



Data Science Upskilling Workshop

Session 2: Programming, Best Practices & Scientific Libraries

Oren Livne and Jiangang Hao
Educational Testing Service

NCME 2022 Training Workshop – 4/10/2022

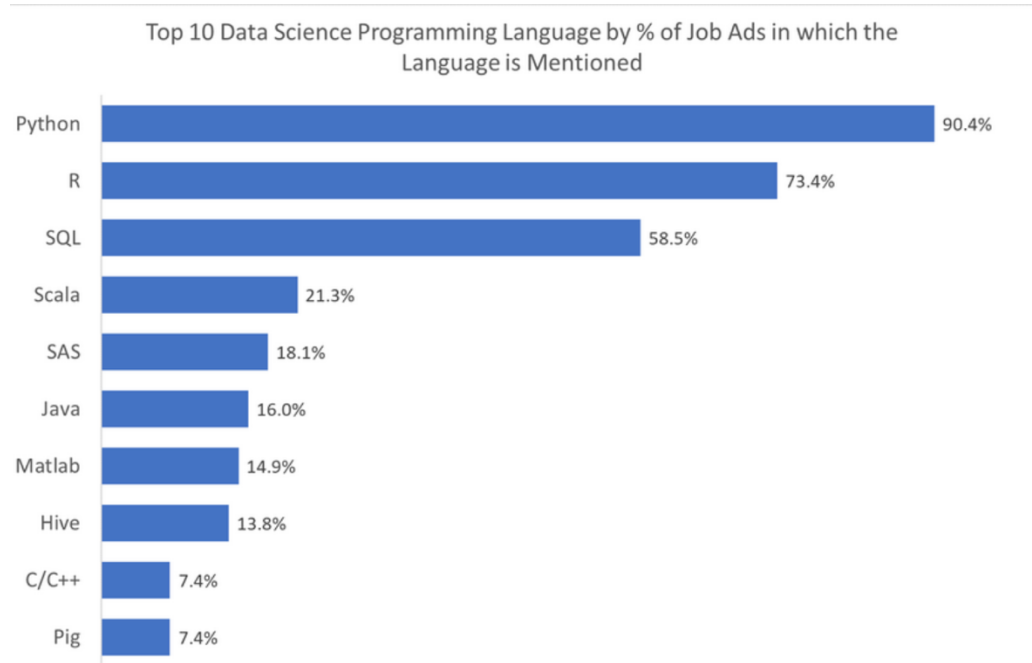


Overview of Python Programming

```
31 def __init__(self):
32     self.file = None
33     self.fingerprints = set()
34     self.logdupes = True
35     self.debug = debug
36     self.logger = logging.getLogger(__name__)
37     if path:
38         self.file = open(os.path.join(path, 'requests.txt'),
39                           'a')
40         self.file.seek(0)
41         self.fingerprints.update(re.sub(r'[\s\S]+', '', self.file))
42
43 @classmethod
44 def from_settings(cls, settings):
45     debug = settings.getbool('SUPERFINGER_DEBUG')
46     return cls(job_dir(settings), debug)
47
48 def request_seen(self, request):
49     fp = self.request_fingerprint(request)
50     if fp in self.fingerprints:
51         return True
52     self.fingerprints.add(fp)
53     if self.file:
54         self.file.write(fp + os.linesep)
55
56 def request_fingerprint(self, request):
57     return request_fingerprint(request)
```

Why Python for Data Science?

- Easy to learn
- Powerful
- Scalable
- Many libraries
 - Machine Learning
 - Deep Learning
 - Visualization
- Python Community



Overview of the Python Language: 1/2

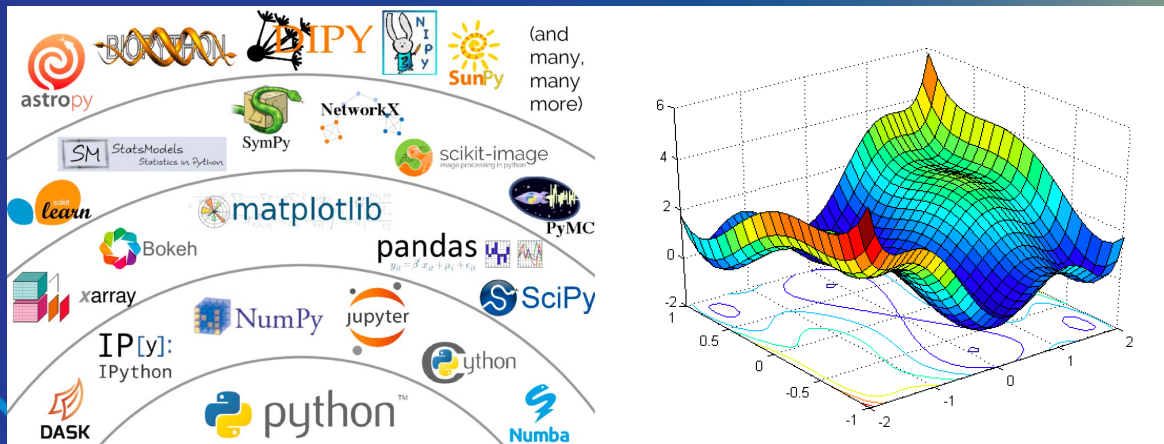
- Data Types
 - None = missing value
- Flow control
 - If-else
 - Loop
 - Function
- Data Structures
 - List []
 - Tuple ()
 - Set {}
 - Dictionary {}
- Printing & formatting
- Importing packages; getting help
- [Python Style Guide](#)
- List comprehension: `[x**2 for x in range(10)]`

Overview of the Python Language: 2/2

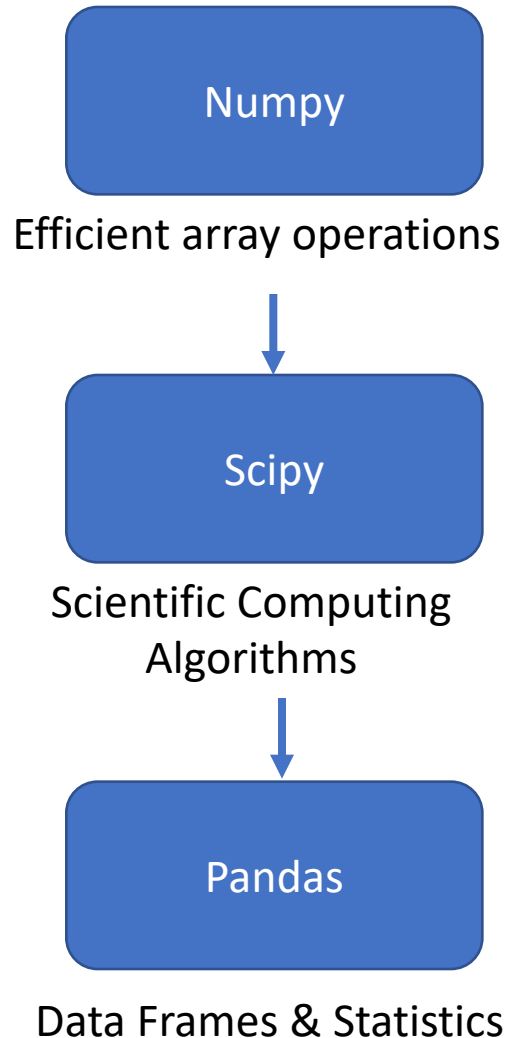
- Recap: data structures, generators
- Index ranges, negative index, str as list
- Filtering lists with list comprehension
- Functional programming: map, reduce, filter
- timeit
- Zip
- Further reading:
 - <https://docs.python.org/3/tutorial/>



Scientific Computing Packages



Basic Data Science Packages



Package Highlights

- **Numpy:** [Tutorial Notebook](#)
- **Scipy:** <https://scipy-lectures.org/intro/scipy.html>
 - scipy.linalg - linear algebra (1.6.3)
 - scipy.curvefit – curve fitting (1.6.5)
 - scipy.stats - statistics and random numbers (1.6.6)
 - Statistical testing
- **Pandas** [Tutorial Notebook](#)
 - Series
 - DataFrame
 - Reading a CSV file
- Further reading:
 - <https://numpy.org/doc/stable/user/quickstart.html>
 - [10 minutes to pandas](#)

Interfacing with other Languages

- Calling an R function from Python
 - [Call R for computing -> numpy array](#)
 - [Call R -> pandas DataFrame](#)

Session 2 - Exercises

1. Complete numpy tutorial assignments in the bottom of the `numpy_tutorial.ipynb` notebook under the Session 2 materials on the Teams site.
2. Calculate summary statistics: mean, std, 0%,10%,20%,..., percentiles (look for how to do that in the numpy documentation).
3. Run a t-test of two Gaussian distribution means using scipy (see <https://scipy-lectures.org/intro/scipy.html>, Sec. 1.6.6.3).