

Name(s): Adel Hamad Alkhamisy
NetID(s):aalkhami
Team name on Kaggle leaderboard:aalkhami

For each of the sections below, your reported test accuracy should approximately match the accuracy reported on Kaggle.

Perceptron

I used many values for hyperparameters (learning rate and epochs) tuning but the values that show the trend in accuracy are presented in table1. The highest accuracy score achieved at learning rate of 0.5 and epochs=10. The accuracies in experiments 2-4 in table1 were fluctuated between 28.17-27.08 on test data, which were low when compared with results at experiment 1. In experiment 5, the results on test data was approximately the same as experiment 1 (29.44 and 29.37). I have implemented the learning rate decay using two approaches which were Exponential Decay and 1/t decay with decay_rate of 0.5. However, the testing accuracy was dropped significantly and fluctuated between 17.9-18.72. I believed that learning rate decay typically increased accuracy, but given the time constraints of this assignment, I might not have been able to discover an appropriate decay rate.

Table1

no	lr	epochs	Training acc	Validation acc	Test acc	Kaggle score
1	0.5	10	33.62	29.42	29.37	0.29
2	0.05	20	32.98	28.27	28.17	-
3	0.005	30	32.45	27.60	27.96	-
4	0.0005	40	32.41	27.28	27.08	-
5	0.6	50	35.81	29.44	29.44	0.29

CIFAR DATASET

Optimal hyperparameters:	lr=0.5, epochs=10
Training accuracy:	33.62
Validation accuracy:	29.42
Test accuracy:	29.37

SVM

The hyperparameter tuning I tried is showed the table2. I have increased the learning rate (lr), epochs, and regularization constant (reg_const) until I found the peak in scores at experiment 4 (highlighted in yellow). In experiment 4, I achieved the highest score compared to other experiments in table1, however, the last submission to Kaggle before the deadline was at experiment 3 (highlighted in green).

I implemented learning rate decay using Exponential decay and 1/t decay, with initial lr=0.05 and 0.5 and decay_rate of 0.005, 0.5, 0.9 but the training accuracy ranged between 27-16 which were not optimal compared to accuracy without learning rate decay. However, in my opinion I thought that the learning rate decay usually improves accuracy, but due to the time limit of this homework, I may not find the good decay_rate.

Table2

no	lr	epochs	reg_const	Training acc	Validation acc	Test acc	Kaggle score
1	0.5	10	0.5	12.61	13.01	12.57	0.12
2	0.05	20	0.05	27.11	25.86	25.80	0.28
3	0.0005	30	0.005	33.77	29.21	28.89	0.28
4	0.0005	30	0.05	34.28	30.34	30.15	0.30
5	0.0005	30	0.005	31.78	27.95	27.32	-

CIFAR DATASET

Optimal hyperparameters:	lr=0.005, epoch=30, reg_const=0.005
Training accuracy:	33.77
Validation accuracy:	29.21
Test accuracy:	28.89

Softmax

I experimented with a variety of values for the hyperparameters: learning rate, epochs and reg_const, but table3 presents the values that best represent the accuracy trend. The maximum accuracy score was attained with a learning rate of 0.0005 and several epochs of 50 and reg_const of 0.05 (highlighted in green). We can conclude that as the learning rate and number of epochs increased from 0.5 and 10 respectively, the accuracy increased until it each a peak at experiment 5. Then, the accuracy start to drop slowly as we increased learning rate and epochs. I used same approaches in implementing learning rate decay as in Perceptrons and SVM: exponential decay and 1/t decay using a decay rate of 0.5. The testing accuracy, however, dramatically declined.

Table3

no	lr	epochs	reg_const	Training acc	Validation acc	Test acc	Kaggle score
1	0.5	10	0.5	17.59	17.57	17.82	0.12
2	0.05	20	0.05	26.16	26.21	26.00	-
3	0.005	30	0.05	28.37	26.07	25.72	0.25
4	0.0005	40	0.05	31.51	27.57	28.11	0.28
5	0.0005	50	0.05	33.73	28.26	28.90	0.29
6	0.0005	60	0.05	32.91	27.29	28.07	-

CIFAR DATASET

Optimal hyperparameters:	
Training accuracy:	
Validation accuracy:	
Test accuracy:	

Reference:

1. <https://cs231n.github.io/optimization-1/>
2. <https://www.deeplearningbook.org/>

Appendix A: