

# Predictive Analytics with Python

## Getting Started

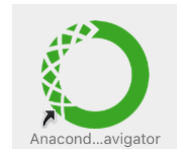
Python is a powerful, general purpose programming language that can be used for many applications ranging from short scripts to enterprise applications. There is a large and growing number of free, open-source libraries and tools for scientific computing. For more information about Python and its use visit [python.org](https://python.org).

### Install Python

There are many ways of using and developing with Python. However, for this course, we will be using Jupyter notebooks, an interactive, browser-based Python interface available through the [Anaconda Distribution](https://anaconda.com/distribution) which is particularly useful for scientific computing. We will be using Python 3.x in this course. While Python 2.x is still available, it is no longer actively developed and many library providers will stop supporting it or

Here is what you need to do:

- Download the Anaconda installer for Python 3.6 or later from <https://www.anaconda.com/download/> for your operating system (you will be asked for your email, however this step is optional and you can proceed without providing it)
- Execute the installer
  - macOS: double-click on the *pkg* file and follow the instructions using the default settings
  - Windows: run the *exe* file and follow the instructions using default settings
  - Anaconda now includes Microsoft Visual Studio Code and you will be asked if you want to install it. This code editor is not required for the course
- Once the application is installed, you can execute *Anaconda Navigator* from the Start Menu (Windows) and the Application folder (macOS)

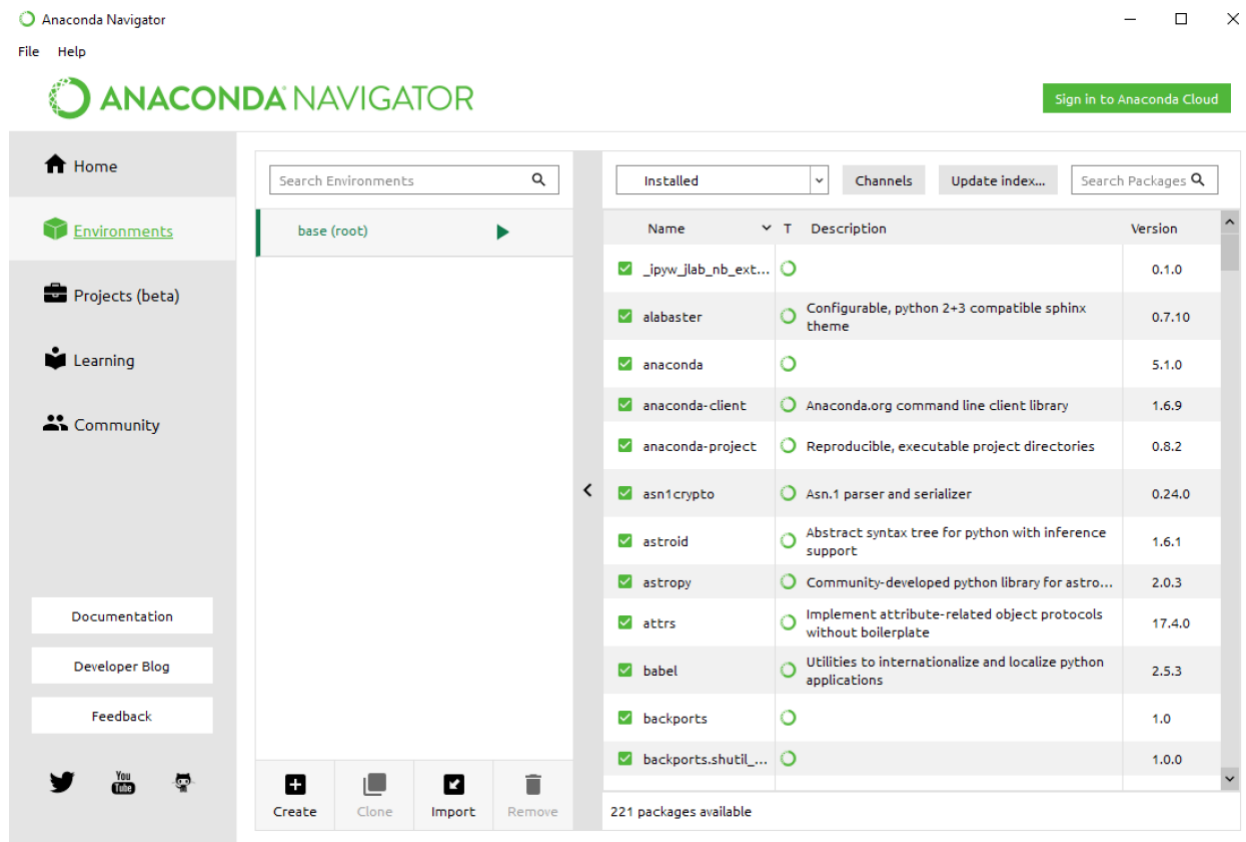


### Anaconda Navigator – update and install packages

You can use *Anaconda Navigator* to manage your Python installation and run the Jupyter application.

Use the *Environments* tab to add packages to your Python installation.

## Predictive Analytics with Python - Getting Started

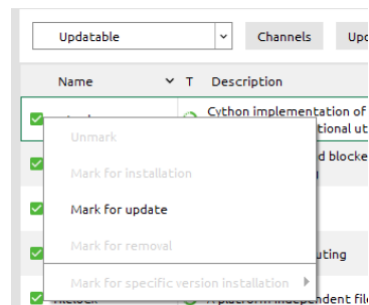








Click the [Update index...] button to refresh the package list. From time to time, it may ask you to update the Anaconda Navigator application. It's good practice to update regularly.

If new versions become available, you will see that the version number changes. The version number of updatable packages are highlighted in blue and with a

✓	asn1crypto	Asn.1 parser and serializer	0.24.0
✓	astroid	Abstract syntax tree for python with inference support	1.6.1
✓	astropy	Community-developed python library for astronomy	2.0.3
✓	attrs	Implement attribute-related object protocols...	17.4.0

This means that you can update the specific package. Change the pull-down menu to [Updatable] and click the green tick mark to select [Mark for update]. Do that for all the packages you want to update, select [Apply] and confirm the update.



 numpy	 Array processing for numbers, strings, records, and objects	 1.13.3
 numpydoc	 Sphinx extension to support docstrings in numpy format	 0.7.0

Once you initiated the update, use the [Clear] button to remove the marking. *Anaconda Navigator* otherwise will indicate that it is busy when you want to close the application.

Updates are done in the background and will take some time and may require confirmation. There is no feedback that an update is finished. You will need to refresh the list using [Update index...] to see the progress.

You will not need to update all packages, however update at least the following packages required for the course:

- **Python**: the Python interpreter
- **Matplotlib**: Python 2D plotting library (<https://matplotlib.org/>)
- **networkx**: Python package for creating and manipulating complex networks (<https://networkx.github.io/>)
- **NumPy**: fundamental package for scientific computing with Python (<https://www.numpy.org/>)
- **Pandas**: high-performance, easy-to-use data structures and data analysis tools (<https://pandas.pydata.org/>)
- **scikit-learn**: machine learning in Python (<http://scikit-learn.org/>)
- **seaborn**: statistical data visualization (<https://seaborn.pydata.org/>)
- **statsmodels**: implementation of different statistical models and tests (<https://www.statsmodels.org/>)

Install the following:

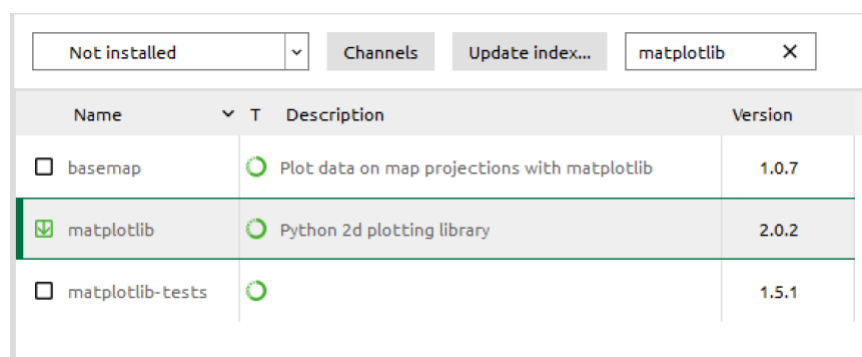
- **cartopy**: a library providing cartographic tools for Python (<http://scitools.org.uk/cartopy/>). Only required if you want to run all examples from the book
- **graphviz**: Application to visualize graphs (<https://www.graphviz.org/>)<sup>1</sup>
- **python-graphviz**: Python interface for graphviz (<https://graphviz.readthedocs.io/en/stable/>)
- **pydotplus**: Python interface to graphviz's dot language. Required to visualize decision trees (<http://pydotplus.readthedocs.io/>)

---

<sup>1</sup> On Windows, you will need to include the graphviz executable in your path variable, e.g. C:\Anaconda3\Library\bin\graphviz

- **gmaps**: Python interface to Google maps. See appendix for details about installing this package (<https://github.com/pbugnion/gmaps>)
- **nltk**: Natural language processing toolkit. Required for more advanced text mining applications (<https://www.nltk.org/>)
- **mlxtend**: machine learning library that provides access to association rules mining algorithms (<https://github.com/rasbt/mlxtend>)
- **scikit-surprise**: a library for recommender systems (<http://surpriselib.com/>)
- **squarify**: algorithm to layout tree map visualizations (<https://github.com/laserson/squarify>)

To install a package, change the pull down to [Not installed] and enter e.g. matplotlib in the [Search packages] field. Click on the rectangle to select the package for download and use the [Apply] button to start the installation.



Once the library is installed, it will be listed under the installed packages.

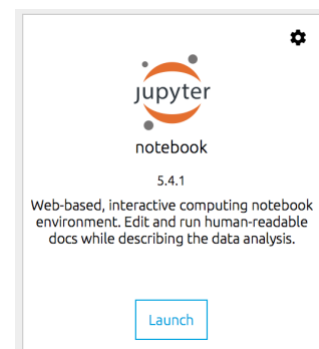
You can also install a library from the command line, which may be faster, by using the command

```
conda install packagename
```

This may be faster than using the interface.

## Anaconda Navigator – Launch Python in a Jupyter Notebook

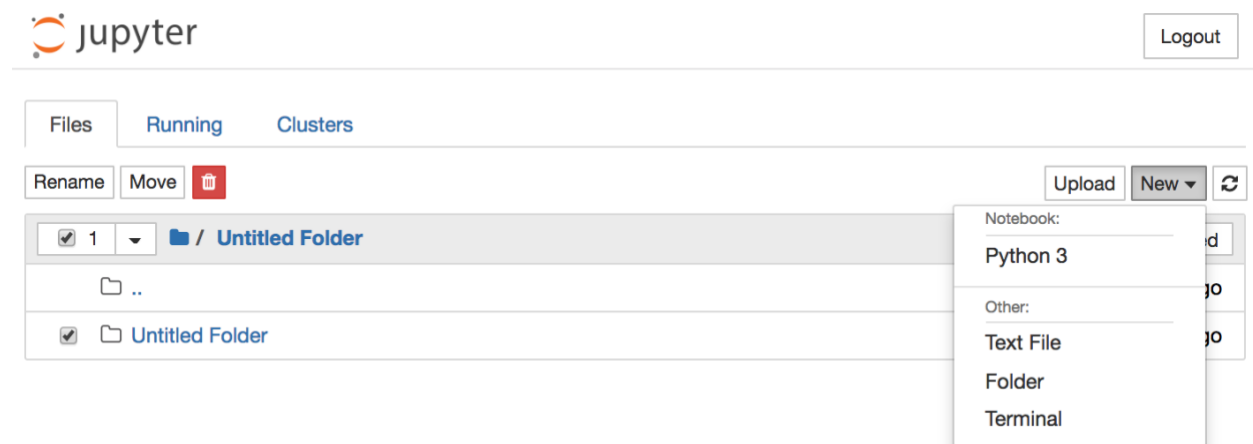
We recommend that you use Jupyter notebooks for the exercises. The Jupyter notebook is a web-based computing environment that runs on your computer and embeds Python code and output together with comments and graphics in one readable document. In the last years, this has become a popular way for interactive data analysis in the data science community.



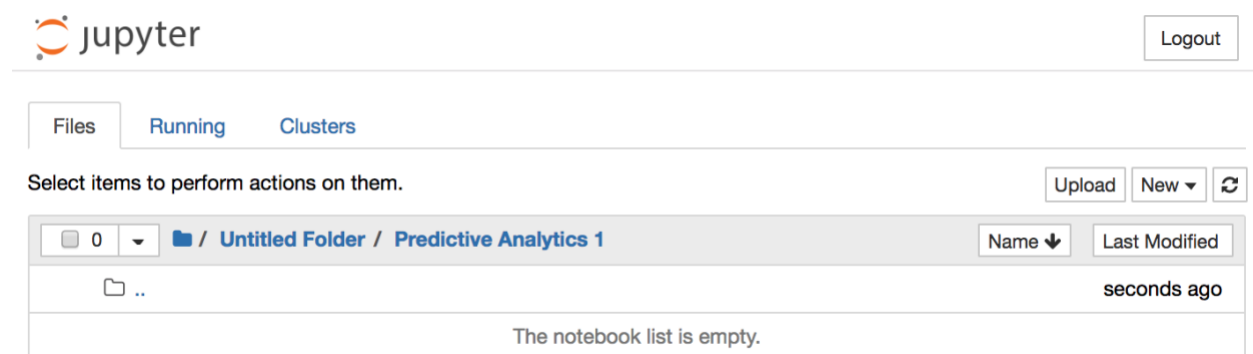
## Predictive Analytics with Python - Getting Started

Select the *Home* tab in the Anaconda Navigator and *Launch* Jupyter notebook. The notebook is launched inside your usual web browser. The supported browsers are *Chrome*, *Safari*, or *Firefox*. An up-to-date version of *Edge* may also work; if not, use one of the supported browsers.

The Jupyter notebook application opens a file manager page which allows you to browse to your working directory. You can also create new folders [*New/Folder*] and text files [*New/Text File*] here.



To rename a file or folder select it and use [Rename] to change the name.

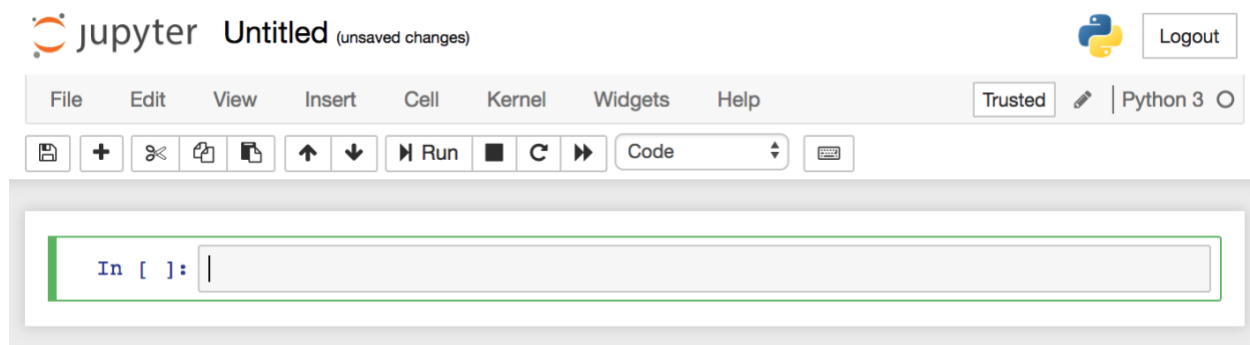


Create a folder to keep your work for the course and navigate into the folder. Next use [*New/Python 3*] to create a new notebook which opens in a separate tab or window.

## Jupyter notebook

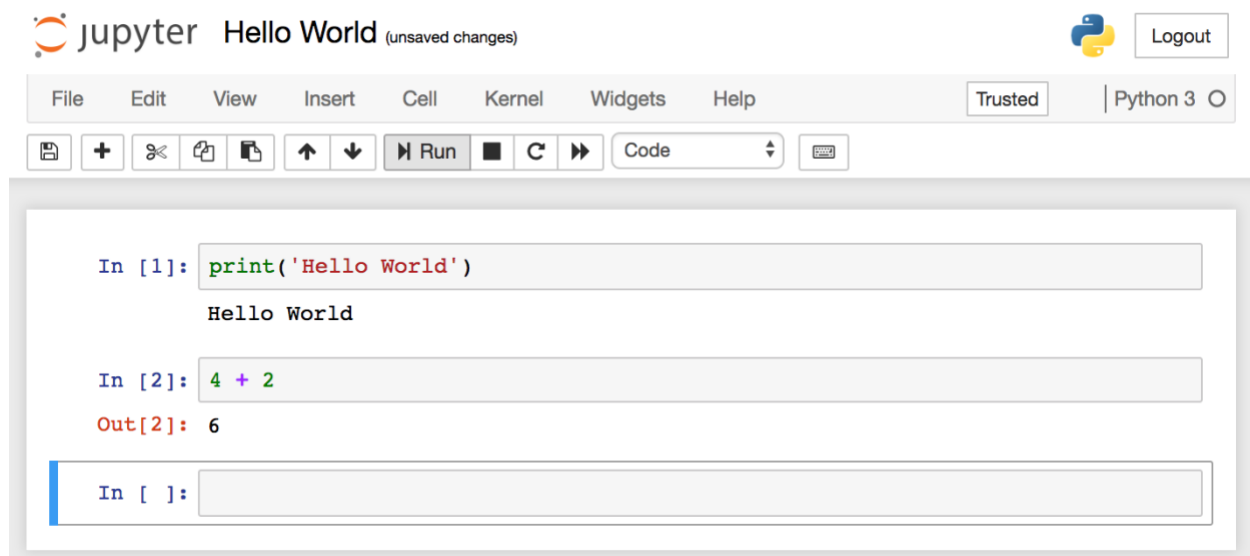
This is what an empty notebook looks like.

## Predictive Analytics with Python - Getting Started



Click on *Untitled* and replace it with as more meaningful title.

You can enter Python code in the code boxes and execute it using the *[Run]* button.



The output and result of the last statement in each code box is printed underneath each block.

Jupyter notebooks regularly saves your work automatically. If you want to trigger the save manually, use the *[Save]* button, the *[File|Save and Checkpoint]* menu or the *[Ctrl/Cmd-S]* key.

If you find an error in your code, you can modify it and rerun the code. From time to time, you may want to rerun the whole code in your notebook; use the menu *[Kernel|Restart & Run All]* for this.

# Appendix

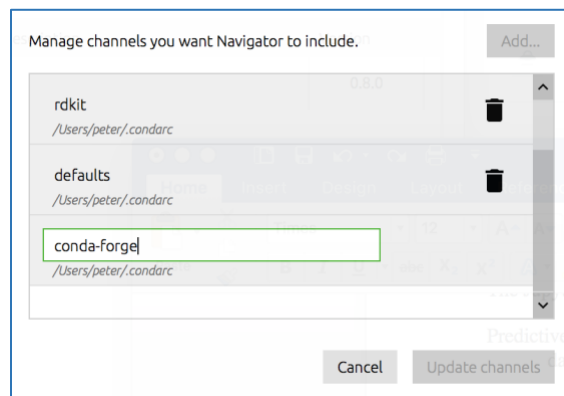
### Installing dmba

The package *dmba* (<https://pypi.org/project/dmba/>) provides a number of utility functions that are used throughout the book. It is available through PyPI, the Python package index, and can be installed using the command

```
pip install dmba
```

on the command line.

### Installing gmaps



The Python package is available from the *conda-forge* channel. In the *Environments* tab of Anaconda Navigator, click the [Channels] button and add the conda-forge channel. Close the dialog using [Update channels].

After [Update index...] the *gmaps* package is available for installation.