

Filename: ./mprun\_demo.py

Line #	Mem usage	Increment	Line Contents
4	71.9 MiB	0.0 MiB	L = [j ^ (j >> i) for j in range(N)]

Filename: ./mprun\_demo.py

Line #	Mem usage	Increment	Line Contents
1	39.0 MiB	0.0 MiB	def sum_of_lists(N):
2	39.0 MiB	0.0 MiB	total = 0
3	46.5 MiB	7.5 MiB	for i in range(5):
4	71.9 MiB	25.4 MiB	L = [j ^ (j >> i) for j in range(N)]
5	71.9 MiB	0.0 MiB	total += sum(L)
6	46.5 MiB	-25.4 MiB	del L # remove reference to L
7	39.1 MiB	-7.4 MiB	return total

Here the Increment column tells us how much each line affects the total memory budget: observe that when we create and delete the list L, we are adding about 25 MB of memory usage. This is on top of the background memory usage from the Python interpreter itself.

For more information on `%memit` and `%mprun`, as well as their available options, use the IPython help functionality (i.e., type `%memit?` at the IPython prompt).

## More IPython Resources

In this chapter, we've just scratched the surface of using IPython to enable data science tasks. Much more information is available both in print and on the Web, and here we'll list some other resources that you may find helpful.

### Web Resources

#### *The IPython website*

The IPython website links to documentation, examples, tutorials, and a variety of other resources.

#### *The nbviewer website*

This site shows static renderings of any IPython notebook available on the Internet. The front page features some example notebooks that you can browse to see what other folks are using IPython for!

### *A Gallery of Interesting IPython Notebooks*

This ever-growing list of notebooks, powered by nbviewer, shows the depth and breadth of numerical analysis you can do with IPython. It includes everything from short examples and tutorials to full-blown courses and books composed in the notebook format!

### *Video tutorials*

Searching the Internet, you will find many video-recorded tutorials on IPython. I'd especially recommend seeking tutorials from the PyCon, SciPy, and PyData conferences by Fernando Perez and Brian Granger, two of the primary creators and maintainers of IPython and Jupyter.

## **Books**

### *Python for Data Analysis*

Wes McKinney's book includes a chapter that covers using IPython as a data scientist. Although much of the material overlaps what we've discussed here, another perspective is always helpful.

### *Learning IPython for Interactive Computing and Data Visualization*

This short book by Cyrille Rossant offers a good introduction to using IPython for data analysis.

### *IPython Interactive Computing and Visualization Cookbook*

Also by Cyrille Rossant, this book is a longer and more advanced treatment of using IPython for data science. Despite its name, it's not just about IPython—it also goes into some depth on a broad range of data science topics.

Finally, a reminder that you can find help on your own: IPython's ?-based help functionality (discussed in [“Help and Documentation in IPython” on page 3](#)) can be very useful if you use it well and use it often. As you go through the examples here and elsewhere, you can use it to familiarize yourself with all the tools that IPython has to offer.