

Lecture 11: Array programming with numpy

Instructor: Michael Szell

Oct 4, 2023

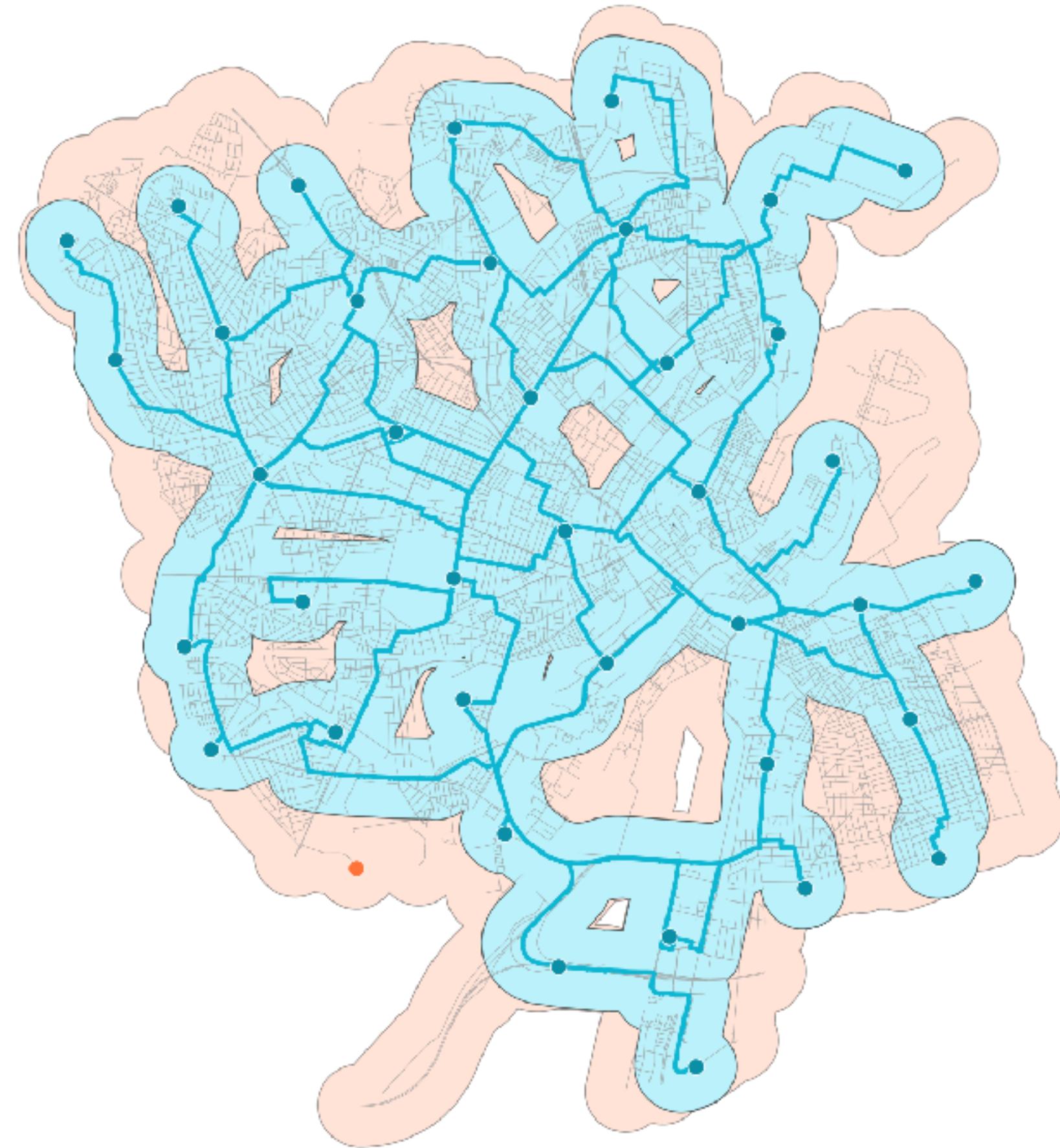


Data Scientist researching:



Human mobility

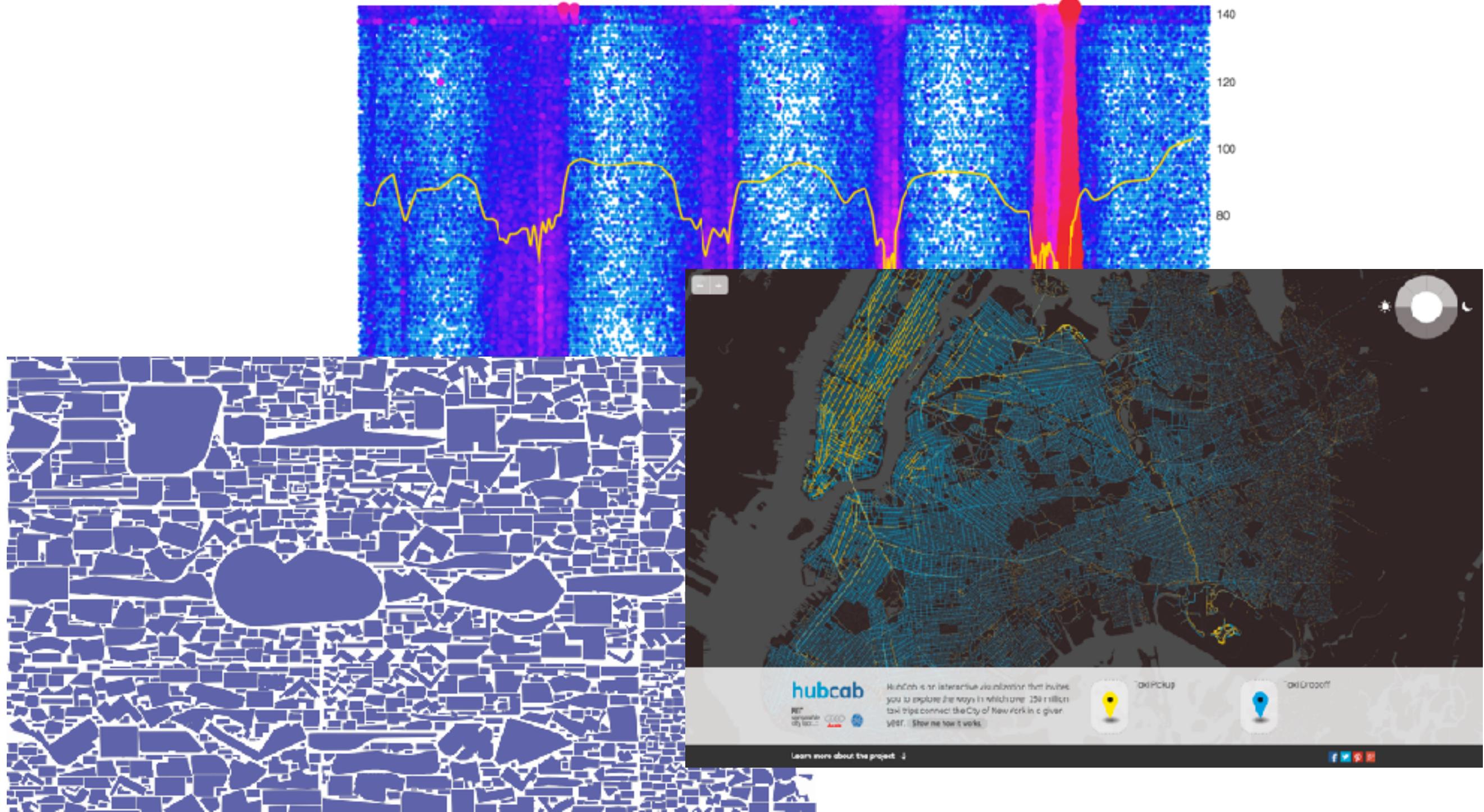
Python



Urban infrastructure networks

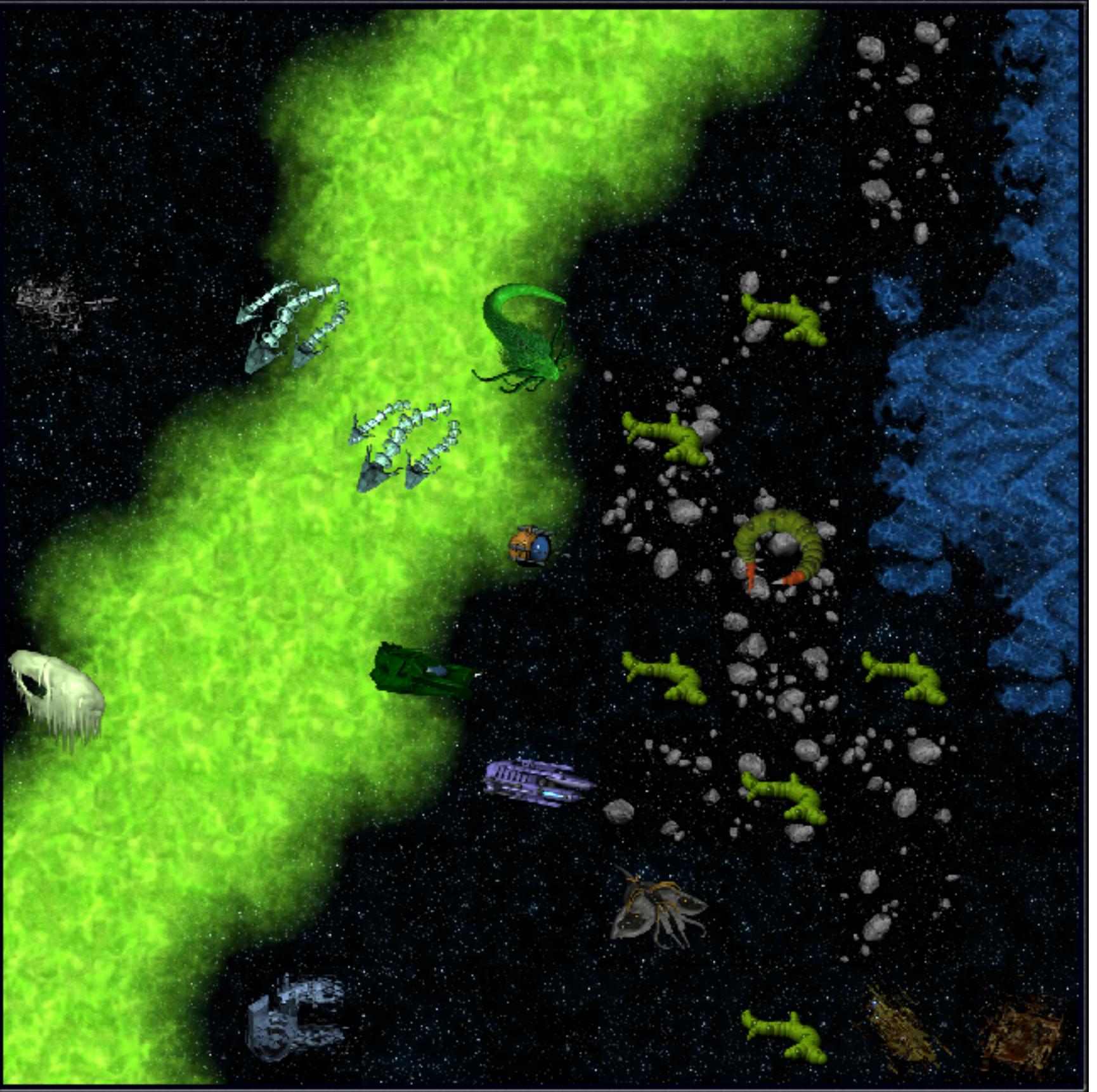
Introduction: Michael Szell

Data Scientist creating:



Data Visualizations

Python, Javascript, MongoDB



MMOG

PHP, C/C++, MySQL



We are researching network and data science applications to social systems

We are interdisciplinary: Computer Science, Physics, Mathematics, GIS

We are alumni of MIT, Harvard, Northeastern, Bell Labs, United Nations

We won the Young Academy's prize Research Environment of the Year 2022

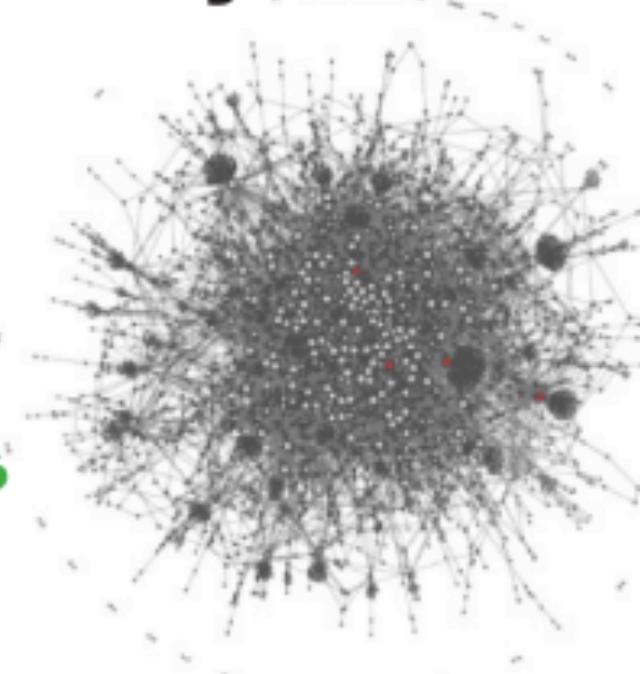
VILLUM FONDEN

CARLSBERG FOUNDATION

Vejdirektoratet
Innovationsfonden



Social Networks
Bicycle Networks
Complex Systems
NLP



We have finished the "Python Crash Course"

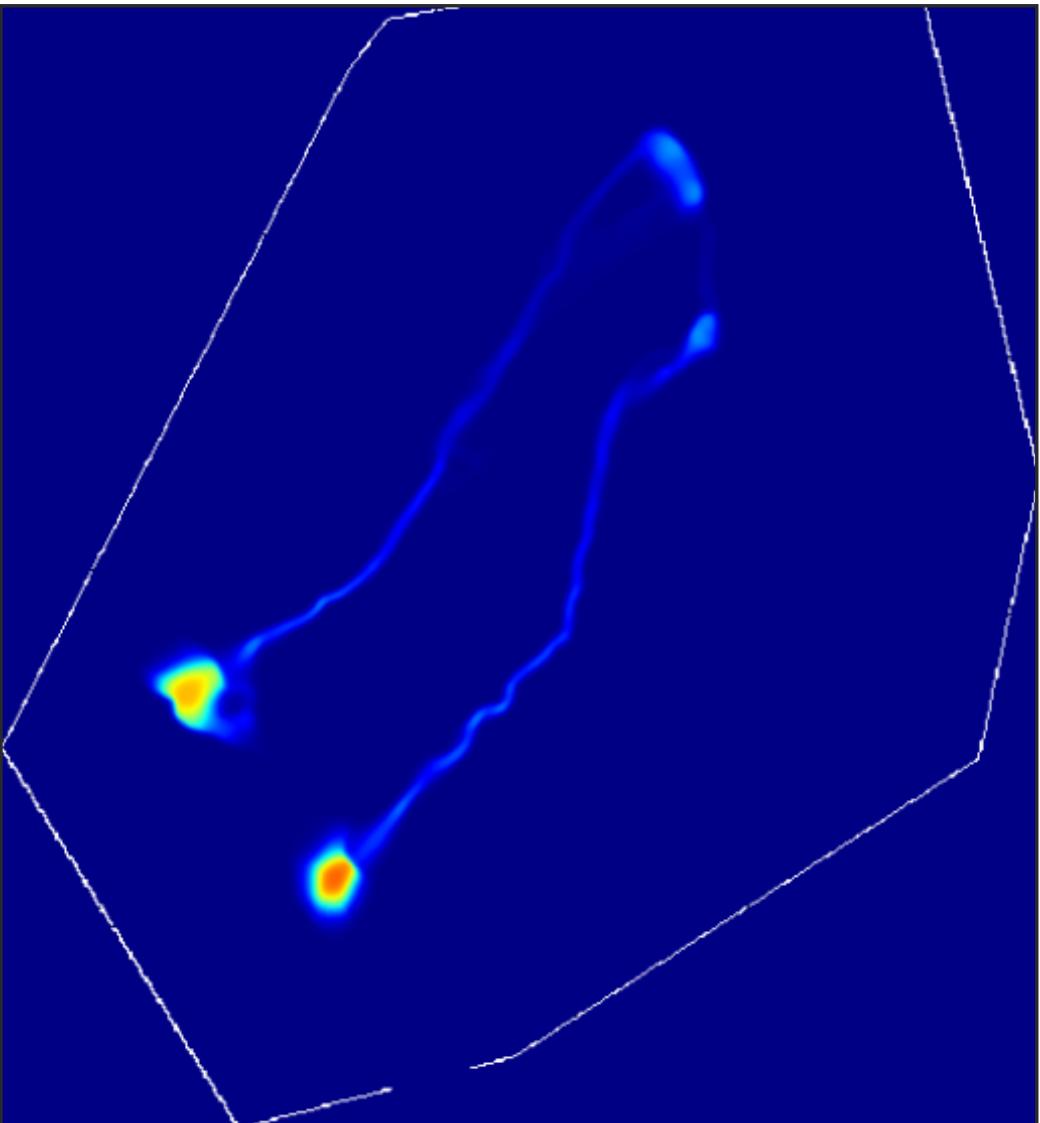


Lecture	Instructor	Time	Date	Day	Place	Topic
1	A	10:00-12:00	Aug 30	Wed	Aud 2	Introduction
2	A	10:00-12:00	Sep 1	Fri	Aud 2	Python Crash Course
3	A	10:00-12:00	Sep 6	Wed	Aud 2	
4	A	10:00-12:00	Sep 8	Fri	Aud 2	
5	A	10:00-12:00	Sep 13	Wed	Aud 2	
6	A	10:00-12:00	Sep 15	Fri	Aud 2	
7	A	10:00-12:00	Sep 20	Wed	Aud 2	
8	A	10:00-12:00	Sep 22	Fri	Aud 2	
9	A	10:00-12:00	Sep 27	Wed	Aud 2	
10	A	10:00-12:00	Sep 29	Fri	Aud 2	
11	M	10:00-12:00	Oct 4	Wed	Aud 2	Exploratory Data Analysis
12	M	10:00-12:00	Oct 6	Fri	Aud 2	
13	M	10:00-12:00	Oct 11	Wed	Aud 2	
14	M	10:00-12:00	Oct 13	Fri	Aud 2	
Holiday week		Holiday week				
15	M	10:00-12:00	Oct 25	Wed	Aud 2	Program Design
16	M	10:00-12:00	Oct 27	Fri	Aud 2	
17	M	10:00-12:00	Nov 1	Wed	Aud 2	
18	M	10:00-12:00	Nov 3	Fri	Aud 2	
19	M	10:00-12:00	Nov 8	Wed	Aud 2	Data Science
20	M	10:00-12:00	Nov 10	Fri	Aud 2	
21	M	10:00-12:00	Nov 15	Wed	Aud 2	
22	M	10:00-12:00	Nov 17	Fri	Aud 2	
23	M	10:00-12:00	Nov 22	Wed	Aud 2	in Practice
24	M	10:00-12:00	Nov 24	Fri	Aud 2	
25	M	10:00-12:00	Nov 29	Wed	Aud 2	
26	M	10:00-12:00	Dec 1	Fri	Aud 2	
27	M	No lecture	Dec 6	Wed	-	Mock Exam
28	M	10:00-12:00	Dec 8	Fri	Aud 2	
		09:00-13:00	Jan 3	Wed	TBD	Final Exam

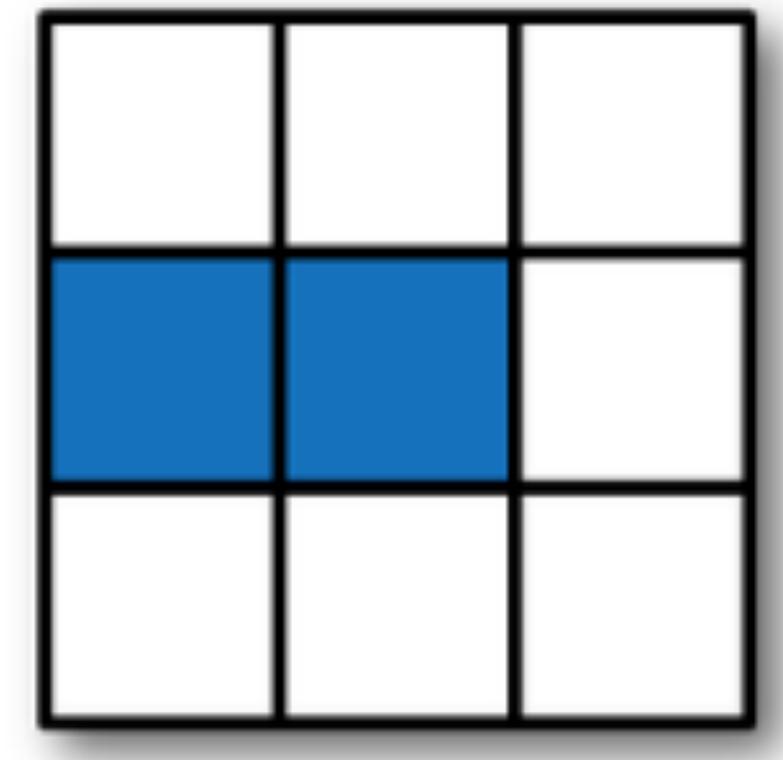
We are here

Today you will learn why, when, and how to use numpy

Scientific programming



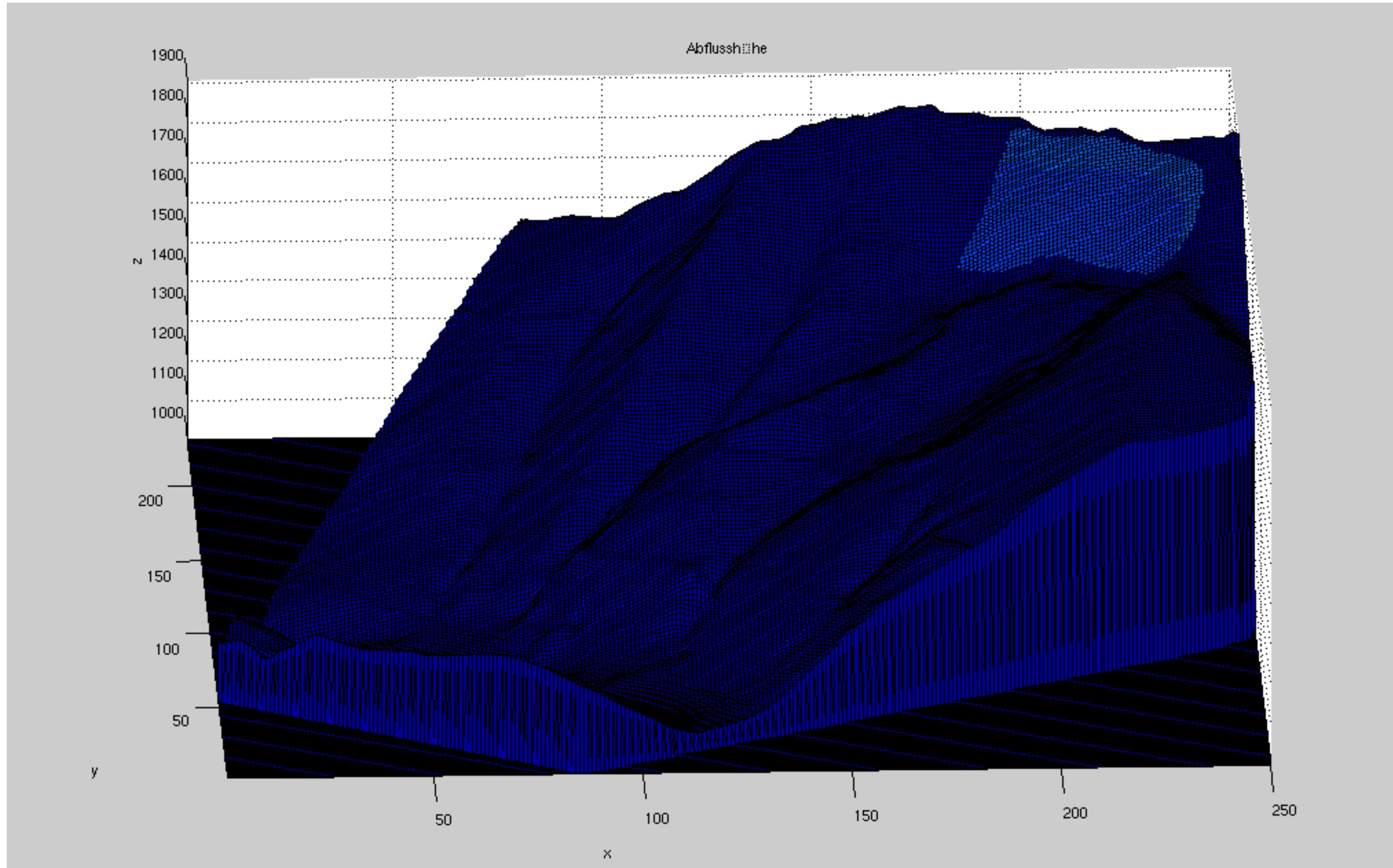
Array manipulation



Data processing



When we do repeated calculations with matrices,
standard Python is **slow**



Elementwise multiplication of two lists in Python

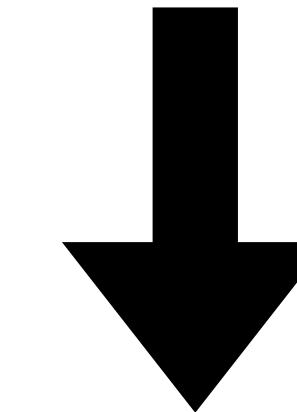
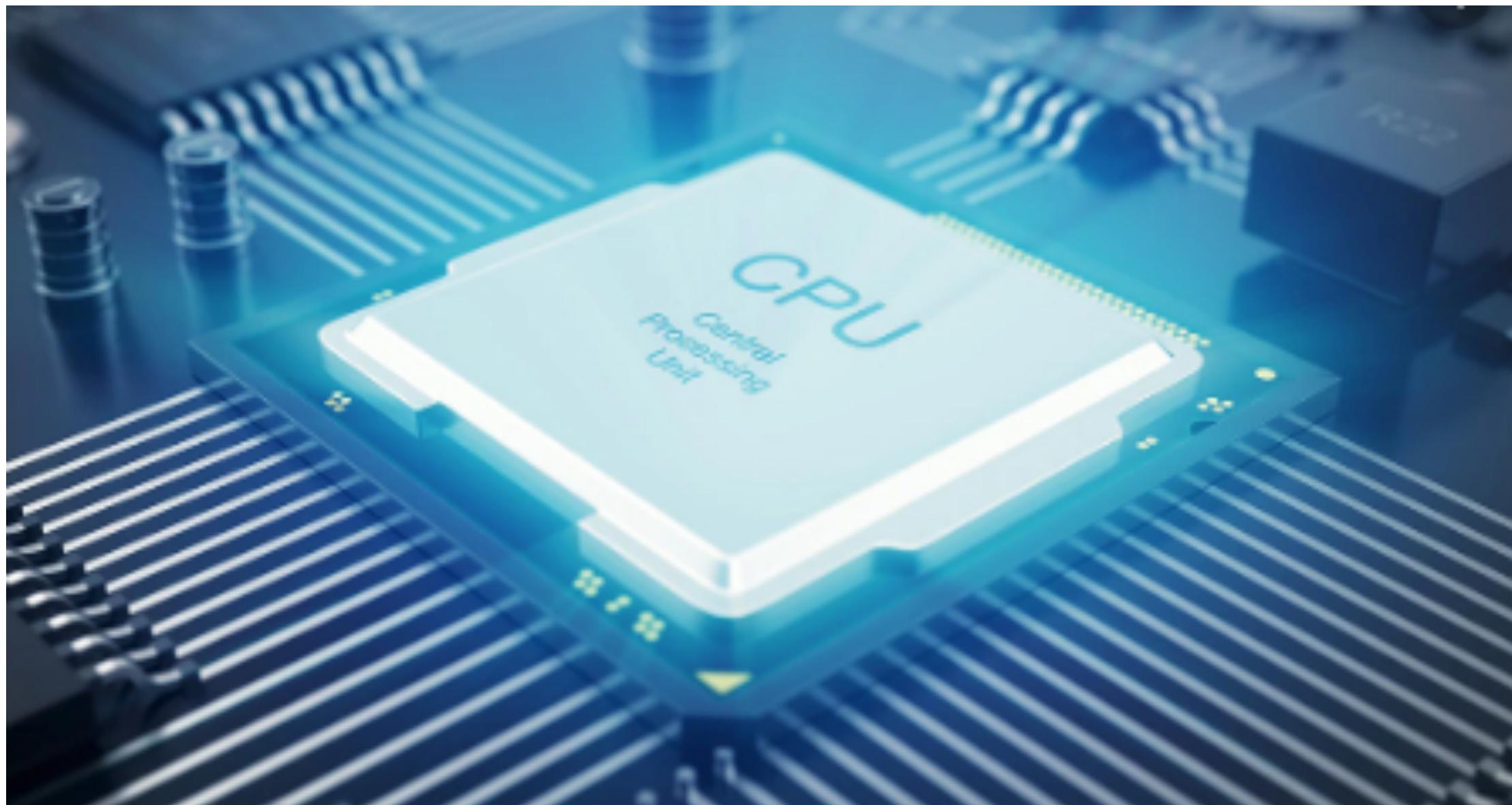
```
c = []
```

Elementwise multiplication of two lists in Python

```
c = []
for i in range(n):
    c.append(a[i]*b[i])
```

Vectorization is the ability to run operations on vectors instead of single numbers

```
c = []
for i in range(n):
    c.append(a[i]*b[i])
```



```
c = a*b
```

Array programming

numpy is optimized for vectorized operations

NumPy = Numerical Python

`ndarray` with vectorized operations

Vectorized mathematical functions

Data I/O

Linear algebra, random numbers

Using pre-compiled C

Use numpy whenever you run into limitations with lists and dicts

Speed

>1D-data

Masking / slicing

Use numpy whenever you run into limitations with lists and dicts

Speed

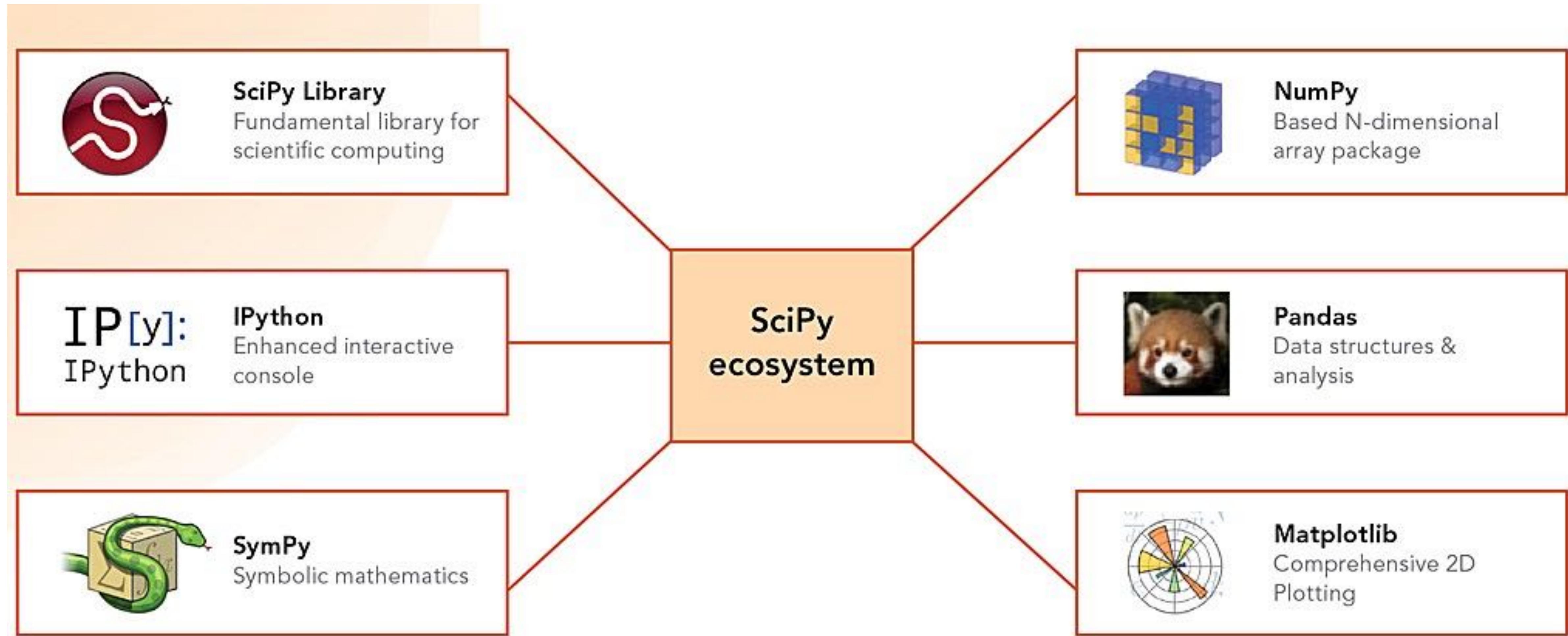
>1D-data

Masking / slicing

If you run into limitations with numpy, try Pandas:

Merging or reshaping data sets

More I/O options (Excel, SQL,...)



Why are we covering numpy now, and not pandas?

numpy is more fundamental:
pandas builds on numpy

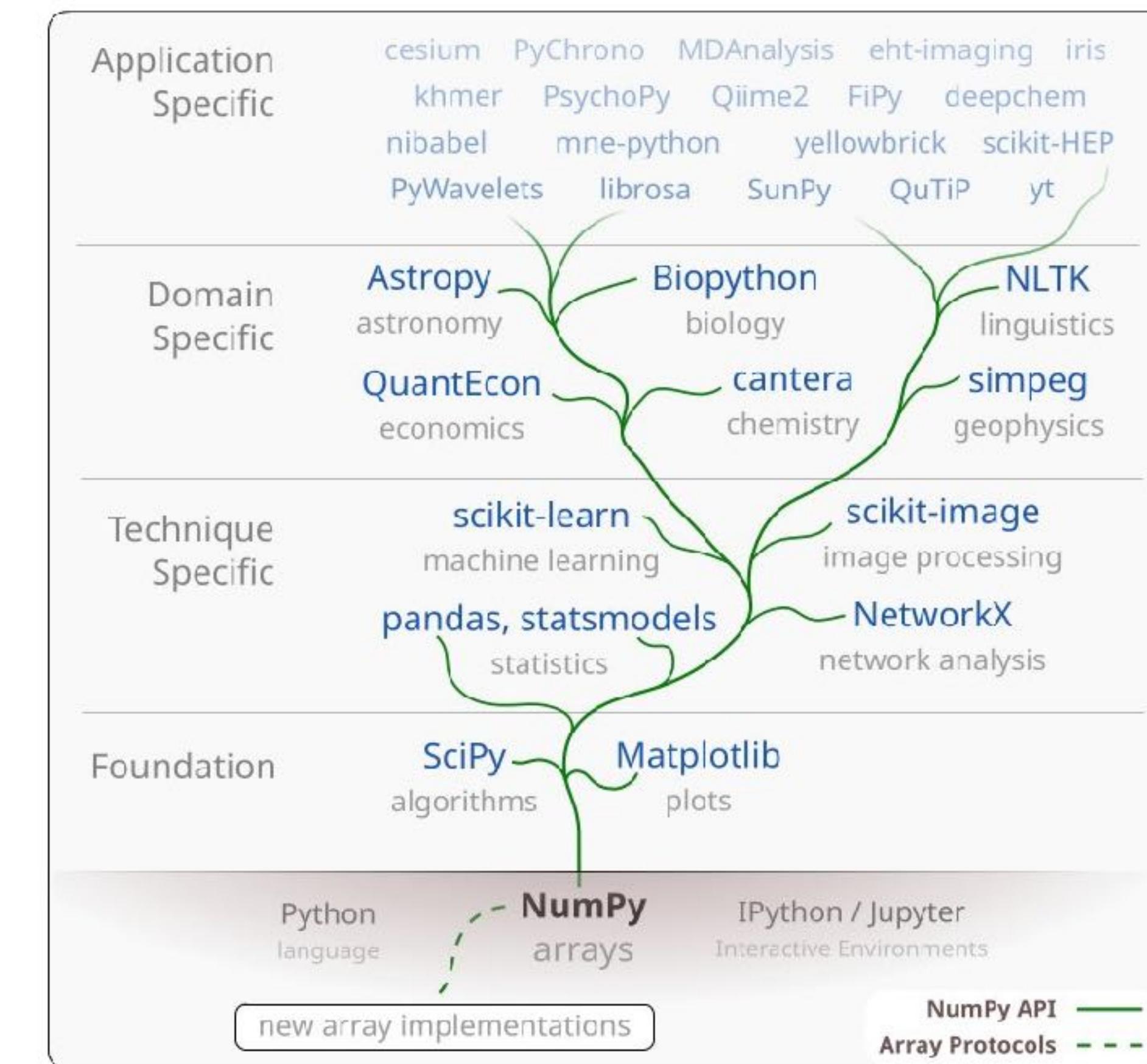


Figure 2. NumPy is the base of the scientific Python ecosystem. Essential libraries and projects that depend on NumPy's API gain access to new array implementations that support NumPy's array protocols (Fig. 3).

Jupyter notebook

Most important concepts from today

Array programming with numpy is 10-100+ times faster than pure Python for working with numerical, multidimensional data, through vectorized operations

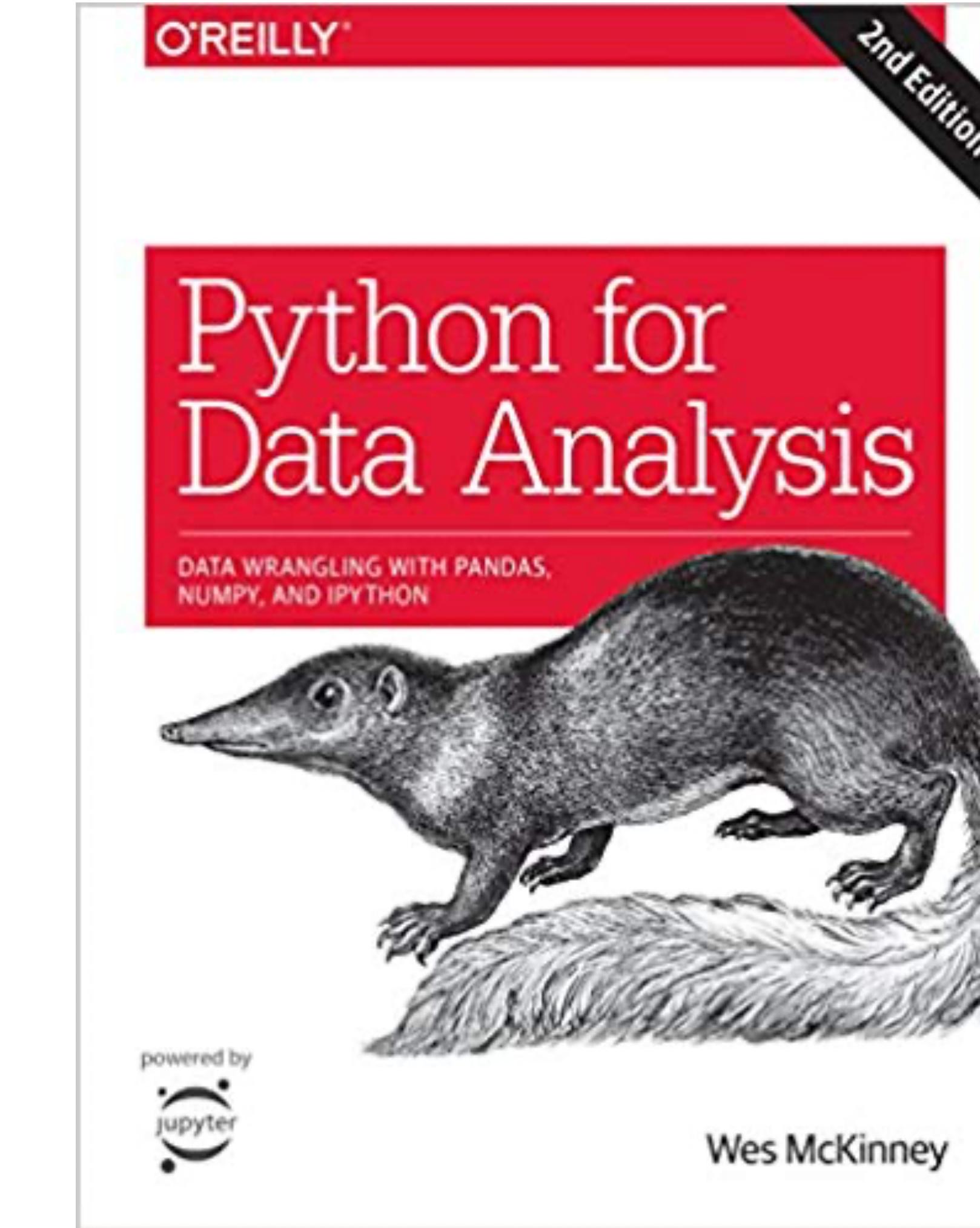
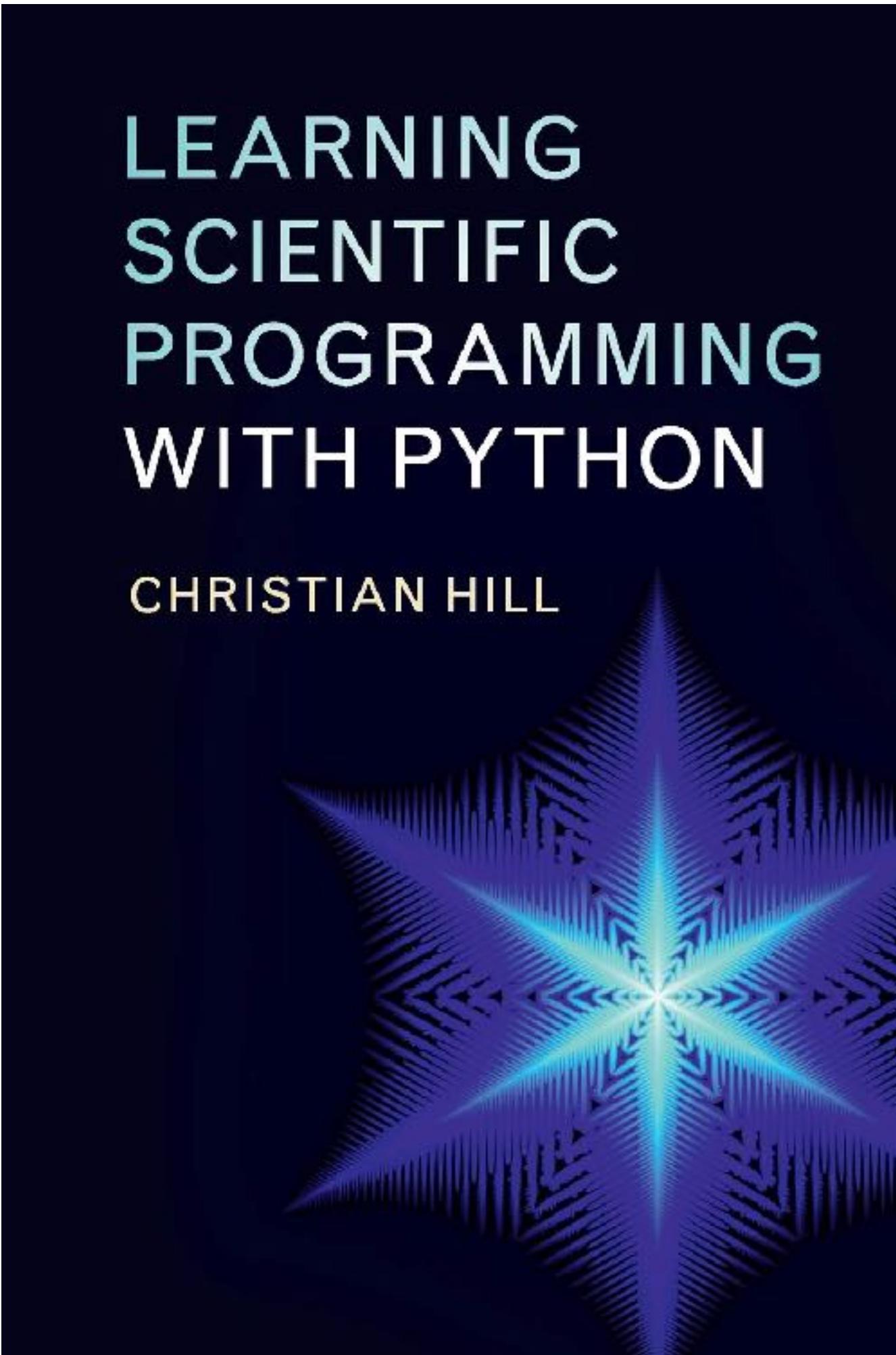
```
mask = (3 < x) & (x < 6.5)
```

numpy has great data masking capabilities



Index slicing is the syntax `M[lower:upper:step]` to extract part of an array

Sources and advanced materials for today's class



<https://www.oreilly.com/library/view/python-for-data/9781449323592/ch04.html>