correlation one

Environment Setup

This document provides instructions intended to help you locally install Anaconda, a popular Python distribution that includes tools useful in data science workflows.

This document aims to support major operating systems, including Windows and OSX.

OSX:

Please issue all syntax highlighted commands into the terminal/shell of your choice. We recommend terminal or iterm2 as terminal clients.

1. Visit the Anaconda download portal and download the **Python 3.7** version appropriate for your operating system of choice. Follow instructions indicated by the installation wizard.

2. Using the terminal, create and navigate to a local workspace on your file system for DS4A:
   `cd ~/Desktop/DS4A_workspace`

3. Next, you must set the recently installed anaconda version as your default Python runtime.
   `export PATH="$HOME/anaconda3/bin/"`

   NOTE: If you choose to keep anaconda as your permanent Python installation, please add this line to your shell RC file (e.g., "~/.bashrc"). Otherwise, you must repeat this command from the workspace whenever you want to open a new Jupyter session.

4. Verify that `conda` installed properly by evoking the command from the shell:
   `conda -V`

   If the shell echoes "conda" followed by a version (e.g., "4.7.10"), then your installation succeeded.

5. Configure `conda` to work with `pip` to manage external dependencies. Run the following command:
   `conda config --set pip_interop_enabled True`

6. Next, you'll need to configure local software dependencies required for this exercise. Thankfully, Anaconda manages these downloads through a single file. You can download this file into your workspace as follows:

```
`cd <workspace>
wget https://ds4a.s3-us-west-2.amazonaws.com/setup/osx_environment.yml # change final line to include your <username>
conda env update --file osx_environment.yml`
```

   Please note manual intervention is necessary prior to calling `env update`, as the yml file is invalid until the path is updated.

   The `env update` command will default to conda but fall back on `pip` if the package is not available to conda.

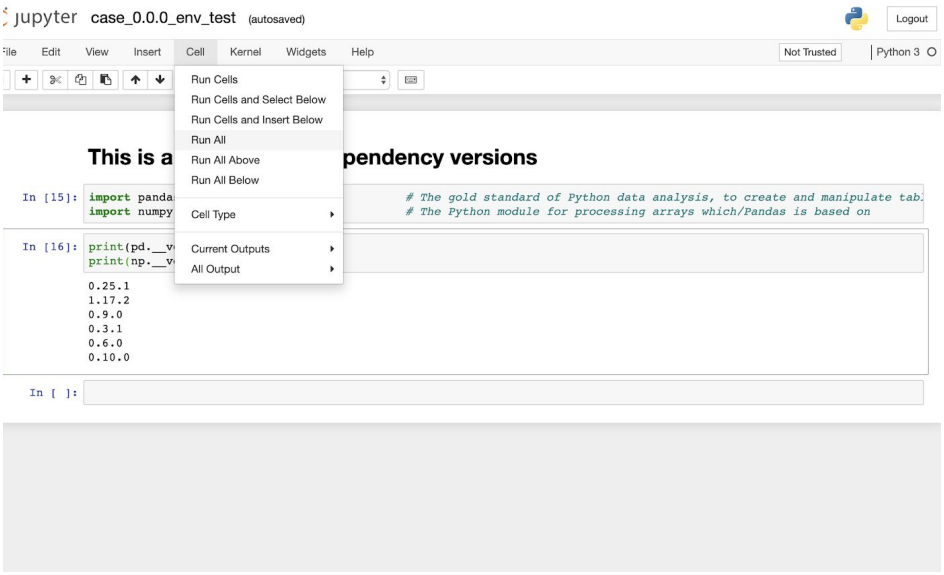
7. Next, download a copy of a template ipython notebook into your workspace directory.

```
`cd workspace
wget https://correlation1-public.s3-us-west-2.amazonaws.com/scripts/case_0.0.0_env_test.ipynb`
```

8. Launch jupyter notebook with the command below and navigate to and open up the case you just downloaded called case_0.0.0_env_test.ipynb.

```
`jupyter notebook`
```

9. Jupyter notebook opens a browser session with an interactive environment for editing snippets of Python code. You may test to see if the installation and software dependency retrieval worked by selecting "Run All" from the Cell dropdown item on the top menu.

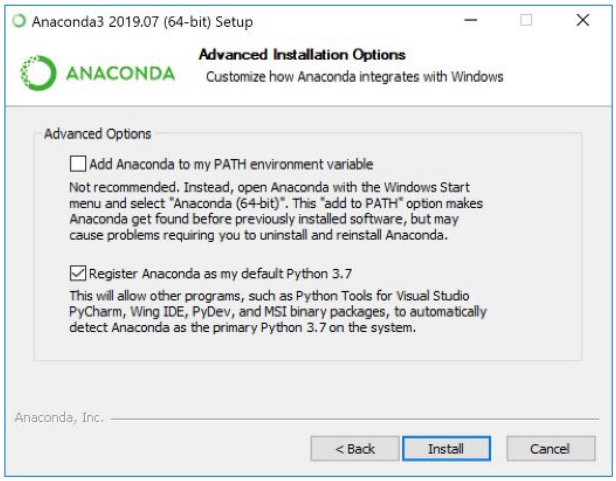


If running these snippets succeeds, your environment configuration most likely succeeded. Happy coding!

Windows:

1. Visit the Anaconda download portal and download the **Python 3.7** version appropriate for your operating system of choice. Follow instructions indicated by the installation wizard. For Windows, please use the default installation settings, which should set `conda` as your default `python` installation.

   NOTE: If prompted with the following screen, choose only the second checkbox (as done by default):



2. Create a folder on your Desktop called “DS4A_workspace”. The contents of today’s session will live in this folder.

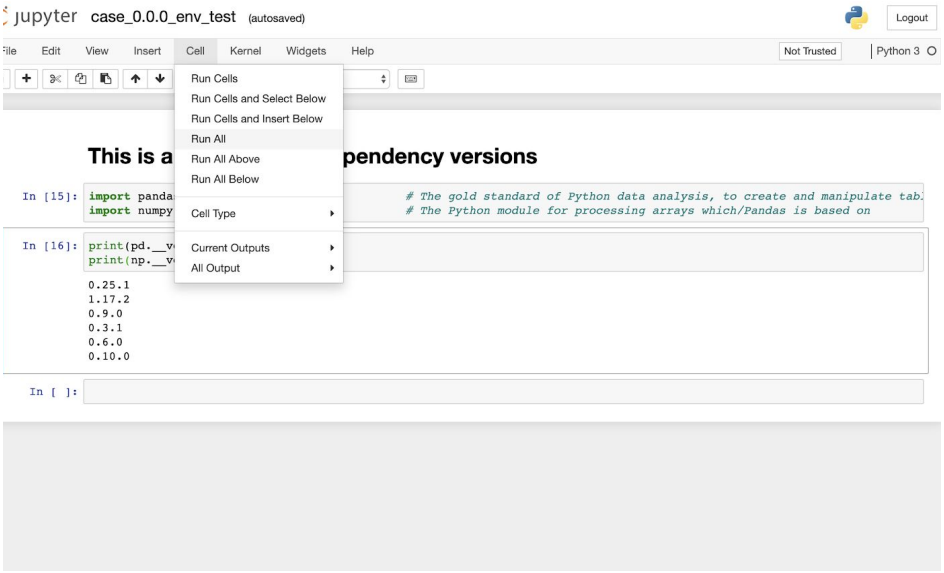
3. Download the files you will need for today’s session: window
4. s_environment.yml and iPython notebook. Please move these items from the default download location to the folder you created for DS4A on your Desktop.

5. Next, you must configure your environment. From the Windows Start menu, open the “Anaconda Navigator” application. Open the “Environments” tab on the left-hand side; “Import” the environmental configuration file referenced the previous step named windows_environment.yml. You may name the environment “DS4A_test”.

6. Open the “Jupyter Notebook” application from the Windows Start menu, your system might prompt with different options of browsers to use. We recommend Chrome, if available.

7. In the open Jupyter session open the case_0.0.0_env_test.ipynb file previously downloaded and stored in your local DS4A workspace. You may open files using the file dropdown menu.

8. Jupyter notebook opens a browser session with an interactive environment for editing snippets of python code. You may test to see if the installation and software dependency retrieval worked by selecting “Run All” from the Cell dropdown item on the top menu.



If running these snippets succeeds, your environment configuration most likely succeeded. Happy coding!

Ubuntu

1. Open the terminal of your choice.
2. Create a workspace directory with an associated install sub-directory.

   `mkdir -p ds4a_workspace/install`

3. In the previously created `install` sub-directory, download the anaconda3 install script:

   `wget https://repo.anaconda.com/archive/Anaconda3-2019.07-Linux-x86_64.sh`

4. Make the installer executable, then run it:

   `chmod +x <installer>.sh`
   `./<installer>.sh`

5. Follow the command-line prompt until the software is successfully installed, agreeing to licenses along the way. Agree to all defaults .Please respond 'yes' when the system prompts about running "conda init" (to initialize your system settings).

6. Assuming bash is your default shell, reload your shell configuration:

   `source ~/.bashrc`

7. Confirm that the installation was successful by prompting `conda` for a version tag:

   `conda --version`

   "conda <version>" will be printed to the prompt if the installation is successful.

8. Download necessary assets into the workspace directory:

   `wget https://ds4a.s3-us-west-2.amazonaws.com/setup/ubuntu_environment.yml`
   `wget https://correlation1-public.s3-us-west-2.amazonaws.com/scripts/case_0.0.0_env_test.ipynb`

9. Update your environment according to the downloaded configuration file:

   `conda env update --file ubuntu_environment.yml`

10. Start a notebook server and view through your browser. From the context of this notebook, open the `ipynb` file downloaded from s3.

    `jupyter notebook`

Troubleshooting:

1. Stuck? Consider consulting this list of Common Errors and Solutions.

2. If you cannot find a solution independently, please feel free to visit the Correlation-One #techsupport slack channel. We are happy to help!

Additional Resources:

1. Curious about `conda`? You may find helpful this page comparing `conda` with `pip` and `virtualenv`.