Python for GIS Librarians:
Toward a Data Science Core Competency for Geospatial Librarianship

Phil White
Earth Sciences & Environment Librarian
University of Colorado Boulder
philip.white@colorado.edu

http://bit.ly/Phil-WAMLpy
WAML San Diego 2018
# field mapping: which fields from quad index table will be joined
fms = arcpy.FieldMappings()
fms.addTable(quadIndex)
fields_sequence = ['USGS_QD_ID', 'QUAD_NAME']
fields_removed = [f.name for f in fms.fields if f.name not in fields_sequence]
for field in fields_removed:
    fms.removeFieldMap(fms.findFieldMapIndex(field))

fms_out = arcpy.FieldMappings()
fms_out.addTable(name+decdeg)
for field in fields_sequence:
    mapping_index = fms.findFieldMapIndex(field)
    field_map = fms.fieldMappings[mapping_index]
    fms_out.addFieldMap(field_map)

# joins specified fields from quad index table to DecDeg table, adds name+DecDeg_Quad feature layer to gdb
arcpy.analysis.SpatialJoin(name+decdeg, quadIndex, name+quad, "JOIN_ONE_TO_ONE", "KEEP_ALL", fms_out, "INTERSECT", None)

# field mapping: which fields from GNIS table will be joined
fms = arcpy.FieldMappings()
fms.addTable(gnisPoints)
fields_sequence = ['FEATURE_A', 'FEATURE_CL']
fields_removed = [f.name for f in fms.fields if f.name not in fields_sequence]
for field in fields_removed:
    fms.removeFieldMap(fms.findFieldMapIndex(field))

fms_out = arcpy.FieldMappings()
fms_out.addTable(name+quad)
for field in fields_sequence:
field mapping: which fields from quad index table will be joined

def field_mapping_quad_index(name, decdeg):
    fms = arcpy.FieldMappings()
    fms.addTable(quadIndex)
    fields_sequence = ['USGS_QD_ID', 'QUAD_NAME', 'quad_index', 'quad_number', 'quad_area', 'quad_perimeter', 'quad_latitude', 'quad_longitude', 'quad_corrected_latitude', 'quad_corrected_longitude']
    fields_removed = [f.name for f in fms.fields]
    for field in fields_removed:
        fms.removeFieldMap(fms.findFieldMapIndex(field))

    fms_out = arcpy.FieldMappings()
    fms_out.addTable(name+decdeg)
    for field in fields_sequence:
        mapping_index = fms.findFieldMapIndex(field)
        field_map = fms.fieldMappings[mapping_index]
        fms_out.addFieldMap(field_map)

    # joins specified fields from quad index
def spatial_junction(name, decdeg):
        arcpy.analysis.SpatialJoin(name+decdeg, quadIndex)

# field mapping: which fields from GNIS to join
def field_mapping_GNIS(name, decdeg):
    fms = arcpy.FieldMappings()
    fms.addTable(gnisPoints)
    fields_sequence = ['FEATURE_NAME', 'FEATURE_CODE', 'GEOграф', 'GEOGRAP', 'GEOG_LATITUDE', 'GEOG_LONGITUDE', 'GEOG_CORRECTED_LATITUDE', 'GEOG_CORRECTED_LONGITUDE']
    fields_removed = [f.name for f in fms.fields]
    for field in fields_removed:
        fms.removeFieldMap(fms.findFieldMapIndex(field))

    fms_out = arcpy.FieldMappings()
    fms_out.addTable(name+decdeg)
    for field in fields_sequence:
        mapping_index = fms.findFieldMapIndex(field)
        field_map = fms.fieldMappings[mapping_index]
        fms_out.addFieldMap(field_map)
Overview of Presentation

- What is Python?
- Why did I learn it? Why should you?
- Learning Resources
- Toward a Data Science Core Competency
Any Python users in the house?
Any Python users in the house?

Any R users in the house?
What is Python?
What is Python?

- Programming language
What is Python?

- Programming language
- Free and open source
What is Python?

- Programming language
- Free and open source
- Among the most used programming languages in the world
What is Python?

- Programming language
- Free and open source
- Among the most used programming languages in the world
- "There's a Python library for that"

Image: wikimedia
Quick Vocab Lesson
Quick Vocab Lesson

**Library**: Collection of functions/methods that you can call up without having to write code from scratch.
Quick Vocab Lesson

**Library:** Collection of functions/methods that you can call up without having to write code from scratch.

- Numpy: math, stats
- xml: working with xml
- Pandas: working with tabular data
- Requests: calling URLs
Quick Vocab Lesson

**Library**: Collection of functions/methods that you can call up without having to write code from scratch.

- Numpy: math, stats
- xml: working with xml
- Pandas: working with tabular data
- Requests: calling URLs

**Script**: A series of commands that you run to automate tasks. Python is a scripting language.
Quick Vocab Lesson

**Library**: Collection of functions/methods that you can call up without having to write code from scratch.

- Numpy: math, stats
- xml: working with xml
- Pandas: working with tabular data
- Requests: calling URLs

**Script**: A series of commands that you run to automate tasks. Python is a scripting language.

**API** (Application Programming Interface): Method of data exchange.
Why did I learn Python?

(still learning)
Why did I learn Python?

(still learning)

- Software Carpentry workshop 2.5 years ago... about 2 hours of Python
Why did I learn Python?
(still learning)

- Software Carpentry workshop 2.5 years ago... about 2 hours of Python
- But in truth, necessity is the mother of invention
Why did I learn Python?

(still learning)

- Software Carpentry workshop 2.5 years ago... about 2 hours of Python

- But in truth, necessity is the mother of invention

- From there, mostly self taught (YouTube! Code Academy)
Why did I learn Python?

(still learning)

- Software Carpentry workshop 2.5 years ago... about 2 hours of Python
- But in truth, necessity is the mother of invention
- From there, mostly self taught (YouTube! Code Academy)
- About 2-3 months gnashing of teeth, pounding of head, tear drops
Why learn Python?
Why learn Python?

- Do it for Open Science!
Why learn Python?

- Do it for Open Science!
- Do it for your students!
Why learn Python?

- Do it for Open Science!
- Do it for your students!
- Is code the future of GIS?
Why learn Python?

- Do it for Open Science!
- Do it for your students!
- Is code the future of GIS?
- Cataloging; metadata creation & manipulation...
Why learn Python?

- Do it for Open Science!
- Do it for your students!
- Is code the future of GIS?
- Cataloging; metadata creation & manipulation...
  ...but so much more!
Why learn Python?

- Do it for Open Science!
- Do it for your students!
- Is code the future of GIS?
- Cataloging; metadata creation & manipulation...
  ...but so much more!
- Data reference:
Why learn Python?

- Do it for Open Science!
- Do it for your students!
- Is code the future of GIS?
- Cataloging; metadata creation & manipulation...
  ...but so much more!
- Data reference: 1. Data munging & transformations
  2. Mining, scraping, APIs, downloading
Example 1:

Data that doesn't play nice.

(Massage the data!!)
Why Learn Python?
Why Learn Python?

- Python is the most common language used in GIS
Why Learn Python?

- Python is the most common language used in GIS
  - Built right in to ArcGIS, QGIS 🌍
Why Learn Python?

- Python is the most common language used in GIS
  - Built right in to ArcGIS, QGIS
- Automate repetitive geoprocessing tasks
Why Learn Python?

- Python is the most common language used in GIS
  - Built right in to ArcGIS, QGIS
- Automate repetitive geoprocessing tasks
- Free geocoding
Why Learn Python?

- Python is the most common language used in GIS
  - Built right in to ArcGIS, QGIS
- Automate repetitive geoprocessing tasks
- Free geocoding
- Fast and efficient
Why Learn Python?

- Python is the most common language used in GIS
  - Built right in to ArcGIS, QGIS
- Automate repetitive geoprocessing tasks
- Free geocoding
- Fast and efficient
- Makes you feel good at your job!
Example 2:

Save hours of work for yourself

(or your heady fwends)
Learning Resources
Learning Resources

- Software Carpentry
- W3Schools
- Code Academy
- Your institution?
- Google!
Learning Resources

- Software Carpentry
- W3Schools
- Code Academy
- Your institution?
- Google!

Recommended Installation

- Anaconda
Learning Resources

- Software Carpentry
- W3Schools
- Code Academy
- Your institution?
- Google!

Recommended Installation

- Anaconda

Interfaces

- PyCharm
- Spyder
Learning Resources

- Software Carpentry
- W3Schools
- Code Academy
- Your institution?
- Google!

Recommended Installation

- Anaconda

Interfaces

- PyCharm
- Spyder

Anaconda comes with Jupyter Notebooks!!
Example 3:

Jupyter Notebooks will help you break through!

(Bonus: Mine data from an API!!)
GIS Libraries for Python

- **ArcPy**: Automate ArcGIS tasks
- **PyQGIS**: Automate QGIS tasks (open source)
- **GeoPandas**: Adds geospatial operations to Pandas (open)
- **Shapely**: Geometry (buffer, intersect, union, etc.) (open)
- **Rasterio**: Raster operations (open)
- **GeoPy**: Geocoding and distance calculations (open)
- **Plotly**: Visualization (free(mium))
- **So Many Others!!!**
Toward a Data Science Core Competency
Toward a Data Science Core Competency

1. Our patrons are increasingly skilled in tech.
Toward a Data Science Core Competency

1. Our patrons are increasingly skilled in tech.

2. GIS Librarians serve a broad community... any discipline!
Toward a Data Science Core Competency

1. Our patrons are increasingly skilled in tech.
2. GIS Librarians serve a broad community... any discipline!
3. There is so much data out there... if we can just get to it!
Toward a Data Science Core Competency

1. Our patrons are increasingly skilled in tech.

2. GIS Librarians serve a broad community... any discipline!

3. There is so much data out there... if we can just get to it!

4. We should embrace open technologies.
Can you learn Python?
Yes We Can!
presentation.close()

print('Thank You')

for q in audience:
    query = ask(q)

email = 'philip.white@colorado.edu'
print(email)