

Installing NumPy

The only prerequisite for installing NumPy is Python itself. If you don't have Python yet and want the simplest way to get started, we recommend you use the [Anaconda Distribution](#) - it includes Python, NumPy, and many other commonly used packages for scientific computing and data science.

NumPy can be installed with `conda`, with `pip`, with a package manager on macOS and Linux, or [from source](#). For more detailed instructions, consult our [Python and NumPy installation guide](#) below.

CONDA

If you use `conda`, you can install NumPy from the `defaults` or `conda-forge` channels:

```
# Best practice, use an environment rather than install in the base

env conda create -n my-env

conda activate my-env

# If you want to install from conda-forge

conda config --env --add channels conda-forge

# The actual install command

conda install numpy
```

PIP

If you use `pip`, you can install NumPy with:

```
pip install numpy
```

Also when using `pip`, it's good practice to use a virtual environment - see [Reproducible Installs](#) below for why, and [this guide](#) for details on using virtual environments.

Python and NumPy installation guide

Installing and managing packages in Python is complicated, there are a number of alternative solutions for most tasks. This guide tries to give the reader a sense of the best (or most popular) solutions, and give clear recommendations. It focuses on users of Python, NumPy, and the PyData (or numerical computing) stack on common operating systems and hardware.

Recommendations

We'll start with recommendations based on the user's experience level and operating system of interest. If you're in between "beginning" and "advanced", please go with "beginning" if you want to keep things simple, and with "advanced" if you want to work according to best practices that go a longer way in the future.