

MATES ED2MIT
Education and Training for Data Driven Maritime Industry

Data Science and Analytics Foundation

Practice 1 – Preparing your working environment

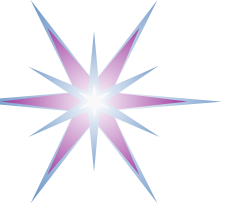
Yuri Demchenko MATES Project
University of Amsterdam

**Maritime Alliance for fostering the
European Blue economy through a
Marine Technology Skilling Strategy**



Co-funded by the
Erasmus+ Programme
of the European Union

The European Commission support for the production of this publication does not constitute an endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



Outline

- Recommended tools for working with Data Analytics tasks
- Installing and configuring Anaconda and Jupyter Notebook
- Configuring your Python working environment
 - Changing location of your working directory
- Installing additional Python libraries
- Setting up RapidMinerEnvironment
- Link to existing tutorials and datasets

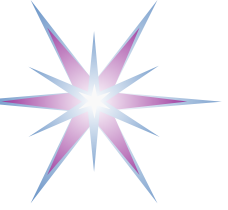


This work is licensed under the Creative Commons Attribution 4.0 International License.
To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>



Co-funded by the
Erasmus+ Programme
of the European Union

The European Commission support for the production of this publication does not constitute an endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



Recommended Learning/working/development Tools

- Working with Jupyter notebooks
 - Anaconda
 - Jupyter Notebook
- Direct working with Python with Command Line Interface (CLI)
 - Python 3.4+
- Visual Studio Code
 - Can work with variety programming languages
 - Can work with python and Jupyter Notebooks
- Rapid Miner Studio



Specialised and Advanced Tools

Pandas Profiling <https://github.com/pandas-profiling/pandas-profiling>

- Generates profile reports from a pandas DataFrame.
 - Extends *pandas df.describe()* function for effective exploratory data analysis with *df.profile_report()* for quick data analysis.
- For each column the following statistics are presented in an interactive HTML report:
 - Type inference: detect the types of columns in a dataframe.
 - Essentials: type, unique values, missing values
 - Quantile statistics like minimum value, Q1, median, Q3, maximum, range, interquartile range
 - Descriptive statistics like mean, mode, standard deviation, sum, median absolute deviation, coefficient of variation, kurtosis, skewness
 - Most frequent values
 - Histogram
 - Correlations highlighting of highly correlated variables, Spearman, Pearson and Kendall matrices
 - Missing values matrix, count, heatmap and dendrogram of missing values
 - Text analysis learn about categories (Uppercase, Space), scripts (Latin, Cyrillic) and blocks (ASCII) of text data.
 - File and Image analysis extract file characteristic and EXIF information.

Good set of examples

- Census Income (US Adult Census data relating income)
- NASA Meteorites (comprehensive set of meteorite landings)
- Titanic (the "Wonderwall" of datasets)
- NZA (open data from the Dutch Healthcare Authority)
- Stata Auto (1978 Automobile data)
- Vektis (Vektis Dutch Healthcare data)
- Colors (a simple colors dataset)
- UCI Bank Dataset (banking marketing dataset)
- RDW (RDW, the Dutch DMV's vehicle registration 10 million rows, 71 features)

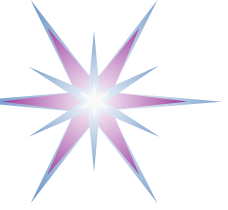


Installing Python and updating main packages

- Required software
 - Install Python 3.6 and newer
 - Install the standard scientific Python stack: Jupyter, NumPy, SciPy, Matplotlib
 - For statistics: pandas, statsmodels, PYMC3
- Checking if packages Available
 - Import numpy – *or if not available* - pip install numpy
 - Import scipy
 - Import pandas
 - Import statsmodels
 - Import pymc3
- Update Python installation module
[python installation directory]python.exe -m pip install --upgrade pip
 - **sudo pip install ansible**

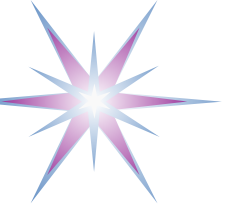
```
Command Prompt - pip install pymc3
Requirement already satisfied: numpy>=1.15 in c:\udevtools\python\python39\lib\site-packages
(from statsmodels) (1.20.2)
Requirement already satisfied: pandas>=0.21 in c:\udevtools\python\python39\lib\site-packages
(from statsmodels) (1.2.4)
Collecting patsy>=0.5
  Downloading patsy-0.5.1-py2.py3-none-any.whl (231 kB)
  | 231 kB 87 kB/s
Requirement already satisfied: scipy>=1.1 in c:\udevtools\python\python39\lib\site-packages
(from statsmodels) (1.6.3)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\udevtools\python\python39\lib\site-packages
(from pandas>=0.21->statsmodels) (2.8.1)
Requirement already satisfied: pytz>=2017.3 in c:\udevtools\python\python39\lib\site-packages
(from pandas>=0.21->statsmodels) (2021.1)
Requirement already satisfied: six in c:\udevtools\python\python39\lib\site-packages (from pa
```

```
Select Command Prompt - pip install pymc3
(from pymc3) (0.5.1)
Requirement already satisfied: pandas>=0.24.0 in c:\udevtools\python\python39\lib\site-packag
es (from pymc3) (1.2.4)
Collecting cachetools>=4.2.1
  Downloading cachetools-4.2.2-py3-none-any.whl (11 kB)
Collecting filelock
  Downloading filelock-3.0.12-py3-none-any.whl (7.6 kB)
Requirement already satisfied: setuptools>=38.4 in c:\udevtools\python\python39\lib\site-pack
ages (from arviz>=0.11.0->pymc3) (49.2.1)
Collecting matplotlib>=3.0
  Downloading matplotlib-3.4.1-cp39-cp39-win_amd64.whl (7.1 MB)
  | 7.1 MB 94 kB/s
Collecting netcdf4
  Downloading netCDF4-1.5.6-cp39-cp39-win_amd64.whl (3.1 MB)
  | 3.1 MB 504 kB/s
Collecting xarray>=0.16.1
  Downloading xarray-0.18.0-py3-none-any.whl (801 kB)
  | 801 kB 467 kB/s
Collecting packaging
  Downloading packaging-20.9-py2.py3-none-any.whl (40 kB)
  | 40 kB 866 kB/s
Collecting kiwisolver>=1.0.1
  Downloading kiwisolver-1.3.1-cp39-cp39-win_amd64.whl (51 kB)
  | 51 kB 60 kB/s
Collecting pyparsing>=2.2.1
  Downloading pyparsing-2.4.7-py2.py3-none-any.whl (67 kB)
  | 67 kB 387 kB/s
Collecting pillow>=6.2.0
  Downloading Pillow-8.2.0-cp39-cp39-win_amd64.whl (2.2 MB)
  | 1.2 MB 242 kB/s eta 0:00:04
```



Downloading and Installing Exercise files and datasets

- Create the directory for all your tutorials or courses
 - For Jupyter Notebook default location is Desktop (you don't to change Jupyter configuration)
 - Default recommended location for exercises and datasets with Download exercise files and datasets and place them in separate directories
 - You can use the same directories structure as in the provided zipped exercise files
 - Optionally, you can create and maintain the separate directory for all datasets



Change location of the working directory in Anaconda

- **How to change Jupyter notebook start up folder in Anaconda**
- [ref] <https://www.planetofbits.com/python/change-jupyter-notebook-startup-folder-anaconda/>
- In Anaconda Navigator open a command prompt window via Environment > base (root) > Open Terminal.
 - Type the command ***jupyter notebook --generate-config*** in the command window and press *Enter*.
 - This will create a file with the name ***jupyter_notebook_config.py*** in the location ***C:\Users\YOUR_USERNAME\jupyter***
- Go to the folder location ***C:\Users\YOUR_USERNAME\jupyter*** and open the file named, ***jupyter_notebook_config.py*** in any text editor.
 - Find the key string, ***#c.NotebookApp.notebook_dir***
 - Uncomment the key string by deleting the ***#*** sign and *in single quotes* type the location of your custom startup folder and save the changes.
- Restart Anaconda and start Jupyter Notebook from the Anaconda Navigator

