Name(s): Ruoyi Feng, Bin Peng
NetID(s):ruoyif2, binpeng2
Team name on Kaggle leaderboard: TT

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

Briefly describe the hyperparameter tuning strategies you used in this assignment. Then record your optimal hyperparameters and test/val performance for the four different network types.

Two-layer Network Trained with SGD

Best hyperparameters (if you changed any of the other default hyperparameters like initialization method, etc. please note that as well):

Batch size:	200
Learning rate:	1e-2
Hidden layer size:	60
Regularization coefficient:	0.01

Record the results for your best hyperparameter setting below:

Validation accuracy:	0.51
Test accuracy:	0.5029

First, I run with default parameter setting.
Then I notice loss and accuracy converges at early stage, so I set epoch to 80.
When learning rate is 1e-3, it decays to 1e-5 after 80 epochs. It means gradients are not updated, so we increase learning rate 1e-2.
Conduct grid search for following parameters setting:
Regularization coefficient = (0.001 0.01, 0.1)
Hidden layer size = (20, 60, 80)
Finally we get optimal parameter setting.

Three-layer Network Trained with SGD

Best hyperparameters (if you changed any of the other default hyperparameters like initialization method, etc. please note that as well):

Batch size:	200
Learning rate:	1e-2
Hidden layer size:	60
Regularization coefficient:	0.01

Record the results for your best hyperparameter setting below:

Validation accuracy:	0.516
Test accuracy:	0.5001

First, I run with default parameter setting.
Then I notice loss and accuracy converges at early stage, so I set epoch to 80.
When learning rate is 1e-3, it decays to 1e-5 after 80 epochs. It means gradients are not updated, so we increase learning rate 1e-2.
Conduct grid search for following parameters setting:
Regularization coefficient = (0.001 0.01, 0.1)
Hidden layer size = (20, 60, 80)
Finally we get optimal parameter setting.

Two-layer Network Trained with Adam

Best hyperparameters (if you changed any of the other default hyperparameters like initialization method, etc. please note that as well):

Batch size:	200
Learning rate:	1e-2
Hidden layer size:	60
Regularization coefficient:	0.01
β_1	0.9
β_2	0.999

Record the results for your best hyperparameter setting below:

Validation accuracy:	0.6352499999999996
Test accuracy:	0.4965

Since adam itself can adjust learning rate, we set learning rate decay = 0
First, I run with default parameter setting.
Then I notice loss and accuracy converges at early stage, so I set epoch to 80.
When learning rate is 1e-3, it decays to 1e-5 after 80 epochs. It means gradients are not updated, so we increase learning rate 1e-2. But as result shows, gradient changes too dramatically that it drops very quickly and don't converge well. So we still set learning rate to 1e-3.
Conduct grid search for following parameters setting:
Regularization coefficient = (0.001 0.01 0.1)
Hidden layer size = (20, 60, 100)
Finally we get optimal parameter setting.

Three-layer Network Trained with Adam

Best hyperparameters (if you changed any of the other default hyperparameters like initialization method, etc. please note that as well):

Batch size:	200
Learning rate:	0.001
Hidden layer size:	60
Regularization coefficient:	0.01
β_1	0.9
β_2	0.999

Record the results for your best hyperparameter setting below:

Validation accuracy:	0.6921750000000003
Test accuracy:	0.5117

Since adam itself can adjust learning rate, we set learning rate decay = 0
First, I run with default parameter setting.
Then I notice loss and accuracy converges at early stage, so I set epoch to 80.
When learning rate is 1e-3, it decays to 1e-5 after 80 epochs. It means gradients are not updated, so we increase learning rate 1e-2. But as result shows, gradient changes too dramatically that it drops very quickly and don't converge well. So we still set learning rate to 1e-3.
Conduct grid search for following parameters setting:
Regularization coefficient = (0.001 0.01 0.1)
Hidden layer size = (20, 60, 100)
Finally we get optimal parameter setting.

Comparison of SGD and Adam

Attach two plots, one of the training loss for each epoch and one of the validation accuracy for each epoch. Both plots should have a line for SGD and Adam. Be sure to add a title, axis labels, and a legend.

Compare the performance of SGD and Adam on training times and convergence rates. Do you notice any difference? Note any other interesting behavior you observed as well.

As the graph shows, the training times required to converge of SGD is larger than Adam. Validation accuracy of Adam is much better than SGD, however, the training loss of Adam is much larger than SGD.