

Laboratory of Machine Learning with Python

Numpy / Matplotlib / Scikit-learn

Luca Erculiani

University of Trento



Download and extract the Scikit-learn lecture material from:

<http://disi.unitn.it/~passerini/teaching/2018-2019/MachineLearning/>

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

```
$> ./jupyter-scikit.sh
```

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:

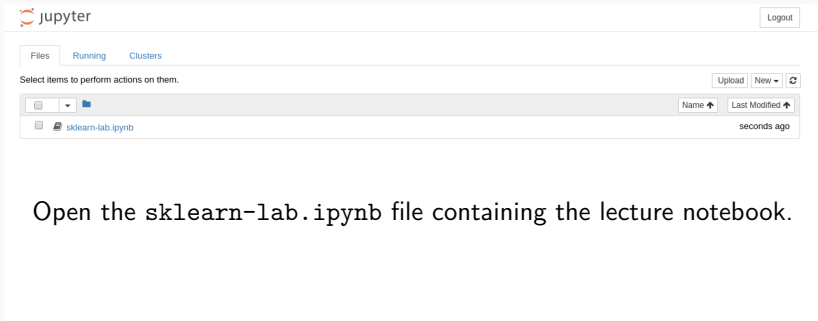
<http://disi.unitn.it/~passerini/teaching/2018-2019/MachineLearning/>

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

```
$> jupyter notebook
```

Setup: Jupyter notebook

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

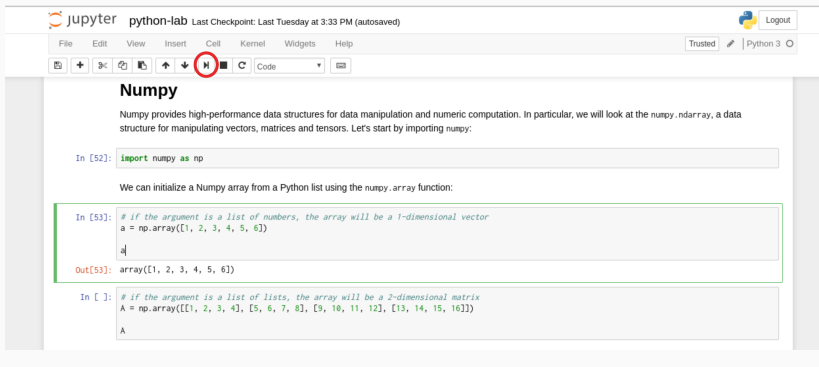


The screenshot displays the JupyterLab web interface. At the top left is the Jupyter logo and the text "jupyter". On the top right is a "Logout" button. Below the header are three tabs: "Files" (selected), "Running", and "Clusters". Under the "Files" tab, there is a prompt "Select items to perform actions on them." and a toolbar with "Upload", "New", and a refresh icon. A file browser table is shown with two columns: "Name" and "Last Modified". The table contains one entry: "sklearn-lab.ipynb" with a "seconds ago" timestamp.

Name	Last Modified
sklearn-lab.ipynb	seconds ago

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

Setup: Jupyter notebook



The screenshot shows a Jupyter Notebook interface for a Python environment. The top bar includes the Jupyter logo, the name 'python-lab', and the last checkpoint information: 'Last Tuesday at 3:33 PM (autosaved)'. On the right, there is a 'Logout' button. Below the top bar is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. A toolbar contains icons for running, undo, redo, and other actions. The 'Run' button (a play icon) is circled in red. The main content area has a title 'Numpy' and a paragraph explaining its use. Below this, there are three code cells. The first cell contains the code `import numpy as np`. The second cell contains `a = np.array([1, 2, 3, 4, 5, 6])` and its output is `array([1, 2, 3, 4, 5, 6])`. The third cell contains `A = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12], [13, 14, 15, 16]])` and its output is `A`.

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. 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;
- The test set labels;
- A README containing info about the dataset.
this file also contains the reference baseline accuracy;
- Other info files.

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. Train your classifier over the full training set;
5. Use the classifier to predict the examples in the test set;
6. 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.
7. 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 cross validation folds and over the test set.

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;

- After completing the assignment submit it via email
- Send an email to `mllab@unitn.it`
- Subject: `sklearnSubmit2018`
- Attachment: `id_name_surname.zip` containing:
 - The text file, named `test-pred.txt`, 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