Prerequisites

Course materials on GitHub

The course materials are available on GitHub, so that you always have the latest updates. To download them, you first need to install git (if you haven't already).

You can 'clone' the course as follows from the command line (you can also use a GUI)

```
git clone https://github.com/joaquinvanschoren/ML-course.git
```

To download updates, run

```
git pull
```

For more details on using git, see the GitHub 10-minute tutorial. If you want to suggest improvements, you can! Just create a pull request and we'll review it.

Alternatively, you can download the course as a .zip file. Click 'Clone or download'. Download individual files with right-click -> Save Link As...

Python

You first need to set up a Python environment (if you do not have done so already). The easiest way to do this is by installing Anaconda. We will be using Python 3, so be sure to install the right version.

Note: to work with TensorFlow, you need Python 3.5 or 3.6. In fact, Python 3.5 seems to be better supported, especially on Windows.

If you are completely new to Python, we recommend to take an introductory online course, such as the Definite Guide to Python, or the Whirlwind Tour of Python. If you like a step-by-step approach, try the DataCamp Intro to Python for Data Science.

To practice your skills, try the Hackerrank challenges.

Required packages

Next, you'll need to install several packages that we'll be using extensively. You'll need to run these commands on the command line.

NOTE: Windows users: Follow the guidelines under the section 'Installing packages for windows'.

Installing packages with conda

If you are using Anaconda, you can use the conda package manager to install all packages:

```
conda install numpy scipy scikit-learn matplotlib pandas pillow graphviz python-
```

To install *TensorFlow*, follow these instructions for your OS (Windows, Mac, Ubuntu). While installation with conda is possible, they recommend to install with pip, even with an Anaconda setup.

To install *Keras*, follow the instructions here. If anything goes wrong, see the separate file with installation instructions in the repository.

Installing packages with pip

With most other setups (not conda), you can use pip to install all packages. Pip is the Python Package index. It is included in most Python installations.

```
pip install numpy scipy scikit-learn matplotlib pandas pillow graphviz scikit-im
```

Note: we'll be using scikit-learn 0.19, which is currently the latest version.
You also need to install the graphiz C-library:

- OS X: use homebrew: `brew install graphviz`
- Ubuntu/debian: use apt-get: `apt-get install graphviz`.
- Installing graphviz on Windows can be tricky and using conda / anaconda is recommended.

Note: if you are not using Anaconda, and you already have a custom Python environment set up, possibly using a different Python version, it may be wise to set up a virtual environment for this course so that it does not affect your existing environment.

Installing OpenML

OpenML is used to easily import datasets and share models and experiments. You can install OpenML the normal way:

```
pip install openml
```

However, it is recommended to install the latest version via GitHub instead, including the ARFF reader. With git installed, run the following:

```
pip install git+https://github.com/renatopp/liac-arff@master
pip install git+https://github.com/openml/openml-python.git@develop
```

You'll also need an OpenML account to download/upload data. If you don't have one, go ahead and create one.

Installing Jupyter notebooks

As our coding environment, we'll be using Jupyter notebooks. They interleave documentation (in markdown) with executable Python code, and they run in your browser. That means that you can easily edit and re-run all the code in this course.
If you use Anaconda, Jupyter is already installed. If you use pip, you can install it with

```
pip3 install jupyter
```

To test if it works, run

```
jupyter notebook
```

A browser window should open showing the files in your current directory. You can shut down the notebook by typing CTRL-C in your terminal.
If you are new to notebooks, take this quick tutorial, or this more detailed one. Optionally, for a more in-depth coverage, try the DataCamp tutorial.

Testing

To test if everything works, download the course materials and run Jupyter (ideally from the directory where you downloaded the course):

```
jupyter notebook
```

A browser window should open with all course materials. Open one of the chapters and check if you can execute all code by clicking Cell > Run all.

```
[ ]:
```