# Installing Python 3.6 via Miniconda

We will be using Python 3.6 in this class. The easiest way to install this is through the Anaconda distribution, which is platform independent so avoids many potential tricky IT issues. However, the Anaconda installer takes much space and a while to download, since it installs 150 packages at once. I would recommend instead using the **Miniconda** installer, which installs the basic Python environment, and you can later install the specific packages you need. (However, if you have a lot of disk space and don't mind the large download, you can skip the rest of this section and following the full Anaconda installation instructions here.)

To install using Miniconda (skip this if you already installed the full Anaconda by following instructions above), download the installer for your Operating System (OS) here: https://conda.io/miniconda.html.

Then run the installer and keep the default options. (Choose the default location for copying files and to add this location to your "PATH" environment variable.)

- For windows, simply double click the .exe file downloaded, and choose the two options above.
- For Mac or Linux, you would open a Terminal and type in the following, replacing the capitalized terms with the correct location. Then, go through the user agreement and type yes for everything.

```
cd PATH_WHERE_YOU_DOWNLOADED_THE_FILE_STARTING_WITH_SLASH
bash NAME_OF_FILE_YOU_DOWNLOADED
```

For example, depending on where you downloaded the installer, this might be

```
cd /home/user/pengshi/Downloads/
bash Miniconda3-latest-MacOSX-x86_64.sh
```

After this, launch the **Anaconda Prompt** if you are using Windows and type the following command to install the packages we will frequently use. If you are using Mac or Linux, you would have to close the terminal where you installed and start a new one, then type in the following commands into the terminal.

```
conda install numpy scipy matplotlib pandas jupyter ipython spyder
```

In the above command, the first two words are standard syntax. Starting from `numpy`, we are listing a set of Python packages to install. Each package is separated by a space. This is the Syntax for installing future Python packages. The first four packages, numpy, scipy, matplotlib and pandas and bread and butter for manipulating data sets. Jupyter is what is used to create notebooks similar to the one you are reading now. IPython is an interactive shell that is better than the original "Python" shell. Spyder is a Interactive Development Environment (IDE), which is an editor for code, that I recommend (similar to RStudio for R), but there are many IDEs that you can use instead. Here is a webpage suggesting the pros and cons of popular IDEs for Python.

To start running Python, type one of `ipython`, `spyder`, or `jupyter notebook` into the Anaconda prompt (on Windows) or any terminal (on Linux or Mac). The first activates the IPython shell. You can begin typing Python commands and seeing the output. The second is the graphical interface. The third launches a web browser. And you can start a new Python 3 shell and the interface looks like what you see now. You can test your installation by running the code.

```
[2]: print ("My first Python code")
```

```
My first Python code
```

## Installing Graphviz (Optional)

This is a software package you need in order to visualize decision trees using the code provided in the course notes.

In the Anaconda prompt in Windows and in any termina in Mac or Linux, type the following set of commands:

```
conda install -c conda-forge graphviz python-graphviz
```

To test whether this works, go into IPython or Jupyter notebook and type the following, and see if there's an error message.

```
from graphviz import Graph
g=Graph('test')
g.node('a')
g.node('b')
g.edge('a','b')
g.view()
```

## Installing Gurobi

Later in the course, we will need to install Gurobi, which is the most powerful software in the world currently for solving large-scale linear optimization and mixed-integer optimization problems. (We will learn about these topics in the second half of the course.)

To do this, follow the instructions here: http://www.gurobi.com/downloads/get-anaconda. This webpage tells you to type into the Anaconda prompt,

```
conda config --add channels http://conda.anaconda.org/gurobi
```

Followed by

```
conda install gurobi
```

If the first line above results in a Permission denied error, an alternative way (replacing the above two lines is to execute the following single command

```
conda install -c http://conda.anaconda.org/gurobi gurobi
```

Afterward, close your Anaconda prompt or terminal and start a new one and you should be good to go.

To use Gurobi, you need to obtain a **free academic license** for your computer. To do so, following steps 2-4 of the instructions on the following webpage, under the heading "*To obtain a free named-user academic license*": http://www.gurobi.com/academia/for-universities.

Note that after registering on the Gurobi website using your USC.edu email address, you also need to run the command grbgetkey followed by the license key, as indicated in step 4 of the above instructions.

To test if you successfully installed, enter IPython and type the following and see if there is an error.

```
import gurobi
```