esri Canada | Education & Research

Bringing GIS Analysis to Life using Python Notebooks

Michael G. Leahy PhD – mleahy@esri.ca

2020 GIS in Education and Research Conference

Workshop Agenda

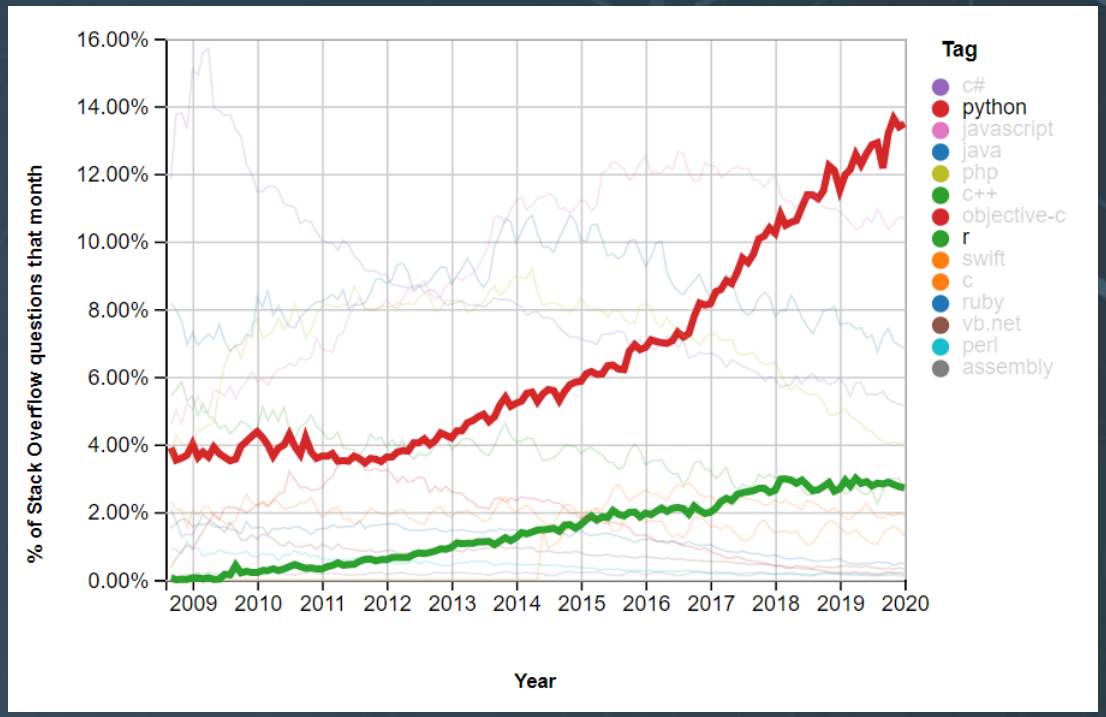
- Introduction: Jupyter Notebooks -> ArcGIS Notebooks
  - Why? What?

- Getting Started:
  - Basics of working with Jupyter Notebooks
  - Notebooks in ArcGIS Pro
  - Hosted Notebooks in ArcGIS Enterprise, ArcGIS Online

- Geospatial Data Science Notebook samples:
  - Data wrangling
  - Data exploration
  - Data analysis

Python

- Arguably **the** most popular programming language today
- Why?
  - Active/supportive community
  - Big Data, ML/DL/AI
  - Many high-quality libraries
  - Reliable and efficient
  - Highly accessible (easy to learn and use)



https://insights.stackoverflow.com/trends

Jupyter Notebook

- A popular way to compose documents that include:
  - Rich formatted text
  - Embedded images
  - Multimedia
  - Math formulas
  - All interlaced with live executable code and visualization of outputs

- Why?
  - Everything in one place
  - Sharable
  - Language independent
  - Customizable
  - Reproducibility & Transparency
  - Teaching
  - Iterative exploration

  - Python!

ArcGIS Notebooks

- Notebooks – around since '80s
  > IPython (2001)
    > Jupyter (2014, multi-language)

- ArcGIS:
  - Geoprocessing with Python since ArcGIS 9.0 (2004)

- 2016: ArcGIS Notebooks =
  Jupyter Notebook server
  + Python
  + ArcGIS API for Python (2016)
  + ArcPy

- 2019: Hosted Notebooks =
  ArcGIS Notebook
  + ArcGIS Enterprise 10.7.1+
  + Notebook Server for ArcGIS
  (or simply ArcGIS Online … soon)

- 2020: ArcGIS Pro Notebooks =
  ArcGIS Pro 2.5+ (2020)
  + ArcGIS Notebooks

What can ArcGIS Notebooks do?

Data Engineering (Wrangling)

(Spatial) Data Analysis

ML / DL / AI

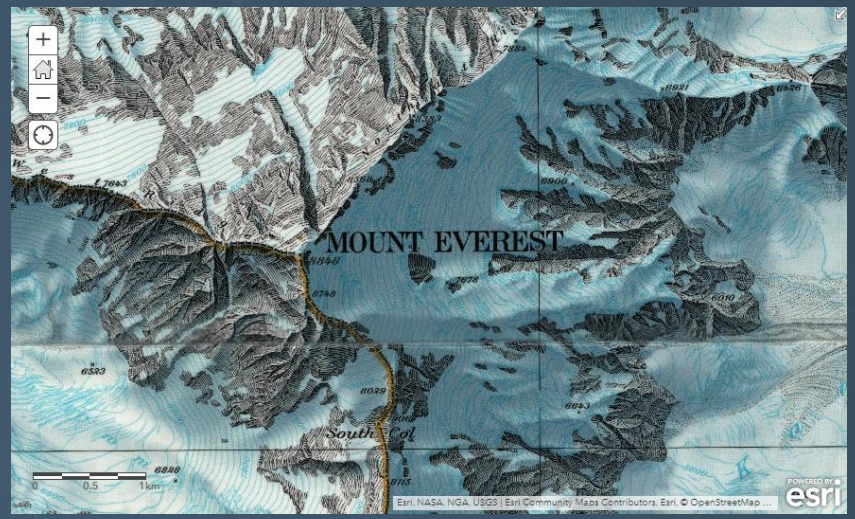
Automation

Collaboration

Storytelling

Iteration & Experimentation (Learning)

ArcGIS StoryMaps...



help you turn this...
(simple web maps)

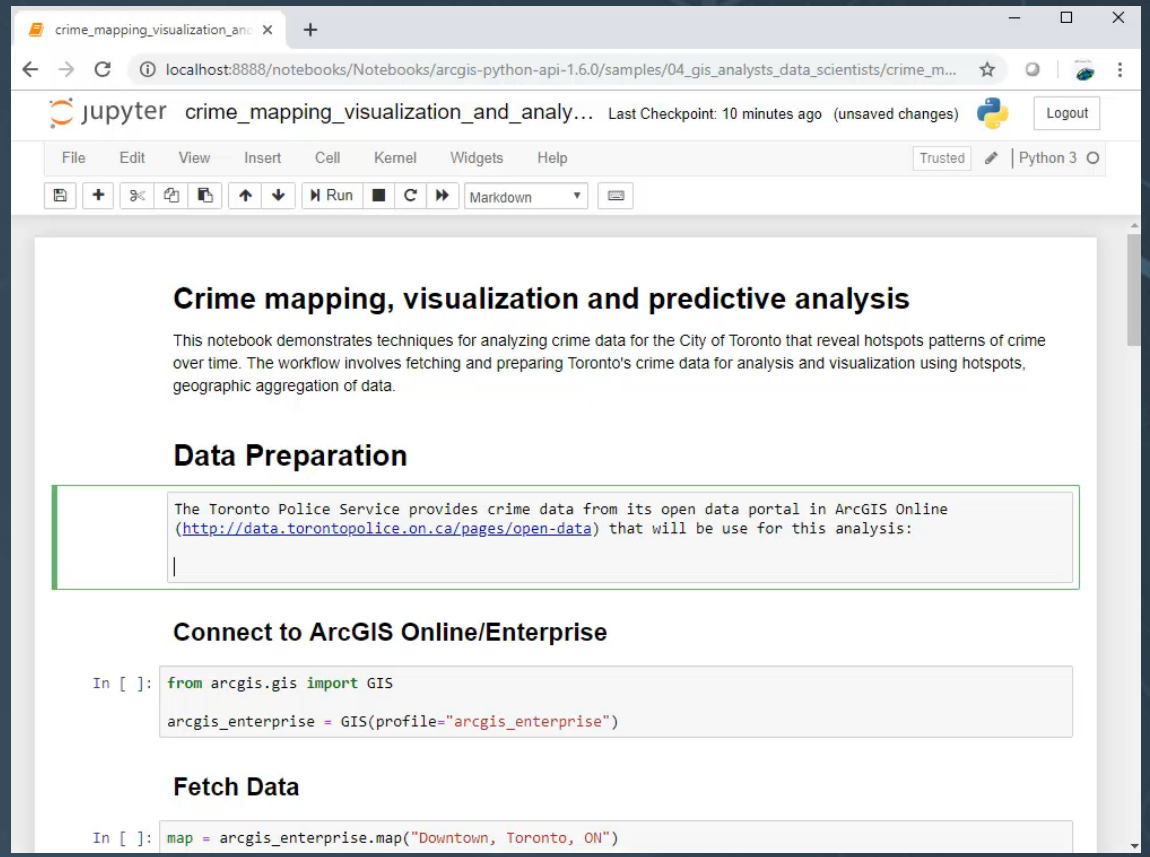
...into this.

ArcGIS Notebooks…

crime_mapping_visualization_and_analysis_toronto.py

```
# Load the layer as a spatial data frame, and save to disk for use with arcpy analysis:
from arcgis.features import SpatialDataFrame
mci_sdf = SpatialDataFrame.from_layer(mci_layer.layers[0])

# Load the arcpy module, and create a Space Time Cube from the crime data
import arcpy
arcpy.env.overwriteOutput = True

# Using downloaded copy of data: D:\ScratchFiles\MCI_2014_to_2017.shp
arcpy.stpm.CreateSpaceTimeCube(r"D:\ScratchFiles\MCI_2014_to_2017.shp",
 r"D:\ScratchFiles\MCI_2014_to_2017.nc", "Date", None, "1 Months", "END_TIME", None, "500
 Meters", None, "HEXAGON_GRID", None, None)

# Execute Emerging Hotspot Analysis
arcpy.stpm.EmergingHotSpotAnalysis(r"D:\ScratchFiles\MCI_2014_to_2017.nc", "COUNT",
 r"D:\ScratchFiles\MCI_2014_to_2017_hotspots.shp", None, 1, None, "FIXED_DISTANCE", None,
 "ENTIRE_CUBE")

from arcgis.features import FeatureLayerCollection

# Upload shapfile as zip archive, and publish new layer:
hotspots_shp = arcgis_enterprise.content.add({type: "Shapefile"},
 r'D:\ScratchFiles\MCI_2014_to_2017_hotspots.zip')
hotspots_layer = hotspots_shp.publish()

# Update symbology of layer from an existing item used as a template:
template_layer = arcgis_enterprise.content.get("a86e128a1e5b472ca926be0ead7e0b10").layers[0]
flc = FeatureLayerCollection.fromitem(hotspots_layer)
flc.layers[0].manager.update_definition({"drawingInfo": template_layer.properties['drawingInfo']})
```

help you turn this…
(python scripts)



…into this.

Getting Started...

esri Canada | Education & Research

https://hed.esri.ca highered@esri.ca

QUESTIONS?

Michael G. Leahy PhD – mleahy@esri.ca

ListServ: https://hed.esri.ca/studentlistserv

Twitter: @GIS4HEd