Autolab for Dummies

Logging In

In your browser, go to https://autolab.andrew.cmu.edu

This will take you to the CMU login page, enter your Andrew ID and password to log in

Carnegie Mellon University

Web Login

AndrewID [   ]
Password [   ]

[Login]

Warning: The URL for this page should begin with **https://login.cmu.edu**.
If it does not, do not fill in any information, and report this site to it-help@cmu.edu.

About | Change Password | Forgot Password?

Once logged in, you should see our course, **11-785: Introduction to Deep Learning (s19)**, listed under Current. **If you are registered for the course but do not see it on autolab, reply to the autolab issues thread on piazza with your Andrew ID.**

Successfully authenticated from Shibboleth account.

Courses

Current

Instructor

11-785:
Introduction to
Deep Learning
(s19)

Homework 0

COURSE PAGE

Under the course, you will find a list of released assignments. To start, select Homework 0

Downloading the Assignment

On the assignment page, you will find a section named **Handouts.**

Handouts

⬇ Handout

⬇ Write-up.pdf

Click on the Handout link to download the starter code package.
Click on the Write-up.pdf link to view the assignment write up and instructions.
Simply follow the instructions in the write up to complete the assignment.

Submitting Solutions

Once you are ready to submit your solutions, follow the instructions in the handout to create the handin tar.

Make sure to agree to the academic integrity policies by checking the affirmation box.

Then, on the assignment page on autolab, use the submit link to select your handin tar.

🕐 Due: **January 19th 2019, 10:43 pm**
📅 Last day to handin: **February 28th 2019, 10:43 pm**

☐ I affirm that I have complied with this course's academic integrity policy as defined in the syllabus.

SUBMIT

(∞ submissions left)

Viewing Your Results

After submitting, wait for the autograder to finish running. This can take up to a couple of minutes, so be patient.

Refresh the page to view your results

Ver	File	Submission Date	Sumproduct Vectorization (10.0)	Relu Vectorization (10.0)	Relu Deravitive Vectorization (10.0)	Fixed Point Slicing (12.0)	Last Point Slicing (12.0)	Random Point Slicing (12.0)	End Padding (12.0)	Centeralized Padding (12.0)	Numpy to Tensor (1.0)	Tensor to Numpy (1.0)	Tensor Sumproduct (4.0)	Tensor ReLU (2.0)	Tensor ReLU Prime (2.0)	Late Days Used	Total Score
32	malikhan@andrew.cmu.edu_32_handin.tar	2018-12-31 12:38:39	10.0	10.0	10.0	12.0	12.0	12.0	12.0	12.0	1.0	1.0	4.0	2.0	2.0	Submitted 0 days late	100.0

Each row represents one of your submissions, and each column represents the score for an individual problem. In the column headings, you will find the name of the problem along with the maximum attainable score for it.

To view the output of the autograder, click on any of the blue scores for the problems

Feedback for Homework 0 - Numpy to Tensor (malikhan@andrew.cmu.edu)

```
Autograder [Mon Dec 31 09:36:00 2018]: Received job 11785-s19_homework0_30_malikhan@andrew.cmu.edu:21
Autograder [Mon Dec 31 09:36:28 2018]: Success: Autodriver returned normally
Autograder [Mon Dec 31 09:36:28 2018]: Here is the output from the autograder:
---
Autodriver: Job exited with status 0
mkdir handin
tar xf handin.tar -C handin
tar xf autograde.tar
cp autograde/setup.cfg handin/setup.cfg
AUTOLAB=1 python3 autograde/runner.py
We are on Autolab
........FFF..                                              [100%]
================================ FAILURES ================================
_____ test_numpy2tensor _____

    def test_numpy2tensor():
        dim1 = np.random.randint(3000)
        dim2 = 3000 - dim1
        X = np.random.randint(-1000, 1000, size=(dim1, dim2))
        tensor_grader = numpy2tensor(X)
        tensor_submission = hw0.numpy2tensor(X)
        assert(type(tensor_grader) == type(tensor_submission))
        assert(tensor_submission.shape == tensor_grader.shape)
>       assert(tensor_submission==tensor_grader).all()
E       AssertionError: assert False
E        +  where False = <built-in method all of torch.ByteTensor object at 0x7fc1e3983808>()
E        +    where <built-in method all of torch.ByteTensor object at 0x7fc1e3983808> = \n -733  -35
E           Full diff:
E
E           -  -733  -354   671 ...    273  -382   815
E           ?    ^     ^     ^          ^     ^     ^
E           +  -732  -353   672 ...    274  -381   816
E           ?    ^     ^     ^          ^     ^     ^
E           -   201  -381   909 ...   -544  -941   199...
E
E           ...Full output truncated (20 lines hidden), use '-vv' to show.all

autograde/tests/test_problems.py:185: AssertionError
```

The output from the autograder is often difficult to interpret, and will vary significantly depending on which problems were failed. Generally, it is possible to get some idea of what is failing.

In the above example, you can see that the test case failed because the solution's *numpy2tensor* output was different from the reference solution.

Note: This will not necessarily always tell you what is wrong, and very rarely will it tell you how to fix it. If you run into a wall, Piazza and TA Office Hours are your friends.

When posting a question regarding a homework on Piazza, make sure you tag it with the appropriate homework number. **This makes it easier for us to answer your question quickly.**