

# Introduction to matplotlib

Olav Vahtras

Leiden 2015-04-13

# Matplotlib

- The standard 2D-plotting library in Python
- Production-quality graphs
- Interactive and non-interactive use
- Many output formats
- Flexible and customizable

# First example

## The absolute minimum you need to know

- You have a set of points (x,y) on file

```
-3.141593 -0.000000  
-3.013364 -0.127877  
-2.885136 -0.253655  
...  
3.141593 0.000000
```

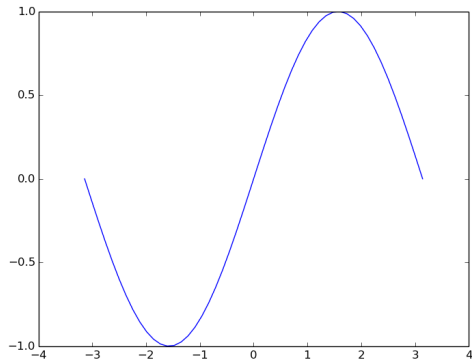
# First example

## The absolute minimum you need to know

- You have a set of points (x,y) on file

```
-3.141593 -0.000000  
-3.013364 -0.127877  
-2.885136 -0.253655  
...  
3.141593 0.000000
```

- How do you get to this



## Next

- Import the plotting library

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

## Next

- Import the plotting library

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

- Load the data from file

```
data = np.loadtxt('filename')
```

## Next

- Import the plotting library

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

- Load the data from file

```
data = np.loadtxt('filename')
```

- Call the `plot` function

```
plt.plot(data[:, 0], data[:, 1])
```

## Next

- Import the plotting library

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

- Load the data from file

```
data = np.loadtxt('filename')
```

- Call the `plot` function

```
plt.plot(data[:, 0], data[:, 1])
```

- Show the result

```
plt.show()
```

*Note:* in ipython notebook you may want to do

```
%matplotlib inline
```



# Next?

## Refinement

- Change color, linestyle, linewidth

# Next?

## Refinement

- Change color, linestyle, linewidth
- Change window size (ylim)

# Next?

## Refinement

- Change color, linestyle, linewidth
- Change window size (ylim)
- Change xticks

# Next?

## Refinement

- Change color, linestyle, linewidth
- Change window size (ylim)
- Change xticks
- Set title

# Next?

## Refinement

- Change color, linestyle, linewidth
- Change window size (ylim)
- Change xticks
- Set title
- Multi-line plots

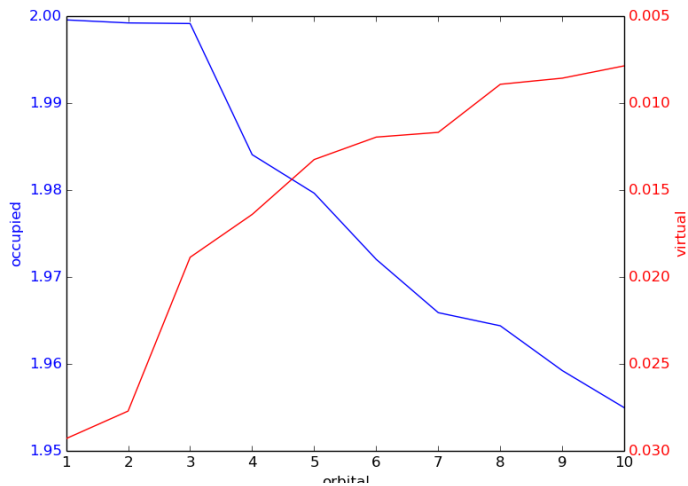
# Next?

## Refinement

- Change color, linestyle, linewidth
- Change window size (ylim)
- Change xticks
- Set title
- Multi-line plots
- Legends

## A more advanced example

```
fig, ax1 = plt.subplots()
ax1.plot(orbital, dataset1, 'b-')
ax1.set_xlabel('orbital')
# Make the y-axis label and tick labels match the line color.
ax1.set_ylabel('occupied', color='b')
for tl in ax1.get_yticklabels():
    tl.set_color('b')
ax2 = ax1.twinx()
ax2.plot(orbital, dataset2, 'r-')
ax2.set_ylabel('virtual', color='r')
ax2.set_ylim(ax2.get_ylim()[::-1])
for tl in ax2.get_yticklabels():
    tl.set_color('r')
```



## In practice

- How do you do when need a particular type of figure?



## In practice

- How do you do when need a particular type of figure?
- Go to the matplotlib gallery: <http://matplotlib.org/gallery>

## In practice

- How do you do when need a particular type of figure?
- Go to the matplotlib gallery: <http://matplotlib.org/gallery>
- Try some exercises at

<http://scipy-lectures.github.io/intro/matplotlib/matplotlib.html#other-types-of-plots-examples-and-exercises>

More

