Scientific Analysis with ArcGIS and SciPy

Shaun Walbridge
Kevin Butler

https://github.com/esrioceans/oceans-workshop-2016

High Quality PDF

Resources

Scientific Computing

Scientific Computing

Computers are now essential in all branches of science, but most researchers are never taught the equivalent of basic lab skills for research computing.

Good Enough Practices in Scientific Computing

Learn to take advantage of your #1 collaborator —

Scientific Computing

Computers are now essential in all branches of science, but most researchers are never taught the equivalent of basic lab skills for research computing.

Good Enough Practices in Scientific Computing

Learn to take advantage of your #1 collaborator —

your future self.

"Your self from 3 months ago doesn't answer email"

Python

Why Python?

- Accessible for new-comers, and the most taught first language in US universites
- Extensive package collection (56k on PyPI), broad user-base
- Strong glue language used to bind together many environments, both open source and commercial

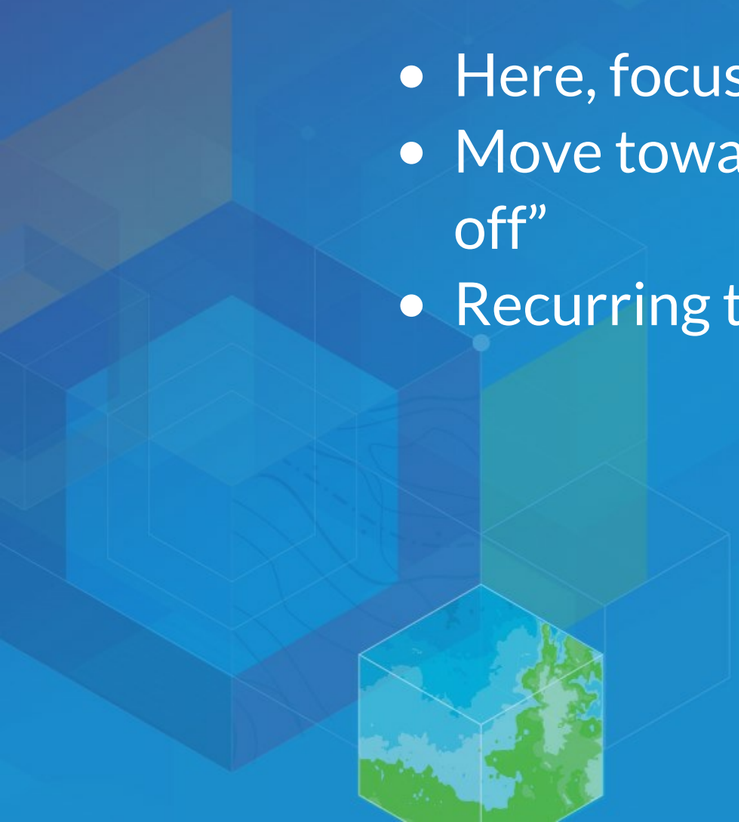
Why Python?

- Accessible for new-comers, and the most taught first language in US universites
- Extensive package collection (56k on PyPI), broad user-base
- Strong glue language used to bind together many environments, both open source and commercial

- Brand new to Python? Will walk you through all examples, show tools which use it.
- Resources include materials that for getting started, 75 minute DevSummit session

Python in ArcGIS

- Here, focus on SciPy stack, what's included out of the box
- Move toward maintainable, reusable code and beyond the "one-off"
- Recurring theme: multi-dimensional data structures



SciPy

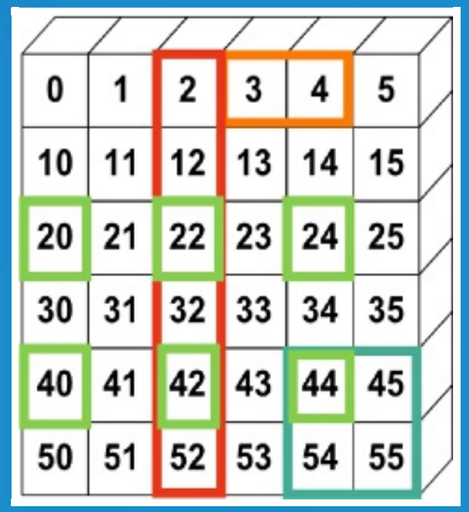
Why SciPy?

- Most languages don’t support things useful for science, e.g.:
  - Vector primitives
  - Complex numbers
  - Statistics
- Object oriented programming isn’t always the right paradigm for analysis applications, but is the only way to go in many modern languages
- SciPy brings the pieces that matter for scientific problems to Python.

Included SciPy

Package	KLOC	Contributors	Stars
matplotlib	121	439	4282
Nose	7	76	1014
NumPy	248	430	3502
Pandas	222	410	7342
SciPy	314	423	2670
SymPy	262	449	3280
Totals	1174	1879	

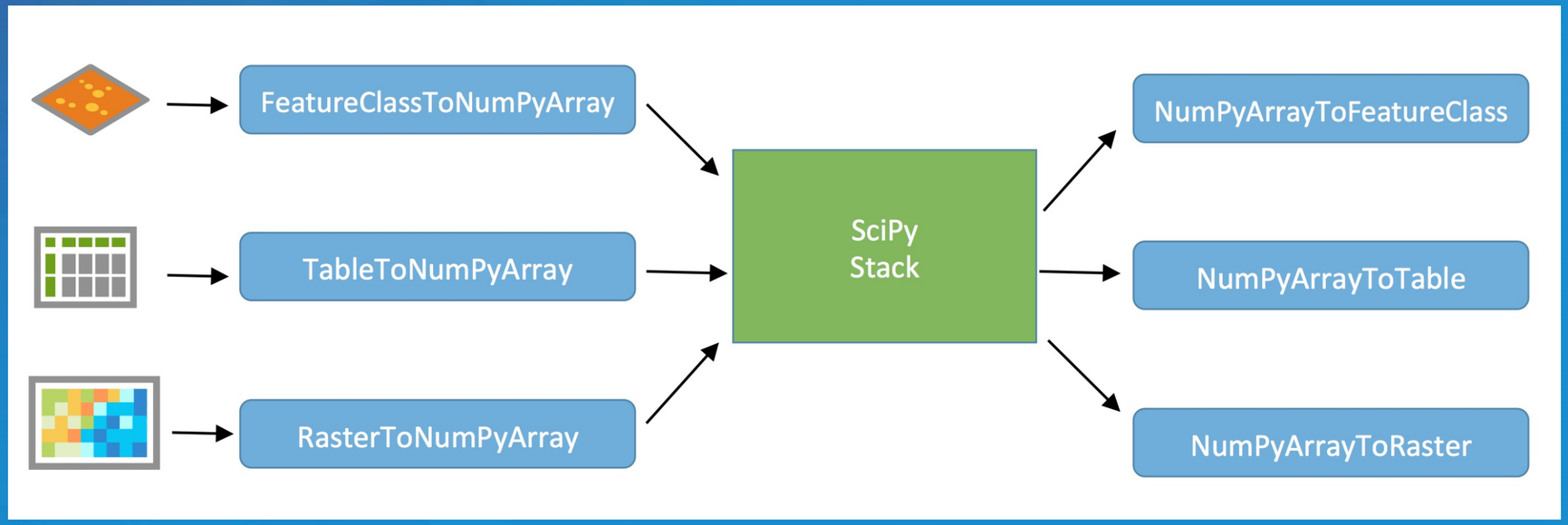
1. An array object of arbitrary homogeneous items
2. Fast mathematical operations over arrays
3. Random Number Generation



ArcGIS + NumPy

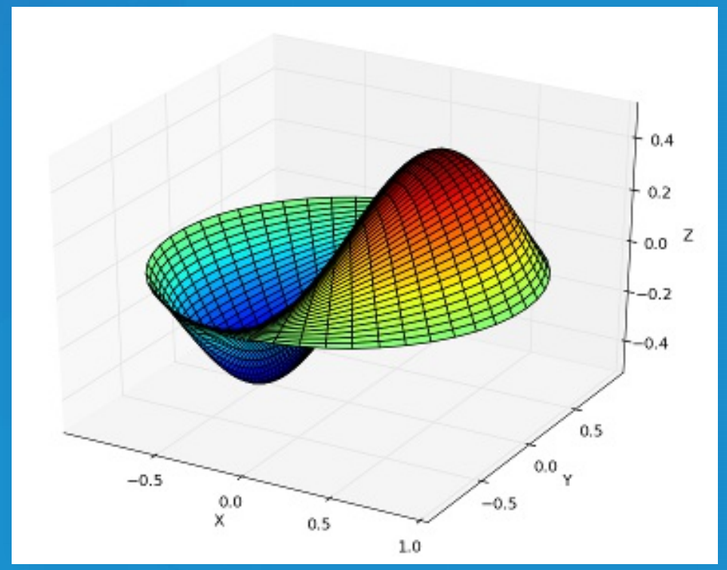
- ArcGIS and NumPy can interoperate on raster, table, and feature data.
- See Working with NumPy in ArcGIS
- In-memory data model. Can process by blocks for larger datasets.

ArcGIS + NumPy



matplotlib

- Plotting library and API for NumPy data
- Matplotlib Gallery



Computational methods for:

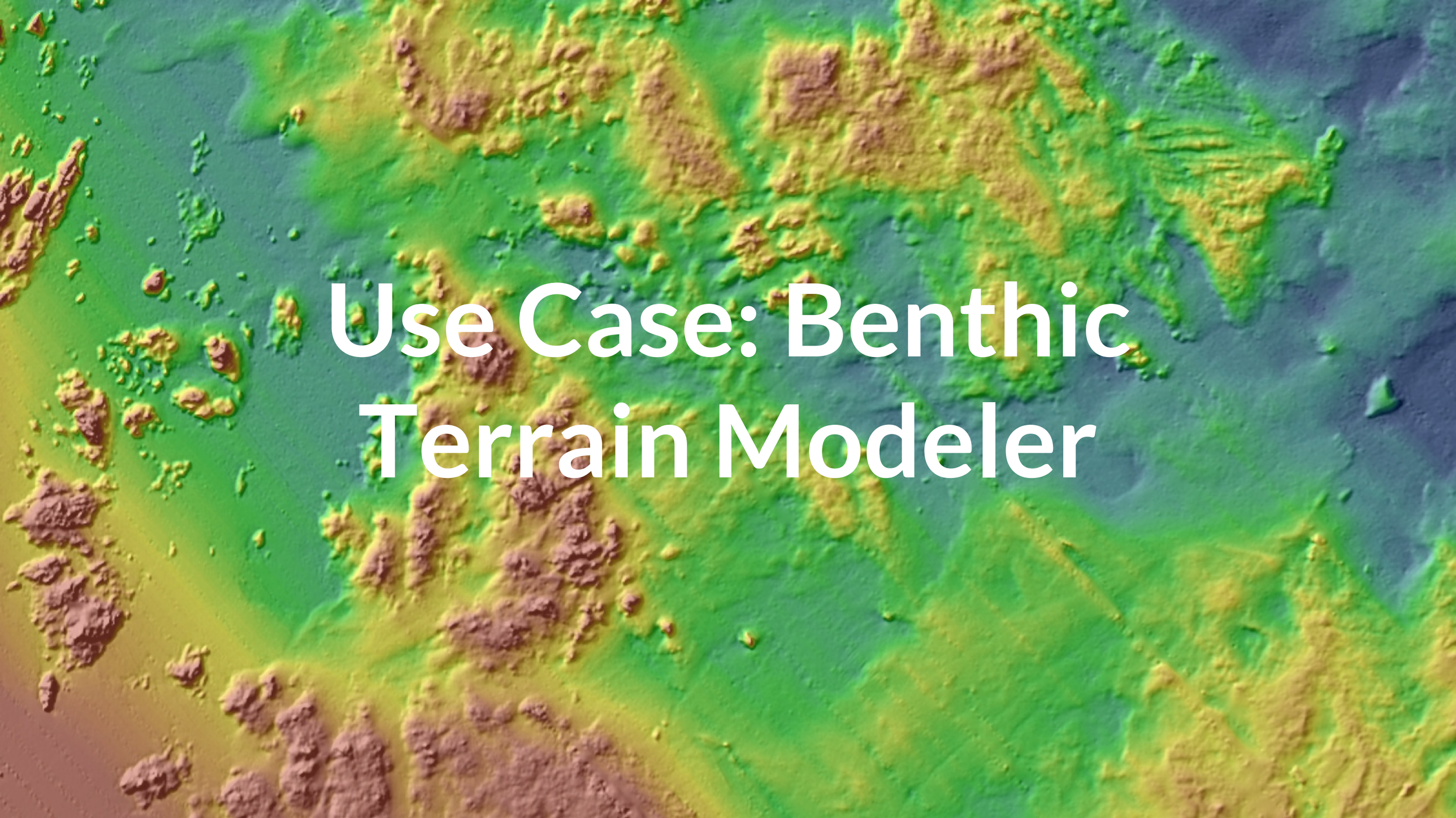
- Integration (scipy.integrate)
- Optimization (scipy.optimize)
- Interpolation (scipy.interpolate)
- Fourier Transforms (scipy.fftpack)
- Signal Processing (scipy.signal)
- Linear Algebra (scipy.linalg)
- Spatial (scipy.spatial)
- **Statistics** (scipy.stats)
- **Multidimensional image processing** (scipy.ndimage)

SciPy: Geometric Mean

- Calculating a geometric mean of an *entire raster* using SciPy (source)

$$\left(\prod_{i=1}^{n} a_i\right)^{1/n} = \sqrt[n]{a_1 \cdot a_2 \cdots a_n}$$

```
import scipy.stats
rast_in = 'data/input_raster.tif'
rast_as_numpy_array = arcpy.RasterToNumPyArray(ras
raster_geometric_mean = scipy.stats.stats.gmean(
    rast_as_numpy_array, axis=None)
```

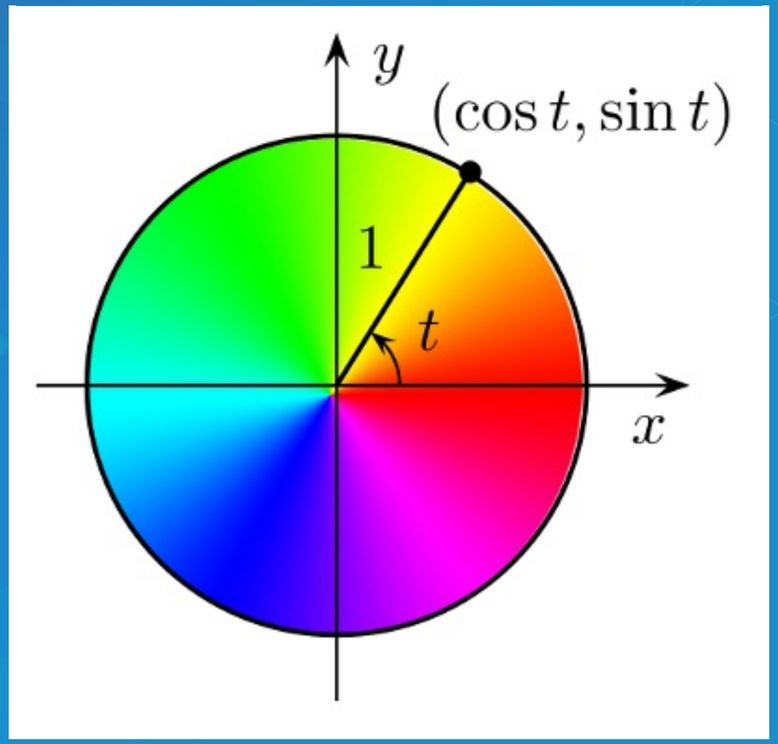


Use Case: Benthic Terrain Modeler

Benthic Terrain Modeler

- A Python Add-in and Python toolbox for geomorphology
- Open source, can borrow code for your own projects: https://github.com/EsriOceans/btm
- Active community of users, primarily marine scientists, but also useful for other applications
- Used in exercises

SciPy Statistics



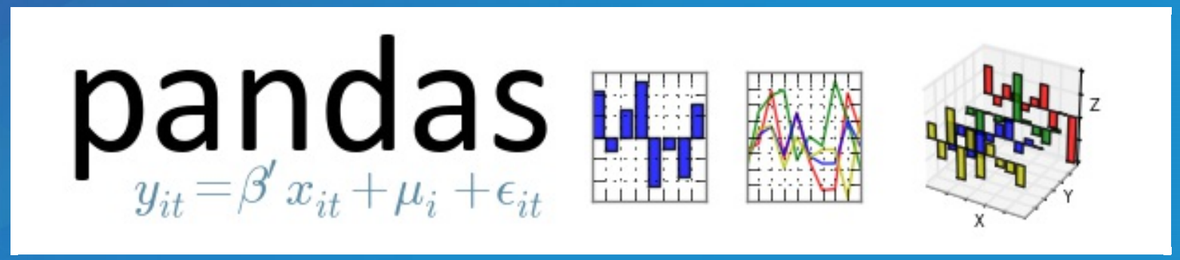
- Break down aspect into `sin()` and `cos()` variables
- Aspect is a circular variable — without this 0 and 360 are opposites instead of being the same value

SciPy Statistics

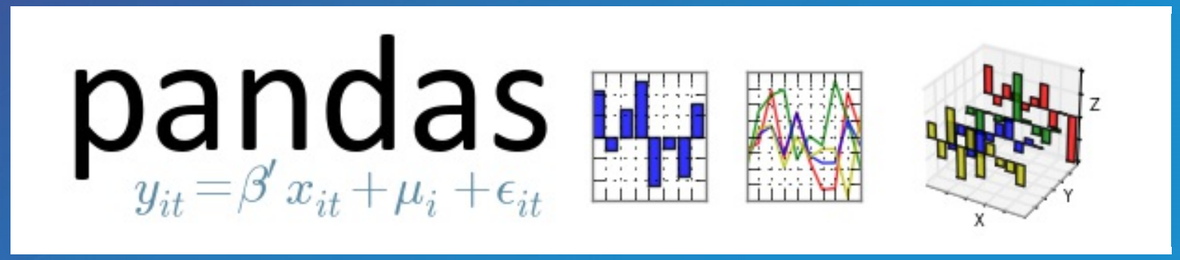
Summary statistics from SciPy include circular statistics (Source)

```
import scipy.stats.morestats

ras = "data/aspect_raster.tif"
r = arcpy.RasterToNumPyArray(ras)

morestats.circmean(r)
morestats.circstd(r)
morestats.circvar(r)
```

Pandas



- **Pan**el **Da**ta — like R "data frames"
- Bring a robust data *analysis* workflow to Python
- Data frames are fundamental — treat tabular (and multi-dimensional) data as a labeled, indexed series of observations.



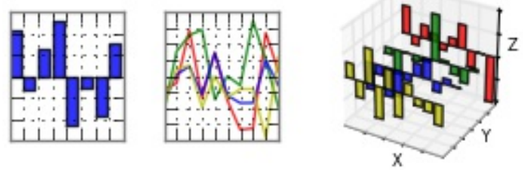
(Source

```
import pandas

data = pandas.read_csv('data/season-ratings.csv')
data.columns
```

```
Index([u'season', u'households', u'rank',
       u'tv_households', u'net_indep',
       u'primetime_pct'], dtype='object')
```

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$




```
majority_simpsons = data[data.primetime_pct > 50]
```

	season	households	tv_households	net_indep	primetime_pct
0	1	13.4m[41]	92.1	51.6	80.751174
1	2	12.2m[n2]	92.1	50.4	78.504673
2	3	12.0m[n3]	92.1	48.4	76.582278
3	4	12.1m[48]	93.1	46.2	72.755906
4	5	10.5m[n4]	93.1	46.5	72.093023
5	6	9.0m[50]	95.4	46.1	71.032357
6	7	8.0m[51]	95.9	46.6	70.713202
7	8	8.6m[52]	97.0	44.2	67.584098
8	9	9.1m[53]	98.0	42.3	64.383562
9	10	7.9m[54]	99.4	39.9	60.916031
10	11	8.2m[55]	100.8	38.1	57.466063
11	12	14.7m[56]	102.2	36.8	53.958944
12	13	12.4m[57]	105.5	35.0	51.094891

Where and How Fast?

Where Can I Run This?

- Now:
  - ArcGIS Pro (64-bit) Standalone Python Install for Pro
  - ArcGIS Desktop at 10.4: 32-bit, Background Geoprocessing (64-bit), Server (64-bit), Engine (32-bit)
    - Both now ship with Scipy Stack (sans IPython)
  - MKL enabled NumPy and SciPy everywhere
  - Older releases: NumPy: ArcGIS 9.2+, matplotlib: ArcGIS 10.1+, SciPy: 10.4+, Pandas: 10.4+

How Does It perform?

- Built with Intel's Math Kernel Library (MKL) and compilers—highly optimized Fortran and C under the hood.
- Automated parallelization for executed code

MKL Performance Chart

SciPy Hands-on Activity

SciPy Hands-on Activity

SciPy Exercise