

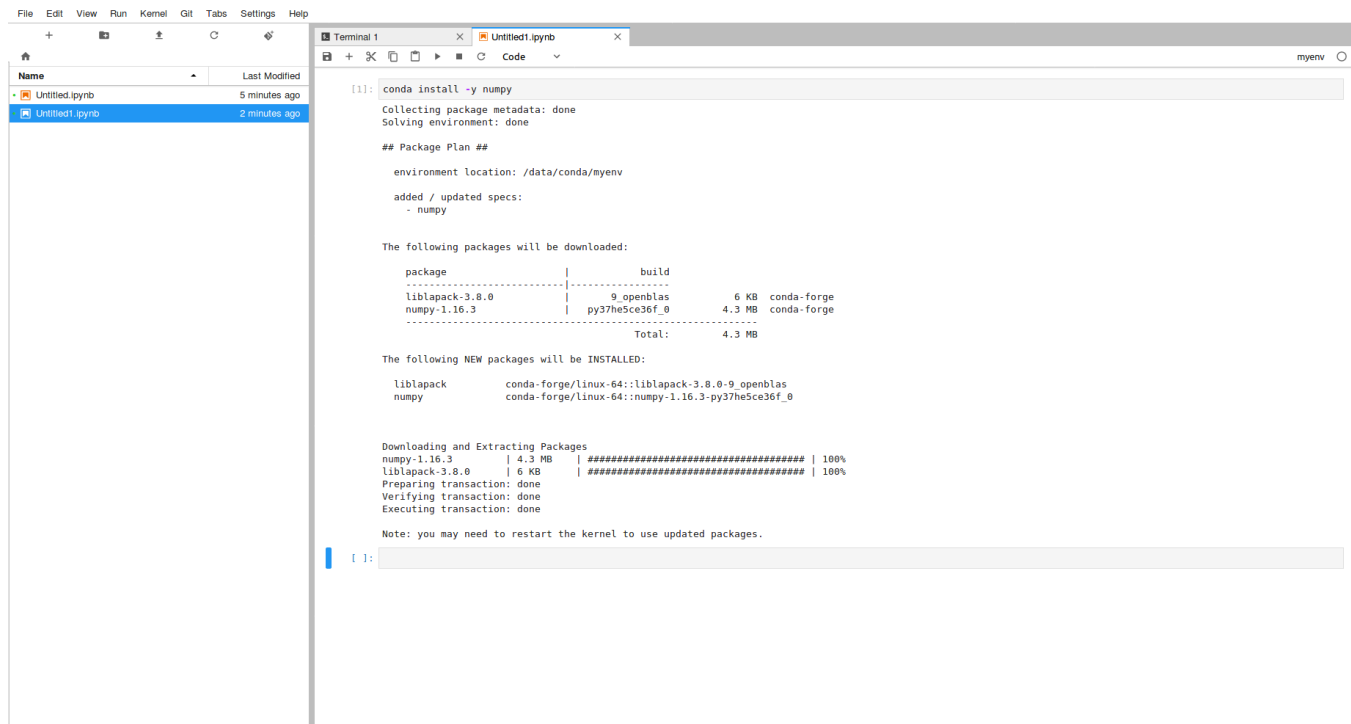
JupyterLab - Adding/Removing Packages

Adding Packages/Libraries to the New Environment

Packages can be added to the Conda environment via two mechanisms either of which work fine.

Python

Python has native Conda Environment support, hence from within a notebook using the Kernel simply calling Conda will install any package correctly to the current environment. This has the added advantage (if this is a python-based project), of capturing your package installs within a Notebook for ease of re-reproducibility for others not working in the same environment.



The screenshot shows the JupyterLab interface. On the left is a file browser with two notebooks: 'Untitled.ipynb' (modified 5 minutes ago) and 'Untitled1.ipynb' (modified 2 minutes ago). The main area displays a code cell in 'Untitled1.ipynb' with the command `conda install -y numpy`. The output shows the conda process installing numpy, including package metadata, a plan for downloading liblapack and numpy, and progress bars for the downloads.

```
[1]: conda install -y numpy
Collecting package metadata: done
Solving environment: done

## Package Plan ##

environment location: /data/conda/myenv

added / updated specs:
- numpy

The following packages will be downloaded:

package | build | size | source
-----|-----|-----|-----
liblapack-3.8.0 | 9_openblas | 6 KB | conda-forge
numpy-1.16.3 | py37he5ce36f_0 | 4.3 MB | conda-forge
Total: 4.3 MB

The following NEW packages will be INSTALLED:

liblapack conda-forge/linux-64::liblapack-3.8.0-9_openblas
numpy conda-forge/linux-64::numpy-1.16.3-py37he5ce36f_0

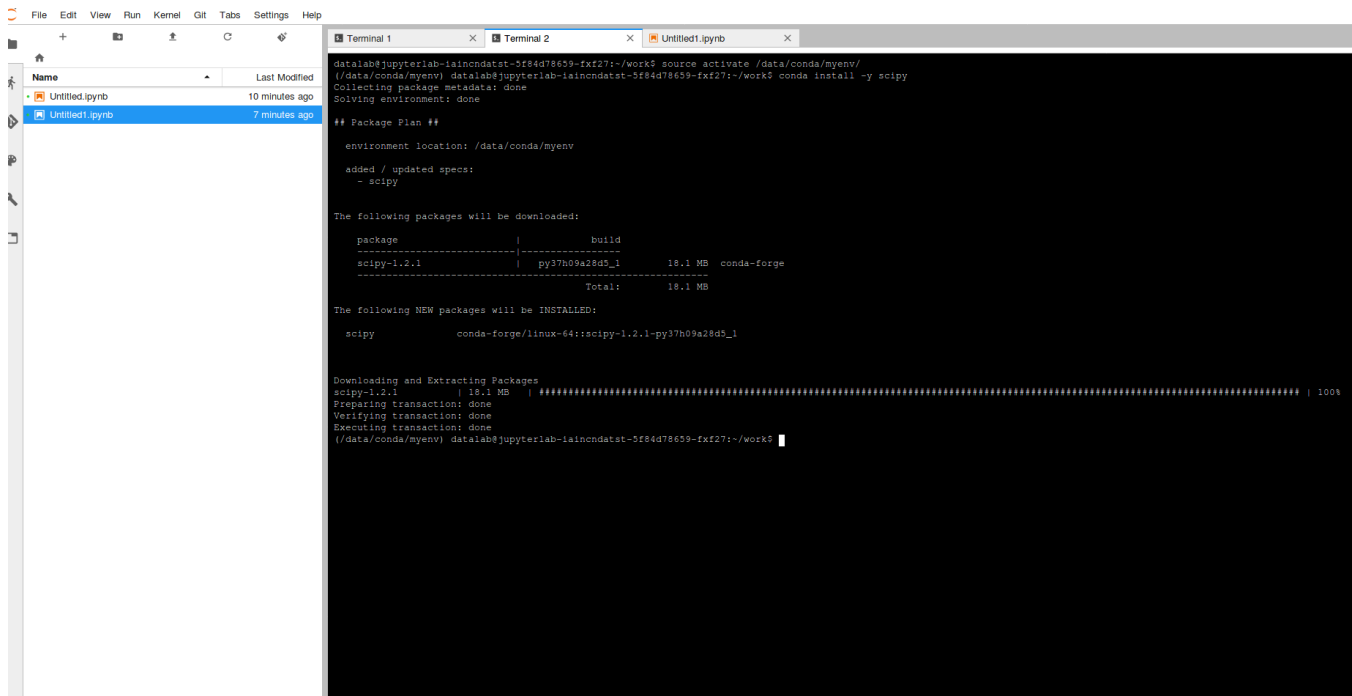
Downloading and Extracting Packages
numpy-1.16.3 | 4.3 MB | ##### | 100%
liblapack-3.8.0 | 6 KB | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done

Note: you may need to restart the kernel to use updated packages.
```

Anything Else

If instead of using python you want to install by other means, the terminal can also be used to install any Conda packages using the following example commands;

```
source activate /data/conda/<environment_name>
conda install -y <package>
# e.g
# source activate /data/conda/myenv
# conda install -y numpy
```



The screenshot shows a JupyterLab application window. On the left is a sidebar with a file browser and a list of files: 'Untitled.ipynb' (10 minutes ago) and 'Untitled.py' (7 minutes ago). The main area contains a terminal window with the following output:

```
data1ab@jupyterlab-laincndatst-5f84d78659-fxf27:~/work$ source activate /data/conda/myenv/
(/data/conda/myenv) data1ab@jupyterlab-laincndatst-5f84d78659-fxf27:~/work$ conda install -y scipy
Collecting package metadata: done
Solving environment: done

## Package Plan ##

environment location: /data/conda/myenv
added / updated specs:
- scipy

The following packages will be downloaded:

package | build
-----|-----
scipy-1.2.1 | py37h09a28d5_1 18.1 MB conda-forge
-----|-----
Total: 18.1 MB

The following NEW packages will be INSTALLED:

scipy conda-forge/linux-64::scipy-1.2.1-py37h09a28d5_1

Downloading and Extracting Packages
scipy-1.2.1 | 18.1 MB | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
(/data/conda/myenv) data1ab@jupyterlab-laincndatst-5f84d78659-fxf27:~/work$
```

Once packages are installed in your environment, they'll be persisted as normal.