Prerequisites

Course materials on GitHub

The course materials are available on Canvas, but the sources are also on GitHub, so that you always have the latest updates. To download them, you first need to install git (https://git-scm.com/book/en/v1/Getting-Started-Installing-Git) (if you haven't already).

You can 'clone' the course as follows from the command line (you can also use a GUI (https://desktop.github.com/))

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

To download updates, run `git pull`

For more details on using git, see the GitHub 10-minute tutorial (https://guides.github.com/activities/hello-world/) and Git for Ages 4 and up (https://www.youtube.com/watch?v=1ffBJ4sVUb4).

Alternatively, you can download the course as a .zip file (https://github.com/ML-course/engineer.git). 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 (https://www.continuum.io/downloads). We will be using Python 3, so be sure to install the right version.

If you are completely new to Python, we recommend reading the Python Data Science Handbook (https://github.com/jakevdp/PythonDataScienceHandbook) or taking an introductory online course, such as the Definite Guide to Python (https://www.programiz.com/python-programming), the Whirlwind Tour of Python (https://github.com/jakevdp/WhirlwindTourOfPython), or this Python Course (https://www.python-course.eu/). If you like a step-by-step approach, try the DataCamp Intro to Python for Data Science (https://www.datacamp.com/courses/intro-to-python-for-data-science).

To practice your skills, try the Hackerrank challenges (https://www.hackerrank.com/domains/python).

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.

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-graphviz scikit-image joblib six
```

Installing TensorFlow

To install *TensorFlow 2*, follow these instructions (https://www.tensorflow.org/install/) for your OS (Windows, Mac, Ubuntu). While installation with `conda` is possible, they recommend to install with `pip`, even with an Anaconda setup.

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 j
oblib six
```

Note: we'll be using scikit-learn 0.22, 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.

Virtual environments

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 (http://docs.python-guide.org/en/latest/dev/virtualenvs/) for this course so that it does not affect your existing environment.

Installing Jupyter

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 jupyterlab
```

In both cases, to play around with the interactive animations in this course, also do:

```
jupyter labextension install @jupyter-widgets/jupyterlab-manager
```

To test if it works, run `jupyter lab`

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 (https://try.jupyter.org/), or this more detailed one (http://nbviewer.jupyter.org/github/jupyter/notebook/tree/master/docs/source/exa
Optionally, for a more in-depth coverage, try the DataCamp tutorial (https://www.datacamp.com/community/tutorials/tutorial-jupyter-notebook#gs.wlHChdo).

Running the course notebooks

Run jupyter lab from the folder where you have downloaded (cloned) the course materials.

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.

Installing OpenML

OpenML is used to easily import datasets and share models and experiments.

```
pip install openml
```

For Windows, you need to have a C++ Compiler installed. If the above install fails, you may need to install this first. Download and install Visual C Build Tools (http://landinghub.visualstudio.com/visual-cpp-build-tools)

You'll also need an OpenML account to upload data. If you don't have one, go ahead and create one (http://www.openml.org).

Alternative environments for running the notebooks

Google Colab

Google Colab allows you to run a notebook on Google Drive (with limited GPU support): https://colab.research.google.com/notebooks/gpu.ipynb (https://colab.research.google.com/notebooks/gpu.ipynb) A more detailed tutorial can be found here (you won't need PyTorch for this course, but we do recommend learning it): https://towardsdatascience.com/fast-ai-lesson-1-on-google-colab-free-gpu-d2af89f53604 (https://towardsdatascience.com/fast-ai-lesson-1-on-google-colab-free-gpu-d2af89f53604)

There are limitations (obviously): right now GPU usage is limited to 12h and RAM is shared among multiple users.

Note: You need to upload your course notebooks to colab yourself (File > Upload Notebook). You can install additional packages from within notebooks with '!pip install package'.