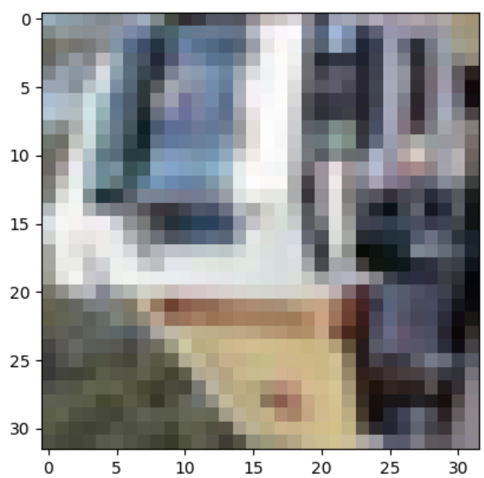

Figure 12: correct

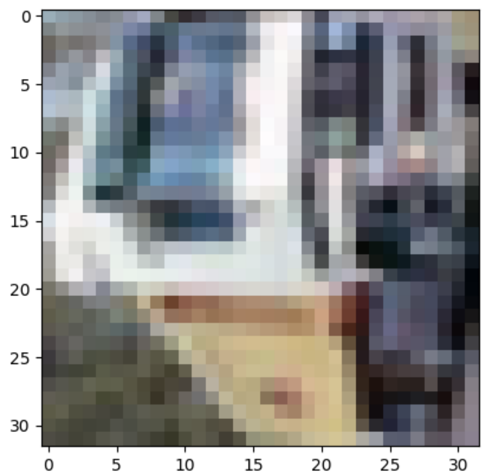

Figure 13: fooled

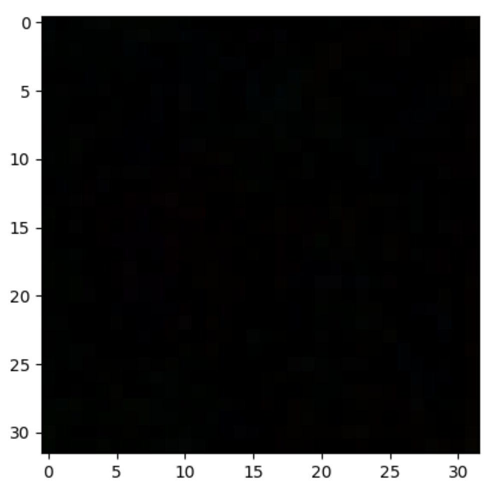

Figure 14: diff