Project: More Experiments on Stochastic Gradient Methods
Goal

- We want to know more the internal details of simpleNN
- We want to roughly compare the two stochastic gradient approaches: SG with momentum and Adam
In our code, stochastic gradient is implemented in a subroutine `gradient_trainer` in `train.py`. You can see a `for` loop there.

It happened that we run the SG steps by ourself, but in Tensorflow there must be a way so that stochastic gradient methods can be directly called in one statement.

We would like to check if under the same initial model, the two settings give the same results.
To check “the same results” you can, for example, compare their models after some iterations or compare their objective values.

One issue is that in the beginning of each update, we randomly select instances as the current batch:

\[
\text{idx} = \text{np.random.choice(}
\hspace{1cm}
\text{np.arange(0, num_data),}
\hspace{1cm}
\text{size=config.bsize, replace=False)}
\]

Tensorflow doesn’t do that so you can replace the code with
idx = np.arange(i*config.bsize, (i+1)*config.bsize)

- The regularization term may be a concern. Need to make sure that the two settings minimize the same objective function
- Another interesting issue is that we load data in MATLAB format and run Tensorflow
- The reason is for the simultaneous development of the MATLAB code
- Please investigate what the most common way people used to load data in Tensorflow
Then check again that results of running stochastic gradient methods are the same

What are your thoughts and suggestions in supporting input formats other than MATLAB?

For the investigation so far, accuracy is not important, so you can just run a few iterations to save time

In the second part of this project we want to check the test accuracy of two stochastic gradient methods: SG with momentum and Adam
Note that in the first project, what we used is the simplest SG without momentum.
We also hope to roughly check the parameter sensitivity.
Under each parameter setting, we run a large number (e.g., 500) of iterations and use the model at the last iteration.
We do not use a model before the last iteration because a validation process was not conducted.
Vary parameters (e.g., learning rate in SGD and Adam) and check the test accuracy.
Please work on the same MNIST and CIFAR10 data sets used in the previous project.

In your report, give your observations and thoughts.

Due to the lengthy running time, no need to try many parameter settings.
Students with the following IDs (last three digits):

??

please do a ??-minute presentation (??-minute the contents and ??-minute Q&A)