Instructions for Creating an Analysis Environment with Anaconda

Liam O'Suilleabhain

February 8, 2019

Development Directory

Here, we are going to define the root directory of our analysis environment.
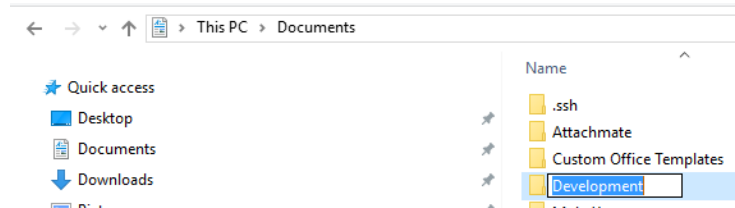

Figure 1: The Root Directory of our Analysis Environment is set to Development

Installation of Miniconda

This section describes steps for installing the Anaconda package manager (Miniconda) for 64-bit Windows 10 laptops and Desktops.

Download Miniconda for 64-bit Windows

Open the Executable

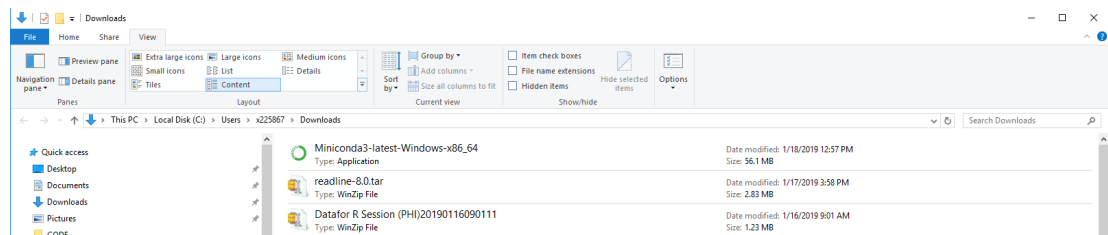Figure 2: Click on the python3.7 64-bit Windows (exe-installer).

Figure 3: Double click on the file downloaded
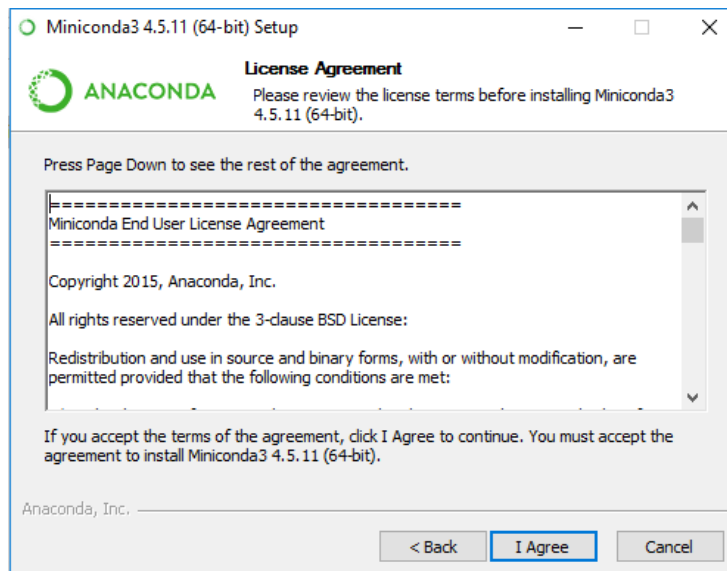
Installation Process

Figure 4: Baby Steps

Figure 5: BSD licenses are a family of permissive free software licenses, imposing minimal restrictions on the use and redistribution of covered software.
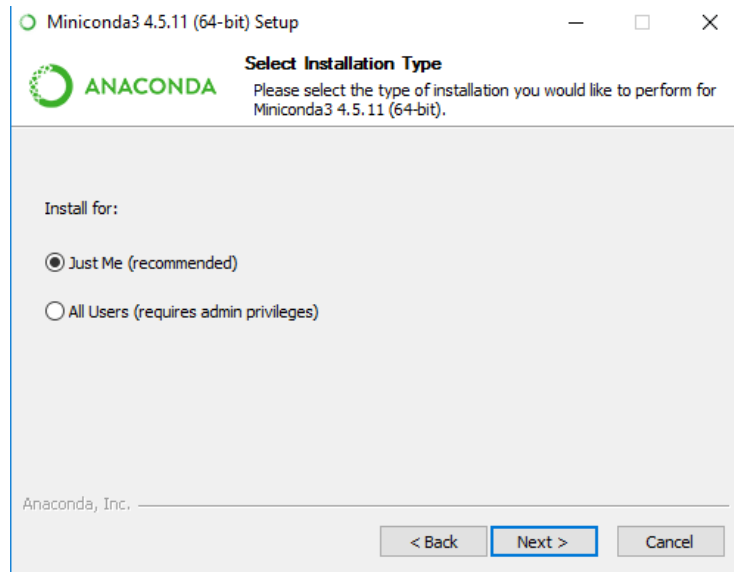
Figure 6: Select the first option (Just Me)
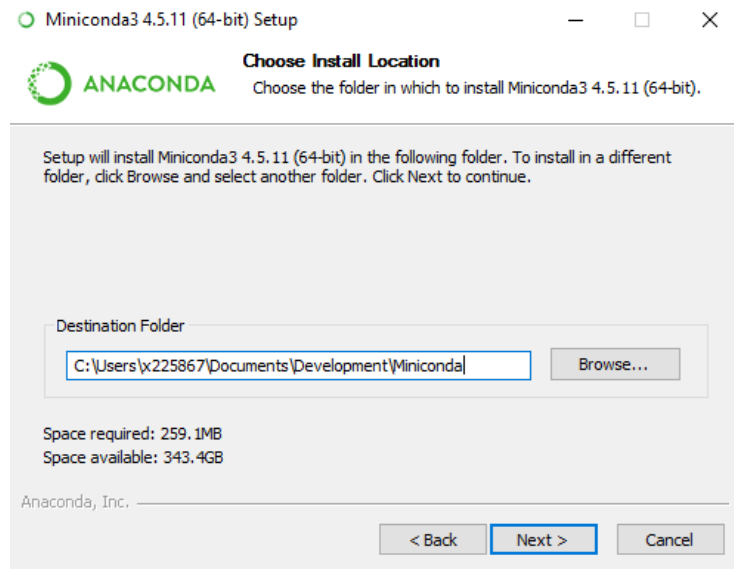
Select Installation Directory

Figure 7: You must install to a new folder in the Development directory
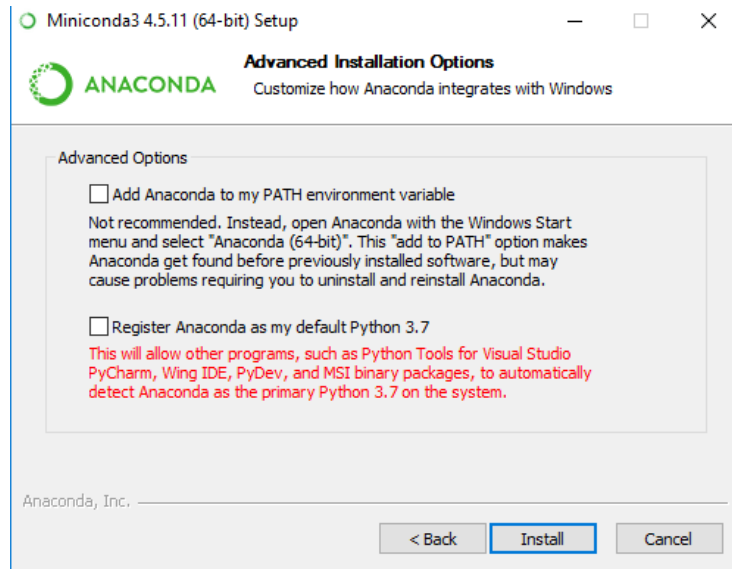
Advanced Options

Figure 8: isolate_install

N.B. Unselect "Register Anaconda in my default Python 3.7". You will not be able to select this unless you have administrative priveleges.
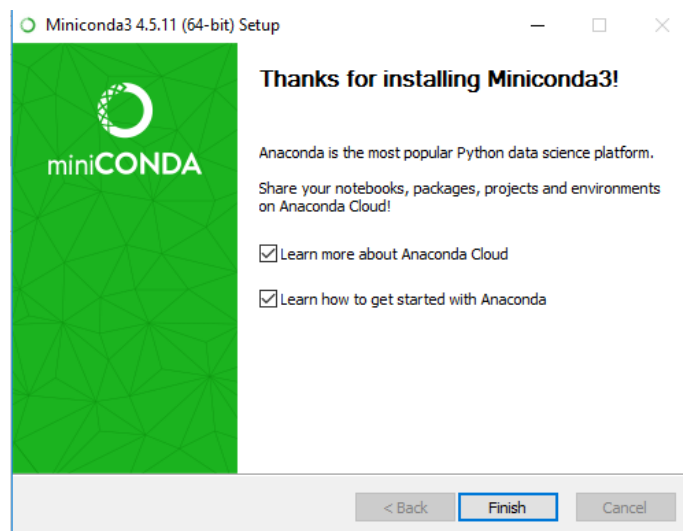

Figure 9: Finish

Installation of Jupyter

This section involves installation and configuration of jupyter.
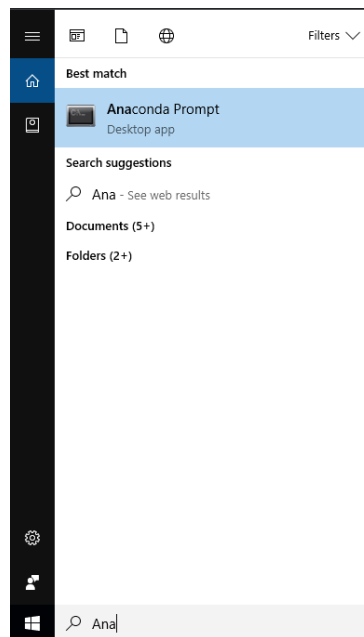
Anaconda Prompt

Figure 10: Press Windows hotkey and open Anaconda

Submit the following command at the prompt

conda install jupyter

Create a Jupyter Notebook Configuration File

Figure 11: "jupyter notebook –generate-config" generates a configuration file
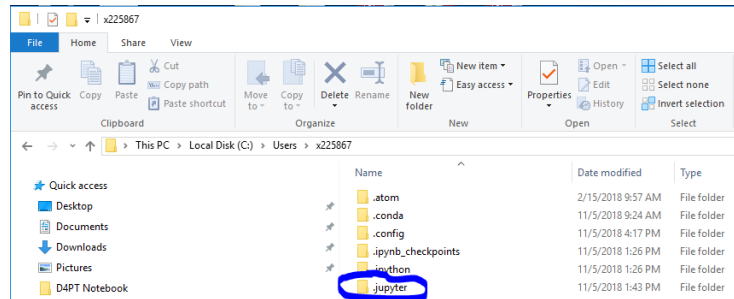
Find .jupyter in File Explorer

Figure 12: Open .jupyter folder (eg C:/Users/x225867/.jupyter)

NB Because the folder ".jupyter" has a dot preceding its name, it is a "hidden file". To view hidden files modify your view settings in file explorer.

Edit Configuration File

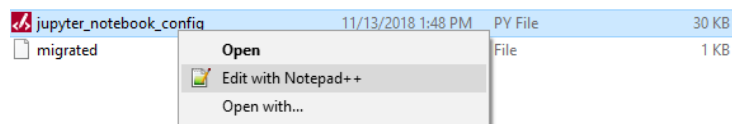The goal is to modify the directory where the notebook opens.

Figure 13: Edit jupyter_notebook_config with notepad++

Set notebook_dir to Development directory (line 262 in my file)
c.NotebookApp.notebook_dir='C:/Users/x225867/Documents/Development/'

In order for jupyter to see the new directory, We need to remove the "#" symbol before c.NotebookApp.notebook_dir.

```
## Gets or sets the maximum amount of memory, in bytes, that is allocated  for
#  use by the buffer manager.
#c.NotebookApp.max_buffer_size = 536870912

## Dict of Python modules to load as notebook server extensions.Entry values can
#  be used to enable and disable the loading ofthe extensions. The extensions
#  will be loaded in alphabetical order.
#c.NotebookApp.nbserver_extensions = {}

## The directory to use for notebooks and kernels.
#c.NotebookApp.notebook_dir = ''
c.NotebookApp.notebook_dir = 'C:/Users/<nuid>/Documents/Development/'

## Whether to open in a browser after starting. The specific browser used is
#  platform dependent and determined by the python standard library `webbrowser`
#  module, unless it is overridden using the --browser (NotebookApp.browser)
#  configuration option.
#c.NotebookApp.open_browser = True
```

Figure 14: Configure Jupyter to open in your analysis environment.

Master Markdown

R, Rstudio and IRkernel

This section brings our environment together; Rstudio is a useful IDE for working with R, and Jupyter allows us to communicate reproducible results in Python or R.

Start Anaconda Prompt
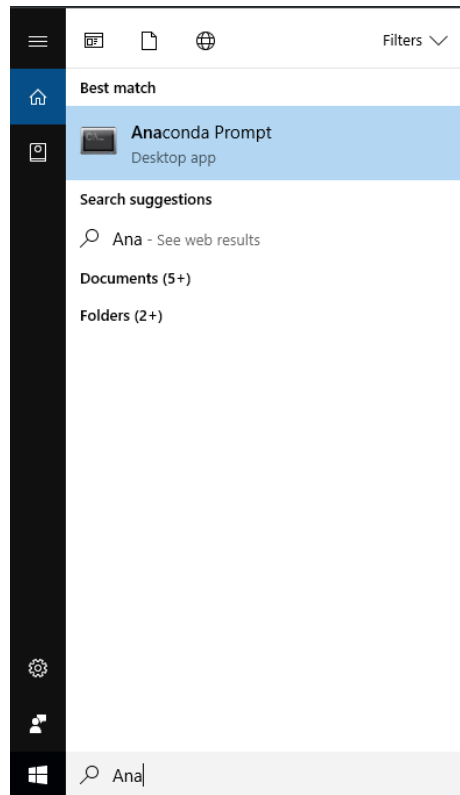
Figure 15: Start Anaconda Prompt

Install Base R

Figure 16: conda install r-base

Install RStudio

A ton of packages get installed, so expect a wait.

Figure 17: conda install -c r rstudio

Install IRkernel

Figure 18: conda install -c r r-irkernel