

Laboratory of Machine Learning with Python

Numpy / Matplotlib / Scikit-learn

Paolo Dragone

University of Trento



`http://lion0b.disi.unitn.it:9999`

(Only available within the DISI network)

Password: ml-lab2

Setup (on your own machine)

Make sure you are using Python 3 for the following steps.

Install Numpy, Scipy, Matplotlib, Scikit-learn and Jupyter:

```
>> pip install numpy scipy matplotlib sklearn jupyter
```

Download and extract the material for the Scikit-learn lab:

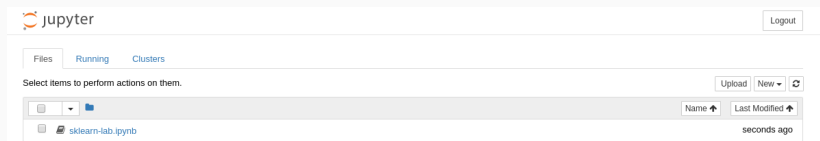
<http://disi.unitn.it/~passerini/teaching/2017-2018/MachineLearning/>

Setup: Jupyter notebook

Open the terminal in the folder containing the extracted archive and run:

```
>> jupyter notebook
```

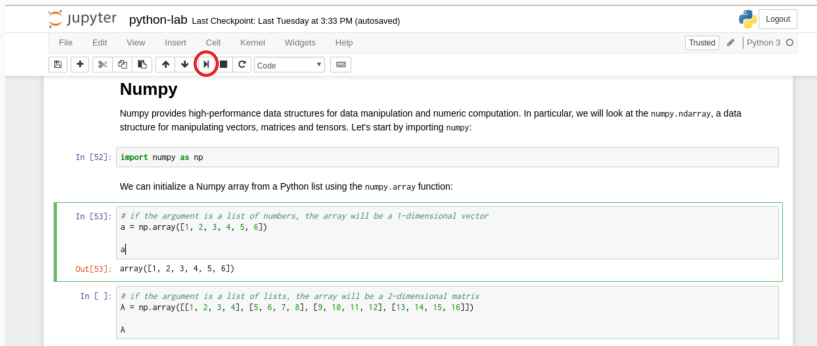
Open the browser at the given address and you'll see something like:



The screenshot shows the Jupyter Notebook web interface. At the top, there's a 'jupyter' logo and a 'Logout' button. Below that, there are tabs for 'Files', 'Running', and 'Clusters'. Under the 'Files' tab, it says 'Select items to perform actions on them.' and there's a list of files. The file 'sklearn-lab.ipynb' is selected. To the right of the file list, there are buttons for 'Upload', 'New', and a refresh icon. Below the file list, there's a table with columns 'Name' and 'Last Modified'. The file 'sklearn-lab.ipynb' is listed with a timestamp of 'seconds ago'.

Open the `sklearn-lab.ipynb` file containing the lecture notebook.

Setup: Jupyter notebook



Execute commands by selecting a cell and clicking the **Run button** on the header of the page or by **Shift+Enter**. You will see the output of the command just below the cell.

You can tweak and modify the code as you wish and execute it again.

For the second Machine Learning assignment you will solve a classification task using **Scikit-learn** over some given dataset. Each available dataset is already split into training and test sets. You have access to the labels of the training examples but the labels of the test set are hidden. Your task is to choose a dataset, train a classifier on the training set and predict the labels on the test set. To pass the assignment, your classifier has to classify the examples in the test set with higher accuracy than the reference baseline for the chosen dataset. Additionally, you need to test your algorithm via cross-validation over the training set and produce a report containing the results obtained.

Assignment — Datasets

OCR

Optical Character Recognition



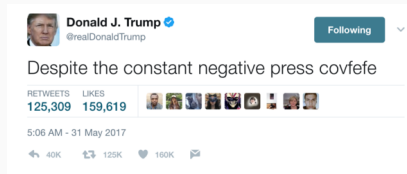
Spambase

Spam email classification



Presidential campaign tweets

Classification of tweets from D. Trump and H. Clinton



Download the assignment material:

<http://disi.unitn.it/~passerini/teaching/2017-2018/MachineLearning/>

The material contains:

- The three datasets, each one containing:
 - The training set examples;
 - The training set labels;
 - The test set examples;
 - A README containing info about the dataset.
this file also contains the reference baseline accuracy;
 - Other info files.
- A helper script;

Assignment — Helper

The helper script can be used to test your predictions. Given a file containing the predicted labels, the helper script sends the labels to our server and receives the prediction accuracy. You can use it in this way:

```
>> ./helper.py your.email dataset test-labels.txt
```

The first parameter is your `unitn` email, the second parameter is the dataset label (one among “ocr”, “spambase” and “tweets”), the third parameter is the path to the file containing the predicted labels. This file should contain one label per line in the same order of the file containing the examples. The labels should be in the same format of the labels in the training set.

The helper also prints the current best accuracy achieved by any of you on that dataset, just to put a bit of healthy competition! :)

Assignment — Step-by-step

1. Choose a dataset;
2. Experiment with a classification algorithm of your choosing;
3. Test your classifier using cross-validation over the training set;
4. Write a report describing the learning algorithm used and discussing the results obtained; The report should contain at least:
 - The average precision, recall, and F_1 over the folds.
Using `cross_val_score` you can specify 'precision', 'recall' and 'f1' for the `scoring` parameter.
For the OCR dataset, in which you do multiclass classification, use weighted averaging, i.e. using 'precision_weighted', 'recall_weighted' and 'f1_weighted';
 - The plot of the learning curve, as shown in the lecture;
5. Train your classifier over the full training set;
6. Use the classifier to predict the examples in the test set;
7. Place the labels in a file, in the same order as you read the test examples and in the same format of the labels in the training set.

- After completing the assignment submit it via email
- Send an email to paolo.dragone@unitn.it (cc: passerini@disi.unitn.it)
- Subject: `sklearnSubmit2017`
- Attachment: `id_name_surname.zip` containing:
 - The text file containing the final predictions;
 - The code used to produce the predictions, the results and the plots;
 - The report in PDF format.

NOTE

- No group work
- This assignment is mandatory in order to enroll to the oral exam