

Data Mining

ETI

John Samuel

CPE Lyon

Year: 2019-2020

Email: john(dot)samuel(at)cpe(dot)fr



Goals

- Lifecycle of data
 - Data acquisition and storage
 - Data extraction and integration
 - Data pre-processing and transformation
 - Data analysis and visualisation
- Introduction to Machine Learning models
- Text Mining

Course Structure:

- Classes: 8h
- Practical sessions: 16h

Course:

- Operating system: Linux
- Programming Language: Python
- Editor: Jupyter

Class:

- Interactive
- Slides: available in English
- Questions: Every 20-30 mins
- Written final Exam: 60% to grades

Practical Session:

- Grades: 40%
- 4 exercises
- Deadline: 1 week for every exercise
- Online submission

Class	Dates
Class 1	4 th February
Class 2	12 th February (morning)
Class 3	12 th February (afternoon)

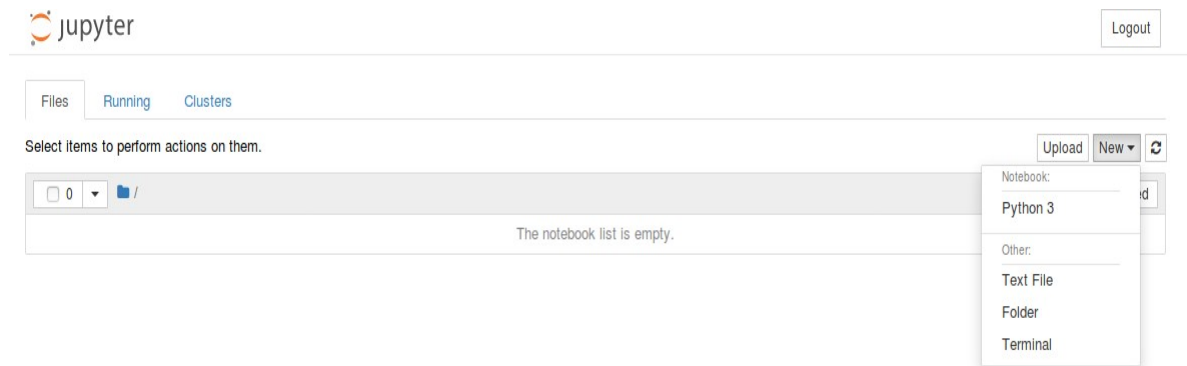
Practical Session	Dates
Practicals 1	5 th February
Practicals 2	24 th February
Practicals 3	12 th March
Practicals 4	16 th March

Practical Sessions

- Pair programming
- Submission of Jupyter Notebook

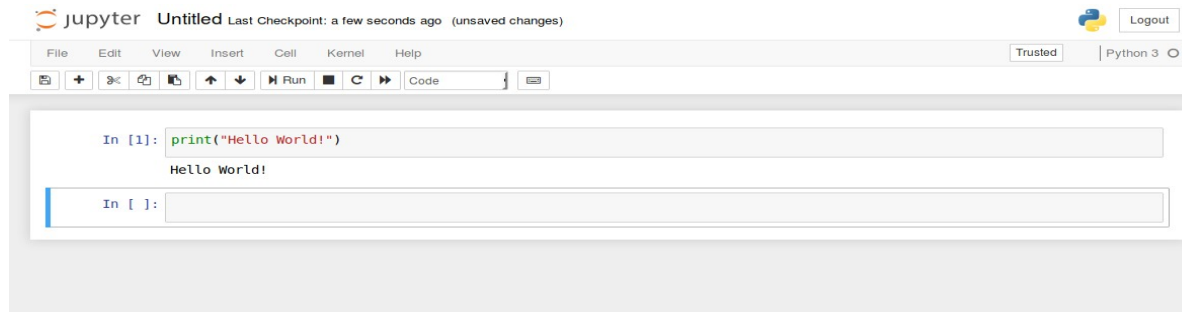


Practical Sessions: Jupyter notebooks



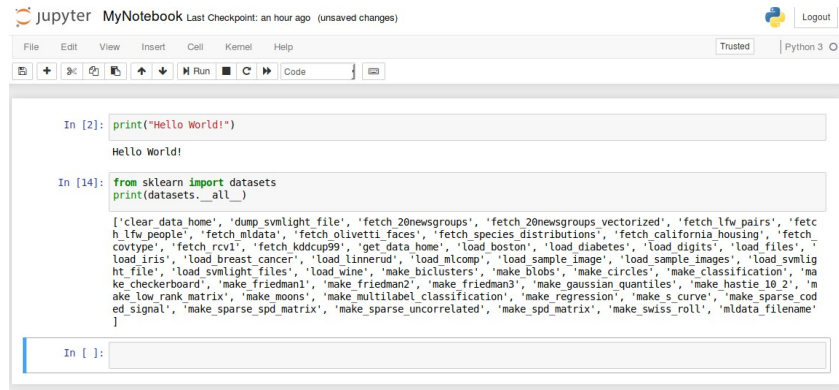
Data Mining: Jupyter Notebooks

Practical Sessions: Jupyter notebooks



Data Mining: Jupyter Notebooks

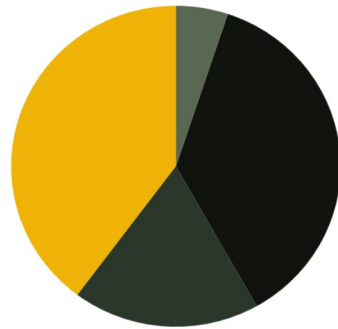
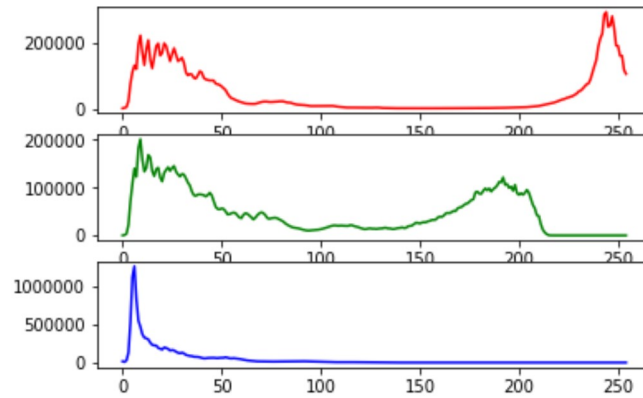
Practical Sessions: Jupyter notebooks



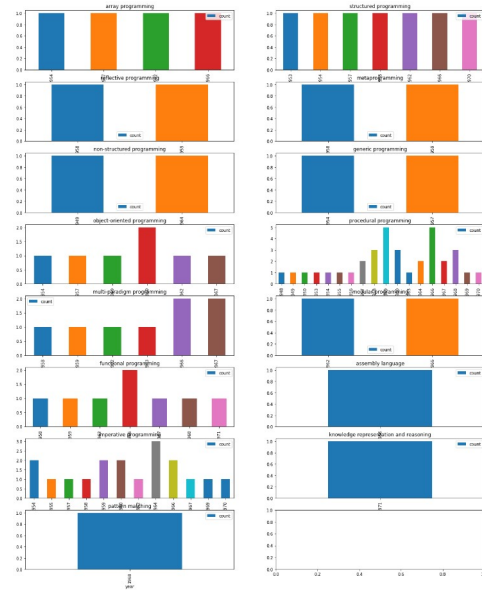
The screenshot shows a Jupyter Notebook interface with the title 'MyNotebook'. The top bar includes a 'Logout' button and a 'Python 3' dropdown. The menu bar contains 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', and 'Help'. Below the menu is a toolbar with icons for file operations, running, and code execution. The notebook content shows two input cells. The first cell, labeled 'In [2]:', contains the code `print("Hello World!")` and its output, 'Hello World!'. The second cell, labeled 'In [14]:', contains the code `from sklearn import datasets` and `print(datasets.__all__)`. The output of this cell is a long list of dataset names available in the sklearn.datasets module, including 'clear_data_home', 'dump_svmlight_file', 'fetch_20newsgroups', 'fetch_20newsgroups_vectorized', 'fetch_lfw_pairs', 'fetch_lfw_people', 'fetch_mldata', 'fetch_olivetti_faces', 'fetch_species_distributions', 'fetch_california_housing', 'fetch_covtype', 'fetch_rcv1', 'fetch_kddcup99', 'get_data_home', 'load_boston', 'load_diabetes', 'load_digits', 'load_files', 'load_iris', 'load_breast_cancer', 'load_linnerud', 'load_mldata', 'load_sample_image', 'load_sample_images', 'load_svmlight_file', 'load_svmlight_files', 'load_wine', 'make_biclusters', 'make_blobs', 'make_circles', 'make_classification', 'make_checkerboard', 'make_friedman1', 'make_friedman2', 'make_friedman3', 'make_gaussian_quantiles', 'make_hastie_10_2', 'make_low_rank_matrix', 'make_moons', 'make_multilabel_classification', 'make_regression', 'make_s_curve', 'make_sparse_coded_signal', 'make_sparse_spd_matrix', 'make_sparse_uncorrelated', 'make_spd_matrix', 'make_swiss_roll', 'mldata_filename'.

```
Jupyter MyNotebook Last Checkpoint: an hour ago (unsaved changes)
File Edit View Insert Cell Kernel Help Trusted Python 3
In [2]: print("Hello World!")
Hello World!
In [14]: from sklearn import datasets
print(datasets.__all__)
['clear_data_home', 'dump_svmlight_file', 'fetch_20newsgroups', 'fetch_20newsgroups_vectorized', 'fetch_lfw_pairs', 'fetch_lfw_people', 'fetch_mldata', 'fetch_olivetti_faces', 'fetch_species_distributions', 'fetch_california_housing', 'fetch_covtype', 'fetch_rcv1', 'fetch_kddcup99', 'get_data_home', 'load_boston', 'load_diabetes', 'load_digits', 'load_files', 'load_iris', 'load_breast_cancer', 'load_linnerud', 'load_mldata', 'load_sample_image', 'load_sample_images', 'load_svmlight_file', 'load_svmlight_files', 'load_wine', 'make_biclusters', 'make_blobs', 'make_circles', 'make_classification', 'make_checkerboard', 'make_friedman1', 'make_friedman2', 'make_friedman3', 'make_gaussian_quantiles', 'make_hastie_10_2', 'make_low_rank_matrix', 'make_moons', 'make_multilabel_classification', 'make_regression', 'make_s_curve', 'make_sparse_coded_signal', 'make_sparse_spd_matrix', 'make_sparse_uncorrelated', 'make_spd_matrix', 'make_swiss_roll', 'mldata_filename']
In [ ]:
```

Practical Sessions: Visualisation in Jupyter notebooks



Practical Sessions: Visualisation in Jupyter notebooks



Practical Sessions: Wikidata (Open Data)

The screenshot displays the Wikidata Query interface. At the top, there are navigation links for Examples, Help, and Tools, along with a language selector set to English. The main interface is divided into three sections: a Query Helper on the left, a SPARQL query editor in the center, and a results table at the bottom.

Query Helper: This section allows users to filter and show results. It includes a filter dropdown set to 'instance of' and a text input containing 'programming language'. There is also a 'Show' button and a 'Limit 100' option.

SPARQL Query: The query editor contains the following SPARQL query:

```
1 SELECT ?languageLabel (YEAR(?inception) as ?year)
2 WHERE
3 {
4   #instances of programming language
5   ?Language wdt:P31 wd:Q9143;
6   wdt:P571 ?inception;
7   rdfs:label ?languageLabel.
8   FILTER(lang(?languageLabel) = "en")
9 }
10 ORDER BY ?year
11 LIMIT 100
```

Results: The results table shows 100 results in 123 ms. The first column is 'languageLabel'. The results listed are:

languageLabel
ENIAC coding system
ENIAC Short Code
Von Neumann and Goldstine graphing system

Below the results table, there is a 'Download' menu with options to download the results as a JSON file, JSON file (verbose), TSV file, TSV file (verbose), or CSV file.

Websites

- <https://jupyter.org/>
- <https://www.wikidata.org/>

Colors

- Color Tool - Material Design

Images

- Wikimedia Commons