Python for Science and Engg: Plotting experimental data

FOSSEE

Department of Aerospace Engineering
IIT Bombay

SciPy.in 2010, Tutorials

Outline

1. Plotting Points
2. Lists
3. Simple Pendulum
4. Summary

Outline

1 Plotting Points

2 Lists

3 Simple Pendulum

4 Summary

Why would I plot f(x)?

Do we plot analytical functions or experimental data?

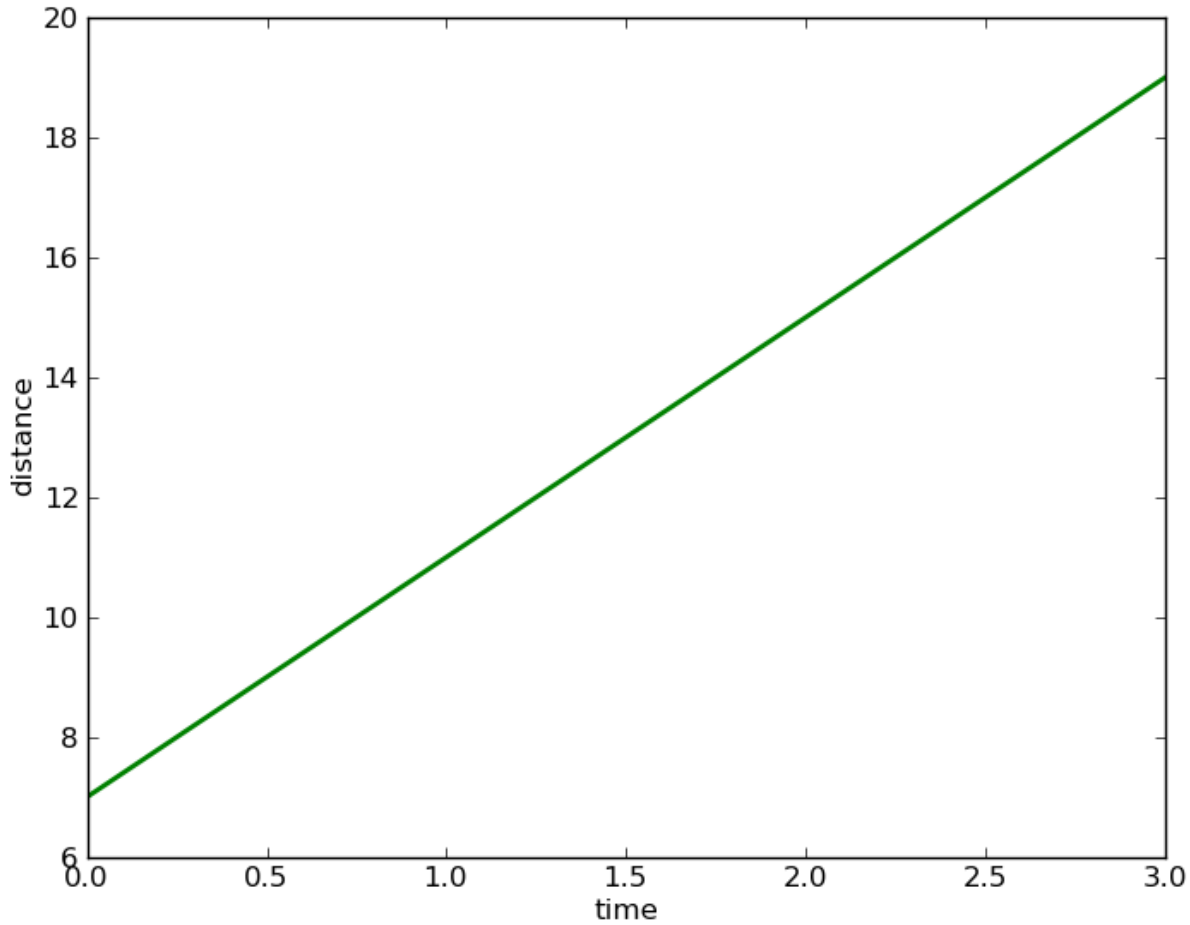
```
In [ ]: time = [0, 1, 2, 3]

In [ ]: distance = [7, 11, 15, 19]

In [ ]: plot(time, distance)
Out[ ]: [<matplotlib.lines.Line2D object at 0xa73a

In [ ]: xlabel('time')
Out[ ]: <matplotlib.text.Text object at 0x986e9ac>

In [ ]: ylabel('distance')
Out[ ]: <matplotlib.text.Text object at 0x98746ec>
```



Is this what you have?

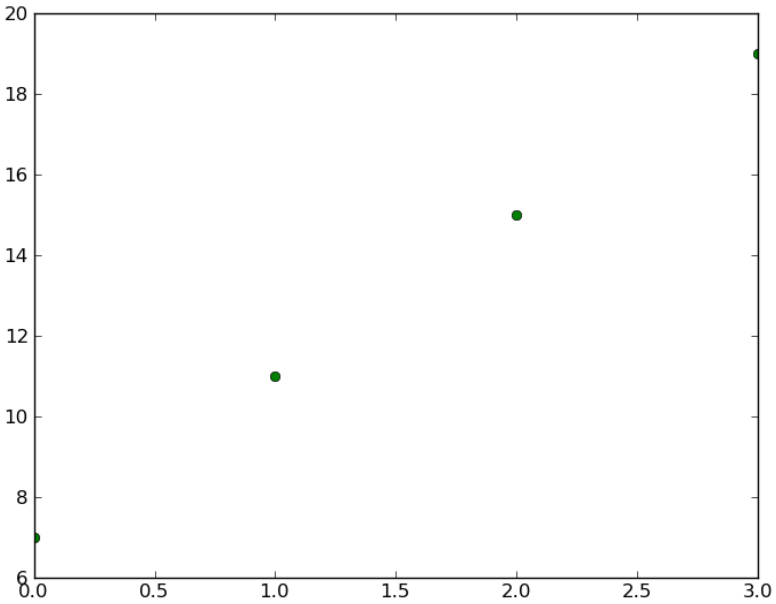
Plotting points

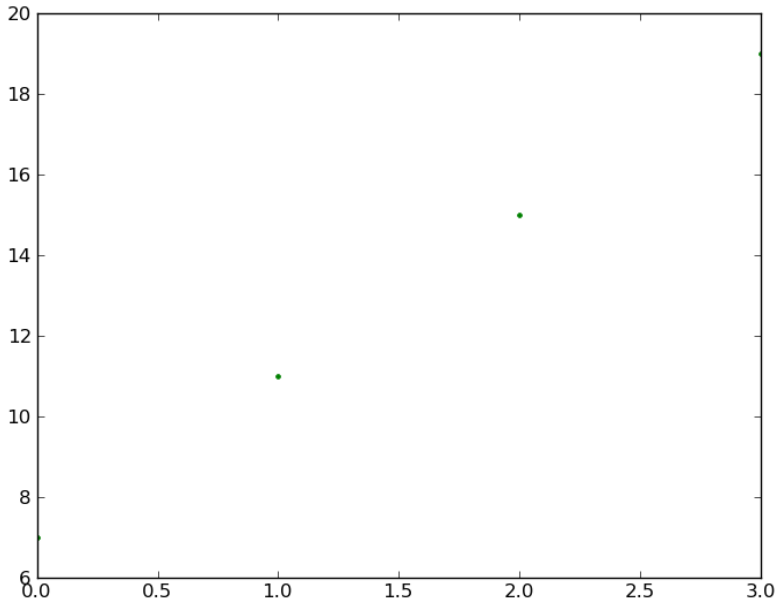
- What if we want to plot the points?

```
In [ ]: clf()

In [ ]: plot(time, distance, 'o')
Out[ ]: [<matplotlib.lines.Line2D object

In [ ]: clf()
In [ ]: plot(time, distance, '.')
Out[ ]: [<matplotlib.lines.Line2D object
```


20
18
16
14
12
10
8
6
0.0
0.5
1.0
1.5
2.0
2.5
3.0


20
18
16
14
12
10
8
6
0.0
0.5
1.0
1.5
2.0
2.5
3.0

Additional Line Styles

- `'o'` - Filled circles
- `'.'` - Small Dots
- `'-'` - Lines
- `'--'` - Dashed lines

Outline

1 Plotting Points

2 Lists

3 Simple Pendulum

4 Summary

Lists: Introduction

```
In []: time = [0, 1, 2, 3]

In []: distance = [7, 11, 15, 19]
```

What are **x** and **y**?

lists!!

Lists: Initializing & accessing elements

```
In []: mtlist = []
```

Empty List

```
In []: p = [ 2, 3, 5, 7]

In []: p[1]
Out[]: 3

In []: p[0]+p[1]+p[-1]
Out[]: 12
```

List: Appending elements

```
In [ ]: p.append(11)
In [ ]: p
Out[ ]: [ 2, 3, 5, 7, 11]

In [ ]: b = [11, 13, 17]
In [ ]: p.append(b)
Out[ ]: [ 2, 3, 5, 7, 11, [11, 13, 17]]
```

Outline

1 Plotting Points

2 Lists

3 Simple Pendulum

4 Summary

Simple Pendulum - L and T

Let us look at the Simple Pendulum experiment.

L	T	T^2
0.1	0.69	
0.2	0.90	
0.3	1.19	
0.4	1.30	
0.5	1.47	
0.6	1.58	
0.7	1.77	
0.8	1.83	
0.9	1.94	

$L \alpha T^2$

Our data is present in **`pendulum.txt`**. Let's look at it.

Gotcha and an aside

Ensure you are in the same directory as **`pendulum.txt`**
if not, do the following on IPython:

```
In []: %cd directory_containing_file
# Check if pendulum.txt is there.
In []: ls
# Also try
In []: !ls
```

Note: **`%cd`** is an IPython magic command. For more information do:

```
In []: ?
```

Looking at `pendulum.txt`

```
In []: cat pendulum.txt
1.0000e-01 6.9004e-01
1.1000e-01 6.9497e-01
1.2000e-01 7.4252e-01
1.3000e-01 7.5360e-01

...
```

- File contains L vs. T values
- First Column – L values
- Second Column – T values

loadtxt

- We shall use the **loadtxt** command to load data
- Let's use **primes.txt** file learn to use it
- **primes.txt** has a single column
- Then, we shall use it for **pendulum.txt** – 2 cols

What's in `primes.txt`?

Lets look at the **`primes.txt`** file.

```
In []: cat primes.txt
2
3
5
7
11
13
.
.
.
```

loadtxt ...

```
In [ ]: primes = loadtxt('primes.txt')
In [ ]: primes
Out[ ]: array([ 2.,  3.,  5.,  7.,  11.,  13.,
                17.,  19.,  23.,  29.,  31.,
                37.,  41.,  43.,  47.,  53.,
                59.,  61.,  67.,  71.,  73.,
                79.,  83.,  89.,  97.])
```

Reading `pendulum.txt`

```
In [ ]: pend = loadtxt('pendulum.txt')
In [ ]: pend
```

- **pend** is 2 Dimensional.
- We don't yet know how to handle it.
- We obtain 1D sequences using **unpack=True**

```
In [ ]: L, T = loadtxt('pendulum.txt',
                        unpack=True)
In [ ]: print L, T
In [ ]: print len(L), len(T)
Out[ ]: 90 90
```

Plotting L vs T^2

- We must square each of the values in `T`
- How to do it?
- We use a `for` loop to iterate over `T`

Plotting L vs T^2

```
In [ ]: tsq = []

In [ ]: for time in T:
  ....:     tsq.append(time*time)
  ....:
  ....:
```

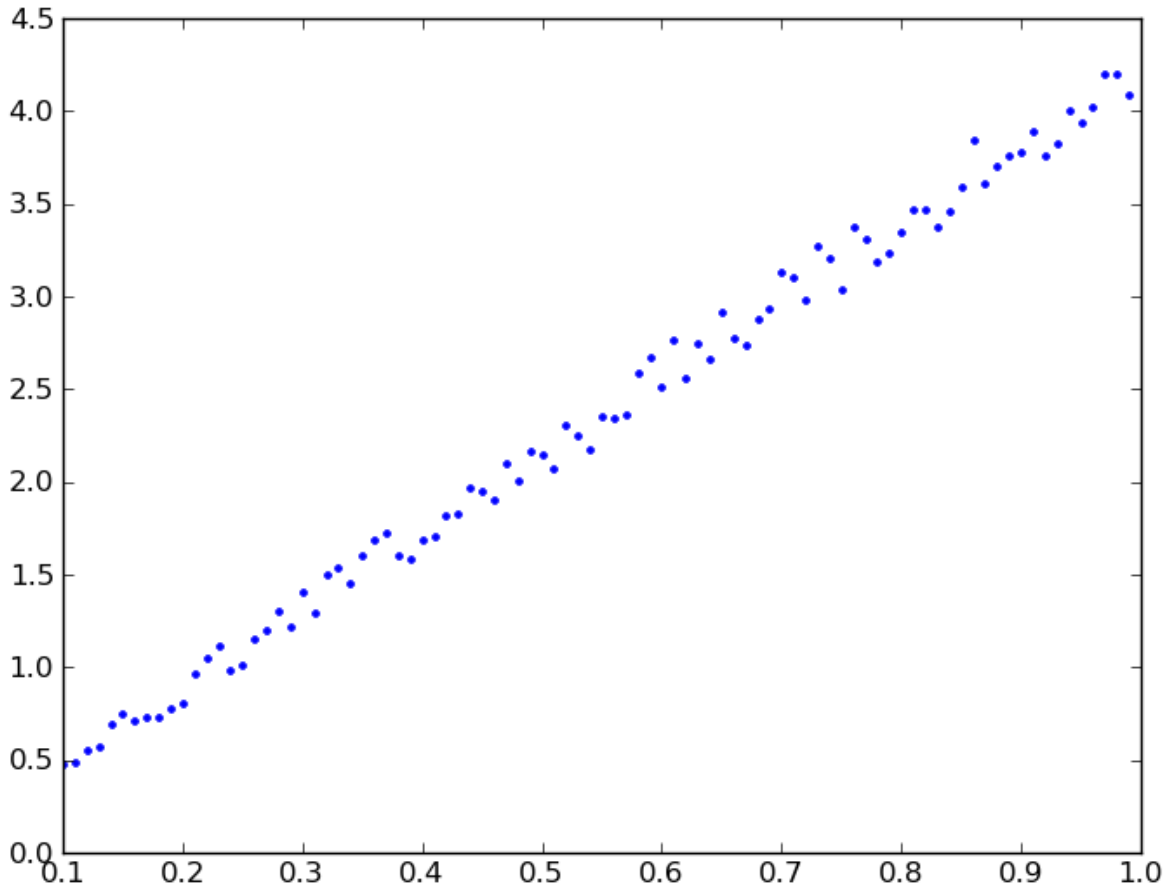
Hit "ENTER" key twice, to get out of `for` loop.

```
In [ ]: print len(tsq)
Out[ ]: 90
```

`tsq` is a `list` of squares of `T` values.

Plotting L vs T^2 . . .

```
In []: plot(L, tsq, '.')
```



Outline

1 Plotting Points

2 Lists

3 Simple Pendulum

4 Summary

What did we learn?

- Plot attributes and plotting points
- Lists
- `for`
- Loading data from files