Installation Guide

5SSD0 Bayesian Machine Learning & Information Processing

Wouter M. Kouw, TU Eindhoven November 12, 2021

This guide will help you install everything you need to run the course notebooks on your own machine (Windows, Linux or MacOS). We assume basic familiarity with the use of shells and prompts on your computer, for instance changing directories with `cd`. If you can't get things to work, check the notes at the end of this document. If that doesn't solve it, ask us in class, on Piazza or raise an issue on Github (`https://github.com/bertdv/bmlip/issues`).

Contents

1 Windows

If things don't work out, consider working from the Windows Subsystem Linux (WSL). For an installation tutorial, see: `https://ubuntu.com/tutorials/tutorial-ubuntu-on-windows`.

1.1 Julia

Go to `https://julialang.org/downloads/` and download the Windows executable for the **stable release v1.6.3**. Finish the installation process with default settings. You can check whether this step succeeded by calling a Julia shell from Start (see Fig. 1).

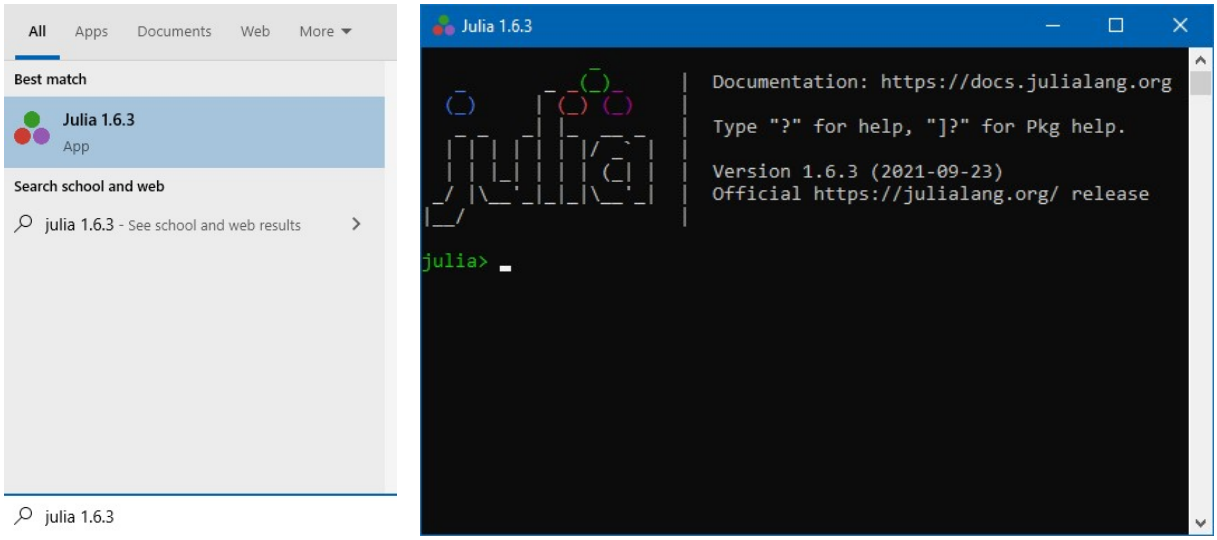

Figure 1: (Left) Call 'julia 1.6.3' from Start. (Right) Julia shell on Windows.

1.2 Jupyter

To install Jupyter, we need a Python environment. If you already have a Python environment, you can skip the next step.

1.2.1 Case: You do not have Python installed

The easiest way to set this up is by installing a Miniconda environment. Go to `https://docs.conda.io/en/latest/miniconda.html#windows-installers` and download the Windows executable for **Python 3.9** (the link called `Miniconda3 Windows 64-bit`). Run the installer with default settings. Now, we can install Jupyter via Conda. Open a Miniconda shell (see Fig. 2) and run the following command:

```
conda install -c anaconda jupyter
```

You'll need to install Matplotlib as well. Run:

```
conda install -c conda-forge matplotlib
```

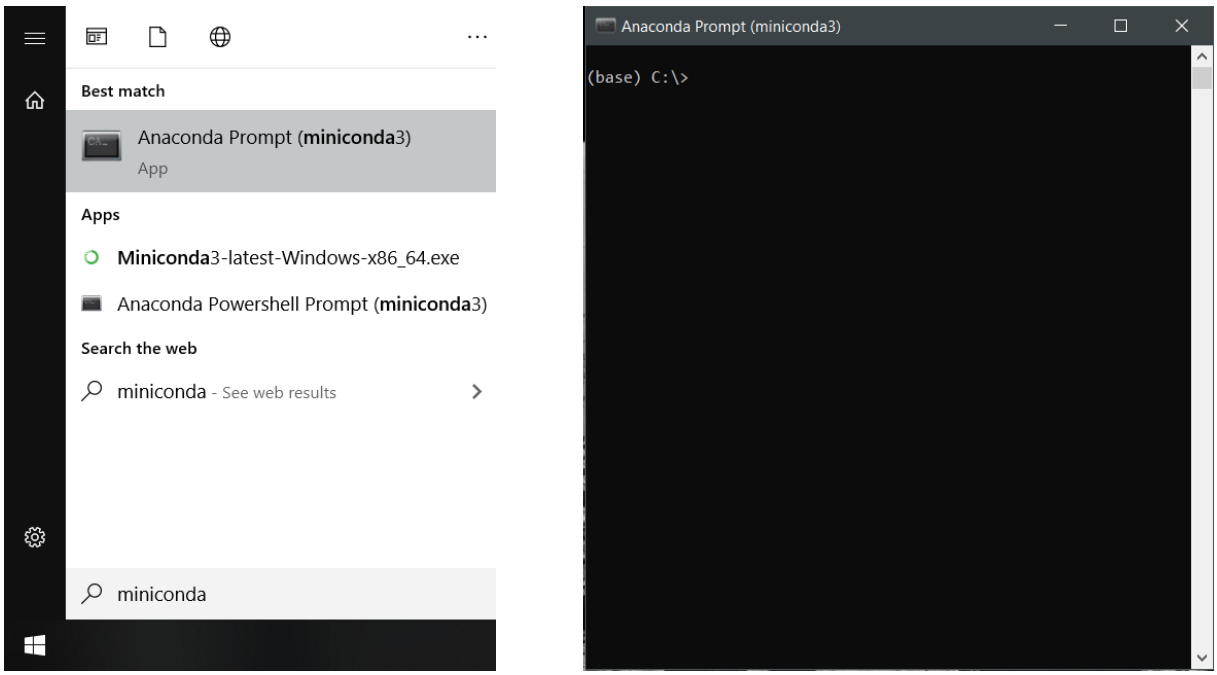

Figure 2: Anaconda prompt on Windows.

To check if the installation was successful, run `conda list` from the Miniconda shell and look for 'jupyter' and 'matplotlib'.

1.2.2 Case: You have Python installed

If you have Python installed (note that you need **Python 3**), but not Conda, then you can install Jupyter and Matplotlib via:

```
pip install jupyter matplotlib
```

If you're having problems with installing via Pip, download Miniconda. It can be removed without affecting your existing Python environment.

1.3 IJulia

IJulia is a Julia package that lets you add a Julia kernel to your Jupyter setup easily. Open Start, type `julia 1.6.3` and run the shell. Enter the following lines of code sequentially:

```
using Pkg
Pkg.add("IJulia")
```

If you now create a new Jupyter notebook, you should have the option to use a Julia kernel (see Figure 3). Within the notebook, the kernel version is listed in the top-right corner.

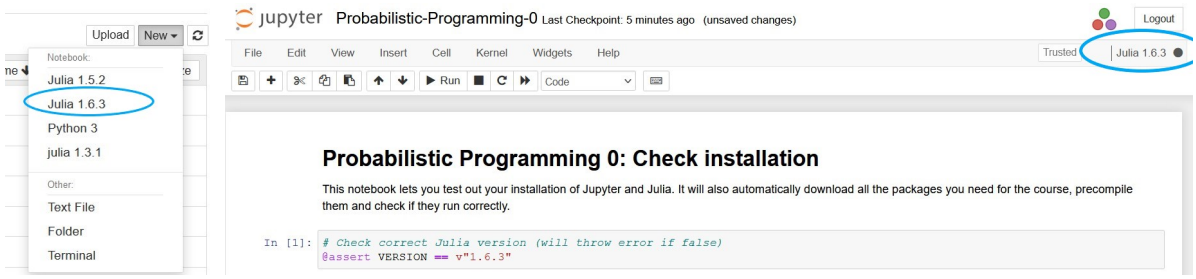

Figure 3: (Left) Kernel specification when creating new notebooks. (Right) Kernel version description within notebook.

1.4 GraphViz

Go the website of GraphViz and download the latest stable version: "graphviz-2.49.3 (64-bit) EXE installer". Step through the installation process until you reach the options for the system PATH. Tick 'Add GraphViz to the system PATH for current user' (see Figure 4).

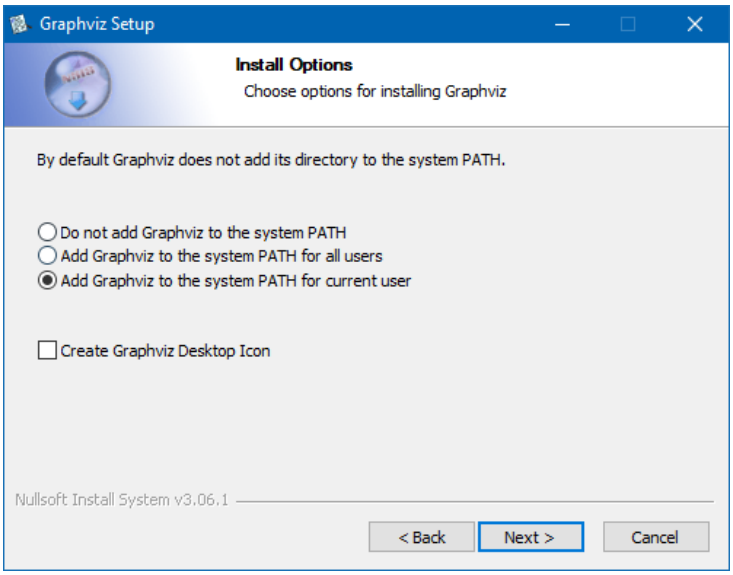

Figure 4: Option to add GraphViz to PATH.

Once the installer is done, open an Anaconda shell and run:

```
dot -V
```

If that gives you a version number, then you're good to go. If it gives you an error stating that it does not know the command `dot`, then something went wrong with adding the binary to PATH. To fix that, type `environment variables` into Start and open the Environment Variables window (see Fig. 5). The top part shows settings for the current user. One of the variables is called "Path", double-click that one. That will give you a list of folders. Click "New" and add the folder where you installed graphviz:

```
C:/<path/to/graphviz>/Graphviz2.49/bin
```

Try running `dot -V` again. If you're still getting the unknown command error, ask for help in class or on Piazza.

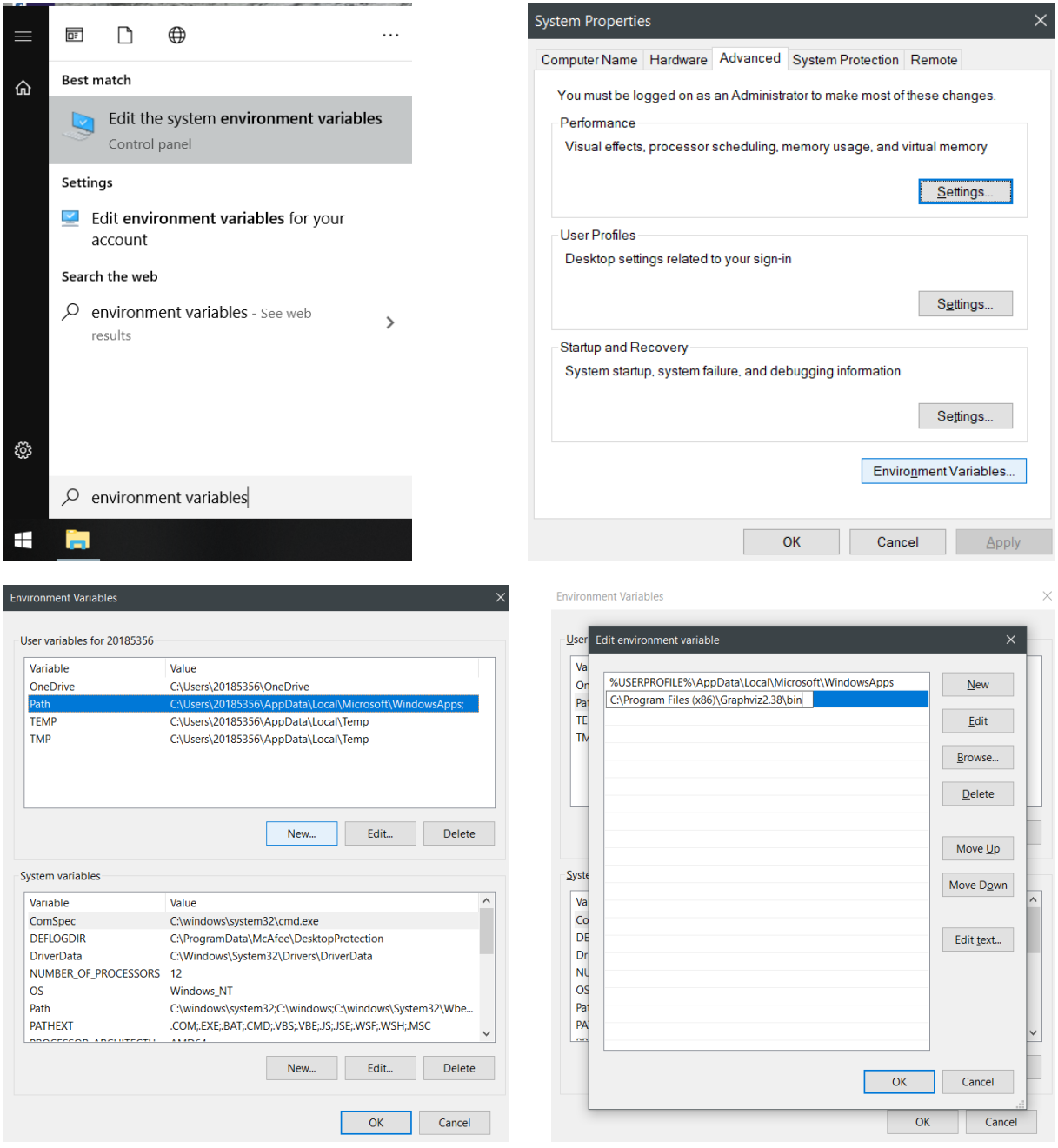

Figure 5: Adding Graphviz to your PATH variable. 1) Type 'environment variables' in Start and click the link to the Control Panel. 2) Click the 'Environment Variables' button. 3) Double-click 'Path' in the list of user variables. 4) Add '<path>\Graphviz2.49\bin' as a 'New' entry.

2 Linux

Linux is the easiest OS to use, since most of the software was designed on Linux machines.

2.1 Julia

We are going to follow this help page: `https://julialang.org/downloads/platform/#linux_and_freebsd`. Start by downloading and extracting the Julia **stable release v1.6.3** tarball ('Generic Linux for x86') from `https://julialang.org/downloads/`:

```
cd ~/Downloads/
wget https://julialang-s3.julialang.org/bin/linux/x64/1.6/julia-1.6.3-linux-x86_64.tar.gz
tar zxvf julia-1.6.3-linux-x86_64.tar.gz
sudo mv julia-1.6.3 /opt/
```

The created directory `/opt/julia-1.6.3/` is your "Julia installation directory". To finish the installation, create a symbolic link from the Julia installation directory to your binaries directory:

```
sudo ln -s /opt/julia-1.6.3/bin/julia /usr/local/bin/julia
```

You should now be able to open a terminal and call `julia` (see Figure 6).

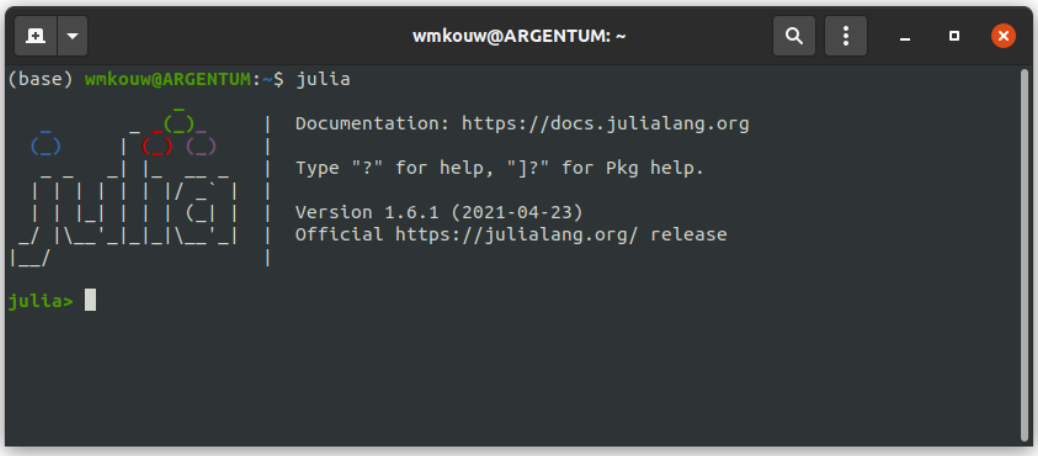

Figure 6: Julia within Terminal.

2.2 Jupyter

To install Jupyter, we need a Python environment. If you already have Python or Conda installed, you can skip the next step.

2.2.1 Case: You do not have Python installed

The easiest way to set up a proper Python environment is by installing Miniconda. Go to `https://docs.conda.io/en/latest/miniconda.html#linux-installers` and download the

package for **Python 3.9** (the link for `Miniconda3 Linux 64-bit`). Run the installer by opening a Terminal, navigating to the download directory and calling:

```
bash Miniconda3-latest-Linux-x86_64.sh
```

Accept all default settings, unless you're familiar with the questions asked. Now that we have a Miniconda environment, we will install Jupyter as well as Matplotlib via Conda. Open a terminal and run the following two commands:

```
conda install -c anaconda jupyter
conda install -c conda-forge matplotlib
```

Check for successful install by running `conda list` and checking whether Jupyter and Matplotlib are listed.

2.2.2 Case: You have Python installed

If you have Python installed, but not Conda, then you can install Jupyter and Matplotlib via:

```
pip install jupyter matplotlib
```

If you're having problems with installing via Pip, download Miniconda and use `conda`. The Miniconda environment can be removed without affecting your existing Python environments.

2.3 IJulia

This one's easy. Open a Terminal, call `julia` and:

```
using Pkg
Pkg.add("IJulia")
```

You should now be able to run a Julia kernel in your Jupyter notebook (specifically, version **1.6.3**; see Figure 3).

2.4 GraphViz

This is a one-liner:

```
sudo apt-get install graphviz
```

Check whether it installed correctly by running `dot -V`.

3 MacOS

3.1 Julia

Go to `https://julialang.org/downloads/` and download the MacOS `.dmg` installer for the **stable release 1.6.3**. The Julia shell is now installed as a stand-alone application. If you wish to add it to your Terminal, see `https://julialang.org/downloads/platform/#macos`.

3.2 Jupyter

If you already have Python or Conda installed, you can skip the next step.

3.2.1 Case: You do not have Python installed

The easiest way to set up a proper Python environment is by installing Miniconda. Go to `https://docs.conda.io/en/latest/miniconda.html#macosx-installers` and download the package for **Python 3.9** (the link called `Miniconda3 MacOSX 64-bit bash`). Run the installer by opening a terminal, navigating to the download directory and calling:

```
bash Miniconda3-latest-MacOSX-x86_64.sh
```

Accept all default settings, unless you're familiar with the questions asked. Now that we have a Miniconda environment, we will install Jupyter as well as Matplotlib via Conda. Open a terminal and run the following two commands:

```
conda install -c anaconda jupyter
conda install -c conda-forge matplotlib
```

Check for successful install by running `conda list` and seeing if jupyter and matplotlib are listed.

3.2.2 Case: You have Python installed

If you have Python installed, but not Conda, then you can install Jupyter and Matplotlib via:

```
pip install jupyter matplotlib
```

If you're having problems with installing via Pip, download Miniconda and use `conda`. The Miniconda environment can be removed without affecting your existing Python environments.

3.3 IJulia

Open the Julia application and enter:

```
using Pkg
Pkg.add("IJulia")
```

You should now be able to run a Julia kernel in your Jupyter notebook (specifically, version **1.6.3**; see Figure 3).

3.4 GraphViz

We are following the guide on Graphviz's website `https://www.graphviz.org/download/#mac`. You need to have either HomeBrew or MacPorts as a package manager. On HomeBrew, the command is:

```
brew install graphviz
```

and via MacPorts, the command is:

```
sudo port install graphviz
```

In both cases, check whether it installed correctly by running `dot -V`.

4 Opening notebooks

If you've completed all of the installation steps above, then you should be able to open and run the course notebooks. To download the notebooks, go to our Github repository, `https://github.com/bertdv/BMLIP`, push the green 'Code' button and then 'Download ZIP' (see Figure 7). Unzip the package, open a terminal (on Windows, it doesn't matter whether you

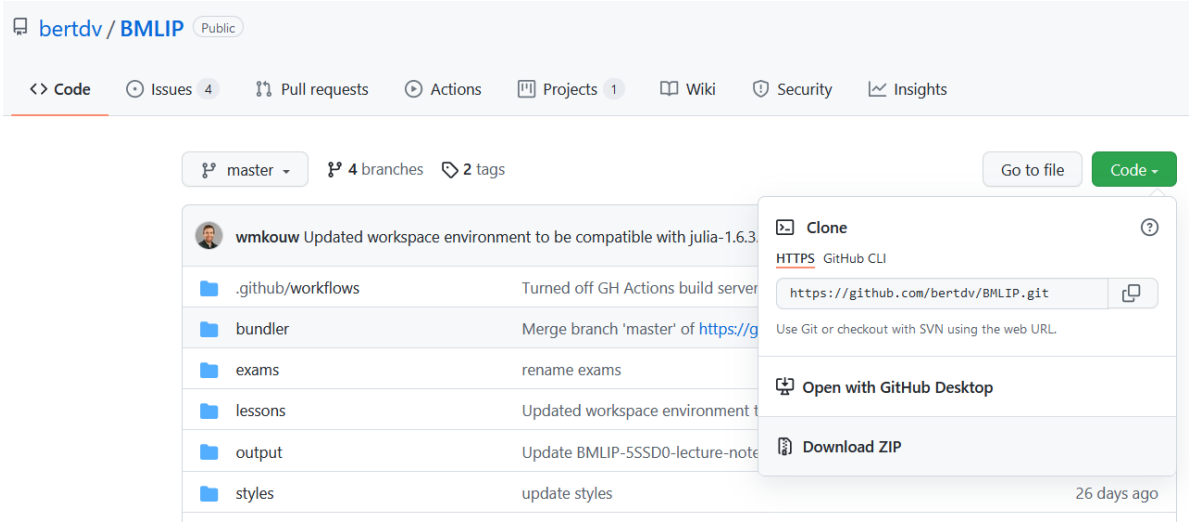

Figure 7: Download course materials from Github.

use an Anaconda prompt or the Command Prompt) and navigate to the course directory (i.e. 'path/where/you/unzipped/package/BMLIP'). Start Jupyter by running

```
jupyter notebook
```

in the terminal/prompt. The localhost link will be opened in your default browser automatically (it should look something like Fig. 8). For more info on Windows, check `https://tinyurl.com/ul6a9hy`. Click 'lessons' and then 'notebooks'. The files ending in `.ipynb` are Bert's lectures. The Probabilistic Programming notebooks are under the directory `probprog`.

Figure 8: Starting a Jupyter notebook in the course repository.

4.1 Testing for successful installation

The notebook called `Probabilistic-Programming-0.ipynb` is there for you to test your installation. To check whether you're all set to go, click 'Cell' and 'Run all'. If you don't get any errors, then you're done.

5 Possible solutions

This is a list of things to try if you run into any unexpected problems.

- If you're getting HTTP request / timeout errors during installation, your internet connection might be unstable and your downloaded could have been corrupted. You should verify the integrity of the downloaded file. For Linux and MacOS, this is done by opening a terminal and running

```
sha256sum <download>
```

You should check whether the number it spits out matches the number listed on the website of the download. On Windows, the command is:

```
CertUtil -hashfile C:\<path\to\download>.msi SHA256
```

via the Command Prompt.