

Module 4: Intro to Matplotlib

October 15, 2025

Content covered:

H1

1. Classes
2. Jupyter Notebooks
3. Intro to Matplotlib
 1. Pyplot

Classes

Classes, also called Objects, are containers in Python that can store values, known as `attributes`; and functions, known as `methods`

Anatomy of a Class

Classes are defined using `class` keyword:

```
class MyClass:  
    def __init__(self, attribute1):  
        self.attribute1 = attribute1
```

Classes cont.

You can also include functions that will be stored in the object by default too:

```
class MyClass:  
    def __init__(self, attribute1):  
  
        self.attribute1 = attribute1  
  
    def stringify(self, value):  
  
        return str(value)
```

Jupyter Notebooks

Matplotlib is the main plotting library in Python.

It's been used to plot generate the first image of a black hole!



The most common submodule in `matplotlib` that you will use is `pyplot`. A submodule is a collection of scripts that belong to a module/package.

A common Python pattern for interacting with this submodule is:

```
import matplotlib.pyplot as plt
```

The keyword `as` works as an alias, which lets you map an import to another variable name.

We saw this when we imported `numpy`:

```
import numpy as np
```

Let's actually bring together what we learned with `numpy` with some new functionality with `matplotlib`.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(0, 2 * np.pi, 200)
y = np.sin(x)

fig, ax = plt.subplots()
ax.plot(x, y)
plt.show()
```

Anatomy of Plotting

```
plt.plot([2, 4, 6, 8])
```

If only a single array is specified, `pyplot` defaults to the input array being the y-values and the x-values will be default to a `range(len(input_array))`, i.e. `[0, 1, 2, 3]`.