

Advice for mandatory project 1

Work outside of the notebook

It may be a good idea to test your code outside the notebook. Outside the notebook it can be easier to use a *debugger* (use breakpoints) and you can get faster development cycles, since you don't have to swap between an editor and a browser.

To write simple tests for the functions, you can add a block to the bottom of each file you are working on. If you add the lines

```
if __name__ == '__main__':  
    #your test code HERE
```

you can write code that is only executed when you run that file explicitly, for example:

```
$ python softmax.py
```

In other words, that code will not be run when you use your notebook. In this block you can test e.g. the `softmax_loss_naive` function with simple values.

Start simple

Even though we use 500 examples with 10 different classes in the notebook, you don't have to when you test your code. Testing with small dimensions for X and W will make it easier to catch errors when you print out your results.

Do hand calculations

If you are unsure whether your function is correct, you can "hand calculations" or use www.wolframalpha.com to find the solution for a simple set of numbers and compare with your solution.

To make this easier for you we provide some simple input with the correct output for **`softmax_loss_naive`** and hope it will help you debug your solution.

Input and Solution

You can generate the data like this:

```
np.random.seed(1)  
W = np.random.randn(2, 3)  
X = np.random.randn(3, 2)  
y = np.arange(3)
```

This should give the following values:

```
W:  
[[ 1.62434536 -0.61175641 -0.52817175]  
 [-1.07296862  0.86540763 -2.3015387 ]]
```

X:

```
[[ 1.74481176 -0.7612069 ]  
 [ 0.3190391 -0.24937038]  
 [ 1.46210794 -2.06014071]]
```

y:

```
[0, 1, 2]
```

When you run the function with those numbers:
loss, grad = **softmax_loss_naive**(W, X, y, 0.0)

you should get **loss**:
0.997810

and **grad**:

```
[[ 0.33479965 -0.0874167 -0.24738295]  
 [-0.47226825 0.06906057 0.40320768]]
```