

How to Run

This project runs on a Python platform called “iPython” (now called “Jupyter”): an interactive Python interface where logical blocks of code can be executed at the user’s discretion. Markdown and Latex can be embedded inline and code output can be inspected. Python 2.7.11 and pip (Python’s package manager, comes default with Python) must be installed.

Here are the steps to install the required dependencies once you’ve unzipped the project file. All dependencies will be installed in a .env folder that exists in the project folder. If you are working on MacOS, run **start_ipython_osx.sh** instead of **ipython notebook** (step 6).

```
1. cd project1
2. sudo pip install virtualenv      # This may already be installed
3. virtualenv .env                 # Create a virtual environment
4. source .env/bin/activate        # Activate the virtual environment
5. pip install -r requirements.txt  # Install dependencies
6. ipython notebook
7. # Work on the assignment for a while ...
8. deactivate                     # Exit the virtual environment
```

Step 6 will launch the iPython/Jupyter file browser in your web browser, where you can find the individual iPython “notebooks” that contain the answers to each question. Once you’re done checking everything, Step 7 will deactivate the virtual environment, in which afterwards you can delete all of the dependencies by deleting the .env folder.

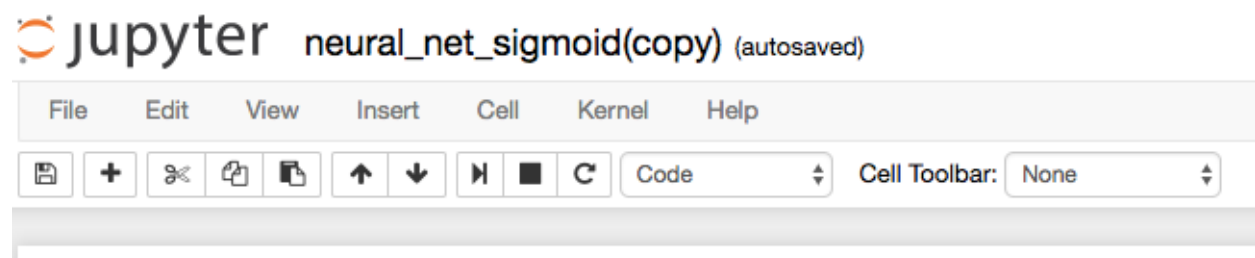
Part I Feature Extraction and Data Virtualization -> pca.ipynb

Part II Linear and Logistic Regression -> linear_and_logistic_regression.ipynb

Part III Support Vector Machines (SVMs) -> svm.ipynb

Part IV Deep Neural Networks (DNNs) -> neural_net.ipynb and neural_net_sigmoid.ipynb

Once you’re in a notebook, you’ll be presented with a combination of code, markdown, and some latex. The markdown and latex is there to provide guidance through the code, and each “cell” of code can be executed by pressing the “play” button on the notebook toolbar. From top to bottom, press play for each cell block and wait for its results. To execute all cell blocks at once, click on Cell -> Run All in the toolbar. I’d advise you to manually execute each cell sequentially instead though.



On the top-left of each code cell there is an indicator “In [number]:” where “number” corresponds to the order you executed the cell block w.r.t. to the other cell blocks and “In” stands for

“input” (shown below). While executing a cell block, “number” will be an asterisk (*) and when the execution has finished “number” will change to its execution number.

Linear Regression and Logistic Regression

In this exercise we will be implementing two types of linear classifiers that will classify between digits. We will learn two 10-class classifiers using linear regression and logistic regression, and report the best performance.

```
In [3]: # some startup
import numpy as np
import src.data_utils as util
import matplotlib.pyplot as plt

%matplotlib inline
plt.rcParams['figure.figsize'] = (10.0, 8.0) # set default size of plots
plt.rcParams['image.interpolation'] = 'nearest'
plt.rcParams['image.cmap'] = 'gray'

# for auto-reloading modules
%load_ext autoreload
%autoreload 2

In [4]: def get_MNIST_data(num_training=50000, num_validation=10000):
        path to dataset = 'data/datasets'
```